

Université de Nantes  
Polytech Nantes  
IRCCyN UMR 6597

Mémoire scientifique présenté par

**Nicolas Normand**

en vue de l'obtention d'une

**HABILITATION À DIRIGER DES RECHERCHES**

# Projections et distances discrètes

Soutenu le 29 novembre 2012  
devant le jury composé de :

<b>Françoise Peyrin</b>	Présidente
<b>Jean-Marc Chassery</b>	Rapporteur
<b>David Coeurjolly</b>	Rapporteur
<b>Édouard Thiel</b>	Rapporteur
<b>Imants Svalbe</b>	Invité
<b>Jeanpierre Guédon</b>	Directeur



# Curriculum Vitæ

## 1 Identification

**Nom** Normand

**Prénom** Nicolas

**Date de naissance** 24 avril 1969

**Situation** Marié, 2 enfants

**Grade** Classe normale, 6<sup>e</sup> échelon

**Établissement** Université de Nantes (Polytech Nantes, département Informatique)

**Section CNU** 27

**Unité de recherche d'appartenance** Laboratoire IRCCyN (UMR 6597) dans l'équipe Image Vidéo Communication (IVC)

**Thèse de doctorat** Nicolas Normand : Représentation d'images et distances discrètes basées sur les éléments structurants à deux pixels. Université de Nantes, janvier 1997. Jury composé de : Jean Serra (président), Annick Montanvert (rapporteuse), Didier Arques (rapporteur), Joseph Ronsin, Jeanpierre Guédon, Christian Viard-Gaudin, Dominique Barba (directeur).

## 2 Activités pédagogiques

### 2.1 Déroulement de carrière

**1<sup>er</sup> septembre 1997** Maître de conférences stagiaire

**1<sup>er</sup> septembre 1998** Maître de conférences titulaire

**1<sup>er</sup> septembre 2008** Prime d'encadrement doctoral et de recherche

**1<sup>er</sup> septembre 2009 - 30 aout 2010** CRCT en Australie à Monash University, Melbourne

### 2.2 Enseignements

Ma charge d'enseignement, qui a oscillé entre 350 et 400 h pendant dix ans (entre 1997 et 2007), a été réduite à 250 h depuis 2009 (sauf en 2010-2011 avec 348h). Dans ce but, je n'ai conservé que les matières liées aux thèmes de l'équipe IVC (géométrie algorithmique, image, réseau) tout en gardant quelques interventions extérieures à Polytech Nantes (CNAM et INSA Rennes en début de carrière, IUT Nantes et Faculté des Sciences et techniques plus récemment).

L'enseignement de la spécialité Informatique de Polytech Nantes est semestrialisé sur les deux années du cycle préparatoire (PEIP semestres 1 à 4) et les trois années du cycle ingénieur (semestres 5 à 10). La nomenclature du nombre d'heures est CM, TD, TP et projet.

**Semestre 2** Projet PeiP (0 0 0 4)

**Semestre 5** Accueil informatique (0 0 3 0) Accueil mathématique (5 0 0 0) Principe des réseaux (0 3 9 0) Théorie des graphes (0 6 0 0)

**Semestre 6** Analyse de Fourier (0 12 0 0) Géométrie Algorithmique (8,75 3 2 0) Introduction à XML (0 4,5 10,5 0) Programmation logique (0 0 3 0) Transmission de l'information (0 0 9 0)

**Semestre 7** Internet et Interopérabilité (10,5 1,5 7,5 4) Miniprojet systèmes à base de connaissances (0 0 9 0) Multimédia (0 0 9 0) Programmation logique (0 1,5 7,5 0) Projet innovant (0 0 0 9) Réseaux 1 (outils intranet) (0 0 18 0) Techniques de base en Intelligence Artificielle (0 3 0 0) Traitement d'image (0 0 3 0)

**Semestre 8** Introduction aux théories de la calculabilité et de la complexité (0 4,5 0 0) Miniprojet réseaux (0 0 10,5 0) Modèles aléatoires (0 3 2 0) Projet innovant (0 0 0 24) Réseaux 2 (0 0 2 0)

**Semestre 9** Internet protocol version 6 (0 0 6 0) Projet recherche (0 0 0 7,5) Sécurité des systèmes et réseaux (0 0 6 0)

**Semestre 10** Stage professionnel (0 0 0 2)

**Semestre 12** Codage Compression - Intelligence Artificielle : dans le cadre de l'École Doctorale STIM (8 0 0 0)

### 2.3 Responsabilités organisationnelles

**De 2003 à 2011** Responsable de l'option « Réseaux et Systèmes » de l'année terminale de la formation d'ingénieur « Systèmes Informatiques, Logiciels et Réseaux » dans le département Informatique de Polytech'Nantes. De 10 à 20 élèves ingénieurs par an, 150 heures d'enseignements spécifiques avec 15 intervenants industriels ou académiques

**Depuis 2011** Responsable des stages de fin d'étude de la formation ingénieur « Systèmes Informatiques : Logiciels et Réseaux ». 65 étudiants par an.

**Depuis 2008 (renouvelé en avril 2012)** Membre élu du conseil de Polytech Nantes, l'équivalent sans le statut juridique d'un conseil d'administration, et de la commission de choix.

## 3 Responsabilités scientifiques

**Depuis 2006** Responsable du thème géométrie discrète dans l'équipe IVC

**De 2001 à 2008** Membre élu de la commission de spécialistes section CNU 27 de l'université de Nantes jusqu'à sa dissolution à la mi-2008.

**Depuis 2009** Participation à des comités de sélection pour le recrutement de maîtres de conférences à Polytech (MCF 1909 en 2009, MCF 1842 en 2011) et au LIRIS, Lyon (MCF 522 2012).

**Septembre 2009** Participation au *review panel* – l'équivalent australien du comité de suivi de thèse – d'OLEG KUZNETSOV, School of Mathematical Sciences, Monash University.

**7 juin 2012** Organisation d'un meeting DGtal (logiciel libre des communautés de géométrie et topologie discrètes).

**8 juin 2012** Organisation de la journée annuelle du GT géométrie discrète du GDR Informatique Mathématique.

## 4 Activités en Recherche

Mes activités de recherche se situent dans le domaine de la géométrie discrète, d'une part autour de la tomographie discrète en particulier la transformation Mojette et la Finite Radon Transform (FRT) avec des applications en imagerie médicale, en réseau et stockage distribué. D'autre part, je développe des travaux autour de la morphologie mathématique et des distances discrètes, avec des développements théoriques et algorithmiques ainsi que des applications en analyse d'images médicales.

### 4.1 Encadrement doctoral

**Benoît Parrein** « *Description multiple de l'information par transformation Mojette,* »  
**10/1998–11/2001, taux 70%.**

Ce travail a porté sur l'utilisation de la transformation Mojette à des fins de protections des données sur les réseaux à perte de paquets en exploitant les propriétés de redondance de la Mojette. Production commune pendant la thèse : 4 communications nationales [113, 130, 131, 135], 4 conférences internationales [68, 69, 134, 138], et 1 revue internationale [70].

**Pierre Verbert** « *Sur la redondance des transformations Mojette en dimension  $n$  et en ligne,* »  
**10/1999–9/2004, taux 30%.**

Au cours de cette thèse, Pierre Verbert a exploré l'extension de la transformation Mojette à des dimensions supérieures à 2 et à des projections incomplètes. Production commune pendant la thèse : 2 conférences internationales [69, 138].

**Myriam Servières** « *Reconstruction tomographique Mojette,* »  
**10/2002–10/2005, taux 60%.**

L'objectif de cette thèse était de mettre à profit les propriétés de la transformation Mojette (géométrie, échantillonnage, structure des opérateurs direct et inverse) dans des algorithmes classiques et nouveaux de reconstruction tomographique. Production commune pendant la thèse : 9 conférences internationales [71, 114, 152, 154, 156, 157, 159, 160, 168].

**Aurore Arlicot** « *Distances discrètes et leurs applications,* »  
**depuis octobre 2008, taux 80%.**

L'objectif de ce sujet est de poursuivre les travaux entamés dans l'équipe autour des distances discrètes [105, 106] en particulier les distances à variation de poids local [99, 121]. Dans cette thèse qui se termine (soutenance prévue octobre 2012), Aurore s'est intéressée aux algorithmes de squelettisation d'images en un balayage, en collaboration avec le laboratoire LIOAD UMRS 791 dans le cadre du projet régional BIOREGOS. Production commune pendant la thèse : 6 conférences internationales [1, 2, 5, 6, 18, 118].

**Henri der Sarkissian** « *Reconstruction tomographique discrète du volume cardiaque en 3D en PET,* »  
**depuis janvier 2012, taux 50%.**

Cette thèse repart des travaux menés avec Myriam Servières pour les appliquer dans un contexte volumique en médecine nucléaire. À noter aussi le besoin d'utiliser une rotation 3D du volume cardiaque qui sera mise en œuvre avec nos outils de géométrie discrète. Cette thèse est co-financée par le projet Quanticardi (FUI) et la société Keosys.

J'ai également participé aux travaux de thèse d'Éloïse Denis et de Tarraf Torfeh en lien avec l'entreprise QualiFormeD entre 2006 et 2009. Production commune : 8 conférences internationales [13, 16, 40, 41, 43, 175–177], 2 communications nationales [14, 42] et un brevet international [39].

### Encadrement de masters de recherche

- 2001** ERDI BALINT (100%) : traitement d'images pour la biométrie  
**2003** PEGGY SUBIRATS (50%) : tomographie Mojette  
**2004** NICOLAS MERCIER (100%) : morphologie mathématique 3D  
**2005** PASCAL LESAGE (50%) : biométrie  
**2007** HADI FAYAD (50%) : tomographie  
**2008** ZHENWEI ZENG (50%) : réseau  
**2010** GUILLAUME DE CHARRETTE (100%), élève ingénieur, stage de recherche dans la School of Physics, Monash University, Melbourne : FRT  
**2011** YU ZHANG (100%), master SEGE en stage dans l'équipe IVC : morphologie mathématique [124] JEAN-DANIEL FISHER (50%, co-encadré par Benoît Parrein, 50%) : synchronisation, GIULIO D'IPPOLITO (50%, co-encadré par Benoît Parrein, 50%) stage dans l'équipe IVC : codage à effacement [132]  
**2012** DIMITRI PERTIN (50%, co-encadré par Benoît Parrein, 50%), élève ingénieur, stage de recherche dans l'équipe IVC : codage à effacement, BENJAMIN VILLAIN, élève ingénieur en stage de recherche à Melbourne University, encadré par Julien Ridoux [183].

### Encadrement de mémoires CNAM en lien avec des problématiques de recherche

- 2002** PHILIPPE COUTON : réseau, SIMON COLOMBIÉ : Mojette et réseau  
**2003** MICHEL LAISNEY : réseau, EMMANUEL CADIOT : réseau, PIERRE EVENOU : Mojette et réseau  
**2004** ANTHONY MOTTIER : imagerie médicale, EMMANUEL DE ROUGÉ : imagerie médicale  
**2006** ÉRIC HAMARD : réseau

Parallèlement, J'ai continué à travailler avec Pierre Evenou, ingénieur de recherche contractuel dans l'équipe IVC sur les thèmes de la tomographie discrète et des distances discrètes [1, 5, 6, 105, 116–119, 127].

Aurore et Pierre ont participé aux journées du groupe de travail géométrie discrète en 2008 (Université de Savoie) et 2010 (Université de Strasbourg). Pierre a participé à la conférence *Discrete Geometry for Computer Imagery* en 2008 à Lyon, Aurore en 2009 à Montréal et à IWCIA à Madrid en 2011. Aurore a été accueillie pendant un mois dans la School of Physics de l'Université Monash en 2010 pendant mon séjour.

## 4.2 Collaborations scientifiques

Lauréat d'une **bourse ARC International Fellowship** LX0989907 de l'ARC (équivalent australien de l'ANR). **Séjour de 12 mois dans la School of Physics, Monash University**, Melbourne, de juin 2009 à juillet 2010. Pendant ce séjour, 5 élèves de Polytech Nantes ont été accueillis en stage dans la school of Physics, j'ai pu faire venir Aurore Arlicot pendant un mois, et Benoît Parrein maître de conférences à Polytech a également été accueilli un mois.

Une participation régulière depuis 2003 à la conférence de référence dans le domaine de la géométrie discrète (*Discrete Geometry for Computer Imagery*) a permis de nouer des liens avec la communauté française et étrangère concrétisés par des publications communes et des accueils de chercheurs.

**Yves Amouriq** Faculté de Chirurgie Dentaire, Nantes, [1, 2, 5, 88]

**Olivier Déforges** IETR, et **Marie Babel**, IRISA, Rennes [8, 38]

**Jérôme Hulin** doctorant du LIS, Marseille, pendant une semaine en 2008

**Imants Svalbe** de l'université Monash à Melbourne, invité plusieurs fois pour un mois en 2008, 2009, 2011 et en 2012 [26, 52, 88, 119, 120, 169, 170]

**Andrew Kingston** Australian National University, Canberra, invité en 2009 pour un mois [26, 119, 120]

**Julien Ridoux** Melbourne University, invité un mois en 2011

**Shekhar Chandra** CSIRO, visite d'une semaine en avril 2012 [26]

**Robin Strand** Center for Image Analysis, Uppsala invité pour une semaine en 2008 (doctorant), pour un mois (associate Pr.) en juin 2012 [116–118, 167]

### 4.3 Brevets

Procédé sécurisé de fourniture de documents payants via un réseau de communication déposé avec la jeune pousse PiBi [80]. Ce brevet utilise la transformation Mojette dans un mécanisme de garantie de bonne fin pour la transmission de documents via un réseau de données.

Procédé de génération d'objets-test numériques avec la jeune pousse QualiFormeD. US Patent number 8160327 [39].

### 4.4 Expertises

**Octobre 2005** Reconnaissance biométrique faciale et vocale

Étude bibliographique sur la reconnaissance biométrique pour le compte de la société PiBi [103]. Participants : Florent Autrusseau (40%), Nicolas Normand (40%) et Jeanpierre Guédon (20%).

Durée : 12 jours.

**Octobre 2004** Expertise pour le compte de la société Courant Multimédia à Poitiers. État de l'art sur les technologies du codage et de la transmission vidéo, proposition d'un cahier des charges pour un système de transmission vidéo par courant porteur [87]. Participants : Nicolas Normand (50%), Vincent Ricordel (25%) et Patrick Lecallet (25%).

Durée : 2 mois.

**Juillet 2003** Analyse de la solution Trident Media Stop. Validation de l'approche proposée par la société *Trident Media Guard* pour la protection d'œuvres numériques sur les réseaux pair à pair [107]. Participant : Nicolas Normand.

Durée : 5 jours.

**QUALINAC** Plate-forme logicielle de gestion et d'analyse assistées par ordinateur de l'ensemble des contrôles de qualité et des maintenances réalisés sur les accélérateurs de particules à usage médical utilisés en radiothérapie.

Projet labellisé par le pôle de compétitivité « Images et Réseaux ».

Partenaires : QualiFormeD (porteur du projet), CRLCC PSHP et IRCCyN UMR 6597.

Participants IRCCyN : Nicolas Normand (responsable scientifique), Jeanpierre Guédon, Pierre Evenou. Durée : 24 mois.

**SEREADMO** Sécurisation des Réseaux Ad hoc et MOjette

Étude d'un nouveau protocole de routage sécurisé pour les réseaux ad hoc.

Projet ANR Programme ARA-SSIA « sécurité, systèmes embarqués et intelligence ambiante »

Partenaires : Thalès communications Cholet (porteur du projet), SIC (FRE CNRS 2731), KEOSYS SARL (Nantes) et IRCCyN (UMR 6597).

Autres participants : Jeanpierre Guédon, Benoît Parrein (responsable scientifique), Salima

Hamma, Eddy Cizeron, Pierre Evenou, Jiazi Yi.  
Durée : 36 mois.

#### 4.5 Collaborations avec l'environnement socio-économique

Dans ce paragraphe sont listées les entreprises (grands groupes comme jeunes pousses) avec lesquelles la collaboration s'est déroulée au travers de projets de toute nature (ANR, expertise, contrat).

**Grandes entreprises** La Poste (encadrement du mémoire CNAM de Michel Laisney), Alcatel Orvault (étude *Tag Switching* Christophe Boscher, Louis Le Gouriellec), Thalès (Éric Grall), Mitsubishi Electric ITE-TCL (Stéphane Morel)

**Entreprises innovantes** SERCEL (encadrement du mémoire CNAM de Philippe Couton), KeoSys (directeur : Jerome Fortineau) : encadrement du mémoire CNAM d'Anthony Mottier, du DEA de Pascal Lesage, Télénantes (directeur des programmes : Julien Kostrèche), Netagis (directeur : Patrick Julien) Lumiplan (Nicolas Tronchon)

**Aide au développement de jeunes pousses** PiBi (directeur : Olivier Jacquier), (encadrement du mémoire CNAM de Simon Colombier), TMG (Bastien Casalta), Mojette Labs devenue Fizians (directeur : Pierre Evenou), Qualidéo devenue AccepTV (directeur : Mathieu Carnec)

#### 4.6 Publications et production scientifique

La production correspond à un **h-index égal à 12** selon les données Google Scholar disponibles le 25 juin 2012

Reuves internationales	8 + 1 soumission en révision [116, 117]
Reuves nationales	2
Chapitres d'ouvrages internationaux	6
Chapitres en langue française	2
Conférences internationales	53 (dont 7 DGCI, 15 SPIE Medical Imaging) + 1 ISPCS [183] en soumission + 1 DGCI [115] et 1 ISMM [124] en préparation

#### Reuves internationales avec comité de lecture

- [117] **Nicolas Normand**, Robin Strand, Pierre Evenou, Aurore Arlicot, « Minimal-delay distance transform for neighborhood-sequence distances in 2D and 3D », *Computer Vision and Image Understanding*, 2012. DOI : 10.1016/j.cviu.2012.08.015, Accepté.
- [26] Shekhar Chandra, Imants Dzintars Svalbe, Jeanpierre Guédon, Andrew Maurice Kingston, **Nicolas Normand**, « Recovering missing slices of the discrete fourier transform using ghosts », *IEEE Transactions on Image Processing*, t. 21, n° 10, p. 4431–4441, oct. 2012. DOI : 10.1109/TIP.2012.2206033, 5-year IF : 4.205 (ISI 2010), ERA 17873 A\*.



- [167] Robin Strand, **Nicolas Normand**, « Distance transform computation for digital distance functions », *Theoretical Computer Science*, t. 448, p. 80–93, 24 août 2012. DOI : 10.1016/j.tcs.2012.05.010, 5-year IF : 1.087 (ISI 2010), ERA 18143 A.
- [38] Olivier Déforges, **Nicolas Normand**, Marie Babel, « Fast recursive grayscale morphology operators : from the algorithm to the pipeline architecture », *Journal of Real-Time Image Processing*, p. 1–10, juil. 2010. DOI : 10.1007/s11554-010-0171-8.
- [105] **Nicolas Normand**, Pierre Evenou, « Medial axis lookup table and test neighborhood computation for 3D chamfer norms », *Pattern Recognition*, t. 42, n° 10, p. 2288–2296, oct. 2009. DOI : 10.1016/j.patcog.2008.11.014, 5-year IF : 3.453 (ISI 2009), ERA 4503 A\*.
- [8] Marie Babel, Benoît Parrein, Olivier Déforges, **Nicolas Normand**, Jeanpierre Guédon, Véronique Coat, « Joint source-channel coding : secured and progressive transmission of compressed medical images on the Internet », *Computerized Medical Imaging and Graphics*, t. 32, n° 4, p. 258–269, juin 2008. DOI : 10.1016/j.compmedimag.2008.01.002, 5-year IF : 1.363 (ISI 2008), ERA 15962 B.
- [133] Benoît Parrein, **Nicolas Normand**, Jeanpierre Guédon, « Multimedia forward error correcting codes for wireless LAN », *Annals of telecommunications*, t. 58, n° 3-4, p. 448–463, mar. 2003.
- [70] Jeanpierre Guédon, Benoît Parrein, **Nicolas Normand**, « Internet distributed image information system », *Integrated Computer-Aided Engineering*, t. 8, n° 3, p. 205–214, août 2001.

**Article en cours de révision** *Image Processing On Line* est un journal en ligne qui traite de traitement et d’analyse d’images. Chaque publication contient un texte, un code source et une démonstration soumis à un comité de lecture.

- [116] **Nicolas Normand**, Robin Strand, Pierre Evenou, Aurore Arlicot, « A minimal delay neighborhood-sequence distance transform », *Image Processing On Line*, 2012, revision.

## Revue nationale avec comité de lecture

- [109] **Nicolas Normand**, Jeanpierre Guédon, « La transformée Mojette : une représentation redondante pour l’image », en français, *Comptes Rendus de l’Académie des Sciences - Série I - Mathématiques*, t. 326, n° 1, p. 123–126, jan. 1998. DOI : 10.1016/S0764-4442(97)82724-3, ERA 107 B.
- [123] **Nicolas Normand**, Christian Viard-Gaudin, « Lecture de codes-barres par vision 2D », en français, *Technique et Science Informatiques*, t. 13, n° 4, p. 493–523, 1994.

## Chapitres d’ouvrages internationaux

- [29] David Coeurjolly, **Nicolas Normand**, « Discrete geometry and projections », dans *The Mojette Transform : Theory and Applications*, J. GUÉDON, éd., Wiley-ISTE, 9 jan. 2009, chap. 1, p. 3–20, ISBN : 9781848210806.
- [62] Jeanpierre Guédon, **Nicolas Normand**, « Direct Mojette transform », dans *The Mojette Transform : Theory and Applications*, J. GUÉDON, éd., Wiley-ISTE, 9 jan. 2009, chap. 3, p. 41–66, ISBN : 9781848210806.

- [64] —, « Reconstructability with the inverse Mojette transform », dans *The Mojette Transform : Theory and Applications*, J. GUÉDON, éd., Wiley-ISTE, 9 jan. 2009, chap. 4, p. 67–83, ISBN : 9781848210806.
- [119] **Nicolas Normand**, Imants Dzintars Svalbe, Pierre Evenou, Andrew Maurice Kingston, « Inverse Mojette transform algorithms », dans *The Mojette Transform : Theory and Applications*, J. GUÉDON, éd., Wiley-ISTE, 9 jan. 2009, chap. 5, p. 85–113, ISBN : 9781848210806.
- [127] Benoît Parrein, Fadi Boulos, **Nicolas Normand**, Pierre Evenou, « Communication, networks and storage », dans *The Mojette Transform : Theory and Applications*, J. GUÉDON, éd., Wiley-ISTE, 9 jan. 2009, chap. 7, p. 151–182, ISBN : 9781848210806.
- [153] Myriam Servières, Jeanpierre Guédon, **Nicolas Normand**, Yves J. Bizais, « Mojette discrete tomography », dans *The Mojette Transform : Theory and Applications*, J. GUÉDON, éd., Wiley-ISTE, 9 jan. 2009, chap. 8, p. 183–214, ISBN : 9781848210806.

## Chapitres en langue française

- [32] Alain Daurat, **Nicolas Normand**, « Transformation et reconstruction par projections », en français, dans *Géométrie discrète et images numériques*, sér. traité IC2, D. COEURJOLLY, A. MONTANVERT et J.-M. CHASSERY, eds., Hermès, 2007, chap. 10, p. 239–251, ISBN : 978-2-7462-1643-3.
- [108] **Nicolas Normand**, Jeanpierre Guédon, « Applications de la transformation Mojette », en français, dans *Géométrie discrète et images numériques*, sér. traité IC2, D. COEURJOLLY, A. MONTANVERT et J.-M. CHASSERY, eds., Hermès, 2007, chap. 16, p. 337–347, ISBN : 978-2-7462-1643-3.

## Conférences internationales avec comité de lecture

- [124] **Nicolas Normand**, Yu Zhang, Imants Dzintars Svalbe, « Granulometry estimation from a neighborhood-sequence distance map », dans *International Symposium on Mathematical Morphology*, Uppsala, Sweden, mai 2013, En préparation.
- [115] **Nicolas Normand**, Robin Strand, Pierre Evenou, « Digital distances and integer sequences », dans *Discrete Geometry for Computer Imagery 2013*, sér. Lecture Notes in Computer Science, Sevilla, 20–22 mar. 2013, Accepté, ERA 43320 C.
- [183] Benjamin Villain, Matthew Davis, Julien Ridoux, Darryl Veitch, **Nicolas Normand**, « Probing the latencies of software timestamping », dans *International IEEE Symposium on Precision Clock Synchronization for Measurement, Control and Communication*, San Francisco, CA, USA, sept. 2012, p. 1–6. DOI : 10.1109/ISPCS.2012.6336623.
- [132] Benoît Parrein, **Nicolas Normand**, Majd Ghareeb, Giulio D’Ippolito, Federica Battisti, « Finite Radon coding for content delivery over hybrid client-server and P2P architecture », dans *International Symposium on Communications, Control, and Signal Processing*, mai 2012. DOI : 10.1109/ISCCSP.2012.6217794.
- [18] Pauline Bléry, Yves Amouriq, Jeanpierre Guédon, Paul Pilet, **Nicolas Normand**, Nicolas Durand, Florent Espitalier, Aurore Arlicot, Olivier Malard, Pierre Weiss, « Microarchitecture of irradiated bone : comparison with healthy bone », dans *SPIE Medical Imaging 2012 : Biomedical Applications in Molecular, Structural, and Functional Imaging*, t. 8317, San Diego, CA, USA : SPIE, fév. 2012, pages. DOI : 10.1117/12.911375.

- [6] Aurore Arlicot, Pierre Evenou, **Nicolas Normand**, « Single-scan skeletonization driven by a neighborhood-sequence distance », dans *Advances in Image Analysis and Applications*, R. P. BARNEVA, V. E. BRIMKOV, K. KOROUTCHEV et E. KORUTCHEVA, édés., Madrid, mai 2011, p. 61–72, ISBN : 978-981-08-7923-5, ERA 43761 C.
- [88] Chuanlin Liu, Jeanpierre Guédon, Imants Dzintars Svalbe, **Nicolas Normand**, Yves Amouriq, « Line Mojette ternary reconstructions and ghosts », dans *Advances in Image Analysis and Applications*, R. P. BARNEVA, V. E. BRIMKOV, K. KOROUTCHEV et E. KORUTCHEVA, édés., Madrid, mai 2011, 12 pages, ISBN : 978-981-08-7923-5, ERA 43761 C.
- [118] **Nicolas Normand**, Robin Strand, Pierre Evenou, Aurore Arlicot, « Path-based distance with varying weights and neighborhood sequences », dans *Discrete Geometry for Computer Imagery 2011*, I. DEBLED-RENNESON, É. DOMENJOUR, B. KERAUTRET et P. EVEN, édés., sér. Lecture Notes in Computer Science, t. 6607, Nancy, France : Springer Berlin / Heidelberg, avr. 2011, p. 199–210, ISBN : 978-3-642-19866-3. DOI : 10.1007/978-3-642-19867-0\_17, ERA 43320 C.
- [170] Imants Dzintars Svalbe, **Nicolas Normand**, « Properties of minimal ghosts », dans *Discrete Geometry for Computer Imagery 2011*, I. DEBLED-RENNESON, É. DOMENJOUR, B. KERAUTRET et P. EVEN, édés., sér. Lecture Notes in Computer Science, t. 6607, Nancy, France : Springer Berlin / Heidelberg, avr. 2011, p. 417–428, ISBN : 978-3-642-19866-3. DOI : 10.1007/978-3-642-19867-0\_35, ERA 43320 C.
- [2] Yves Amouriq, Jeanpierre Guédon, **Nicolas Normand**, Aurore Arlicot, Pierre Weiss, Yassine Benhdech, « Bone texture analysis on dental radiographic images : results with several angulated radiographs on the same region of interest », dans *SPIE Medical Imaging 2011 : Biomedical Applications in Molecular, Structural, and Functional Imaging*, t. 7965, Lake Buena Vista, Florida, USA : SPIE, fév. 2011, p. 796 525. DOI : 10.1117/12.878152.
- [169] Imants Dzintars Svalbe, Nikesh Nazareth, **Nicolas Normand**, Shekhar Chandra, « On constructing minimal ghosts », dans *Digital Image Computing : Techniques and Applications 2010*, Sydney, Australia, déc. 2010, p. 276–281. DOI : 10.1109/DICTA.2010.56, ERA 42717 B.
- [120] **Nicolas Normand**, Imants Dzintars Svalbe, Benoît Parrein, Andrew Maurice Kingston, « Erasure coding with the finite Radon transform », dans *IEEE Wireless Communications & Networking Conference*, Sydney, avr. 2010, p. 1–6. DOI : 10.1109/WCNC.2010.5506385, ERA 43061 B.
- [1] Yves Amouriq, Pierre Evenou, Aurore Arlicot, **Nicolas Normand**, Pierre Layrolle, Pierre Weiss, Jeanpierre Guédon, « Evaluation of trabecular bone patterns on dental radiographic images : influence of cortical bone », dans *SPIE Medical Imaging 2010 : Biomedical Applications in Molecular, Structural, and Functional Imaging*, t. 7626, San Diego, CA, USA : SPIE, fév. 2010. DOI : 10.1117/12.843964.
- [5] Aurore Arlicot, Yves Amouriq, Pierre Evenou, **Nicolas Normand**, Jeanpierre Guédon, « A single scan skeletonization algorithm : application to medical imaging of trabecular bone », dans *SPIE Medical Imaging 2010 : Image Processing*, t. 7623, San Diego, CA, USA : SPIE, fév. 2010. DOI : 10.1117/12.844250.
- [106] **Nicolas Normand**, Pierre Évenou, « Medial axis LUT computation for chamfer norms using H-polytopes », dans *Discrete Geometry for Computer Imagery 2008*, D. COEURJOLLY, I. SIVIGNON, L. TOUGNE et F. DUPONT, édés., sér. Lecture Notes in Computer Science, t. 4992, Lyon, France : Springer Berlin / Heidelberg, avr. 2008, p. 189–200. DOI : 10.1007/978-3-540-79126-3\_18, ERA 43320 C.

- [175] Tarraf Torfeh, Stéphane Beaumont, David Bonnet, Yves Barbotteau, Jeanpierre Guédon, **Nicolas Normand**, Éloïse Denis, Pascal Fenoglietto, Norbert Ailleres, « Digital phantoms for an evaluation of software used for an automatic analysis of the Winston-Lutz test in image guided radiation therapy », dans *SPIE Medical Imaging 2008 : Physics of Medical Imaging*, t. 6913, San Diego, CA, USA : SPIE, fév. 2008, 11 pages. DOI : 10.1117/12.768668.
- [41] Éloïse Denis, Stéphane Beaumont, Jeanpierre Guédon, Tarraf Torfeh, **Nicolas Normand**, Norbert Ailleres, « New automatic quality control methods for geometrical treatment planning system tools in external conformal radiotherapy », dans *SPIE Medical Imaging 2008 : Physics of Medical Imaging*, t. 6913, San Diego, CA, USA : SPIE, fév. 2008, 12 pages. DOI : 10.1117/12.768670.
- [52] H. Fayad, Jeanpierre Guédon, Imants Dzintars Svalbe, Yves J. Bizais, **Nicolas Normand**, « Applying Mojette discrete Radon transforms to classical tomographic data », dans *SPIE Medical Imaging 2008 : Physics of Medical Imaging*, t. 6913, SPIE, fév. 2008. DOI : 10.1117/12.770478.
- [13] Stéphane Beaumont, Tarraf Torfeh, Yves Barbotteau, Norbert Ailleres, Jeanpierre Guédon, **Nicolas Normand**, Sylvain Crespin, « Automatic image quality control of on-board imaging systems used for patient positioning in radiation therapy », dans *Dreiländertagung, Congress of the Swiss, Austrian and German societies of medical physics*, Bern, sept. 2007.
- [40] Éloïse Denis, Stéphane Beaumont, Jeanpierre Guédon, **Nicolas Normand**, Tarraf Torfeh, « Automatic quality control of digitally reconstructed radiograph computation and comparison with standard methods », dans *SPIE Medical Imaging 2007 : Physics of Medical Imaging*, t. 6510, San Diego, CA, USA, 17 fév. 2007. DOI : 10.1117/12.708193.
- [177] Tarraf Torfeh, Stéphane Beaumont, Jeanpierre Guédon, **Nicolas Normand**, Éloïse Denis, « Software tools dedicated for an automatic analysis of the CT scanner quality control images », dans *SPIE Medical Imaging 2007 : Physics of Medical Imaging*, t. 6510, San Diego, CA, USA, 17 fév. 2007, 13 pages. DOI : 10.1117/12.707343.
- [112] **Nicolas Normand**, Andrew Maurice Kingston, Pierre Évenou, « A geometry driven reconstruction algorithm for the Mojette transform », dans *Discrete Geometry for Computer Imagery 2006*, A. KUBA, L. G. NYÚL et K. PALÁGYI, éd., sér. Lecture Notes in Computer Science, t. 4245, Szeged, Hungary : Springer Berlin / Heidelberg, oct. 2006, p. 122–133. DOI : 10.1007/11907350\_11, ERA 43320 C.
- [43] Éloïse Denis, Jeanpierre Guédon, Stéphane Beaumont, **Nicolas Normand**, « Discrete and continuous description of a three dimensional scene for quality control of radiotherapy treatment planning systems », dans *SPIE Medical Imaging 2006 : Physics of Medical Imaging*, t. 6142, San Diego, CA, USA, fév. 2006, 12 pages. DOI : 10.1117/12.650571.
- [158] Myriam Servières, **Nicolas Normand**, Jeanpierre Guédon, « Interpolation method for the Mojette transform », dans *SPIE Medical Imaging 2006*, t. 6142, San Diego, CA, USA, fév. 2006. DOI : 10.1117/12.651009.
- [159] Myriam Servières, **Nicolas Normand**, Jeanpierre Guédon, Yves J. Bizais, « The Mojette transform : discrete angles for tomography », dans *Discrete Tomography and its Applications*, G. T. HERMAN et A. KUBA, éd., sér. Electronic Notes in Discrete Mathematics, t. 20, Elsevier, juil. 2005, p. 587–606. DOI : 10.1016/j.endm.2005.05.086.

- [114] **Nicolas Normand**, Myriam Servières, Jeanpierre Guédon, « How to obtain a lattice basis from a discrete projected space », dans *Discrete Geometry for Computer Imagery 2005*, É. ANDRES, G. DAMIAND et P. LIENHARDT, édés., sér. Lecture Notes in Computer Science, t. 3429, Poitiers, France : Springer Berlin / Heidelberg, avr. 2005, p. 153–160, ISBN : 978-3-540-25513-0. DOI : 10.1007/978-3-540-31965-8\_15, ERA 43320 C.
- [66] Jeanpierre Guédon, **Nicolas Normand**, « The Mojette transform : the first ten years », dans *Discrete Geometry for Computer Imagery 2005*, É. ANDRES, G. DAMIAND et P. LIENHARDT, édés., sér. Lecture Notes in Computer Science, t. 3429, Poitiers, France : Springer Berlin / Heidelberg, avr. 2005, p. 79–91, ISBN : 978-3-540-25513-0. DOI : 10.1007/978-3-540-31965-8\_8, ERA 43320 C.
- [154] Myriam Servières, Jérôme Idier, **Nicolas Normand**, Jeanpierre Guédon, « Conjugate gradient Mojette reconstruction », dans *SPIE Medical Imaging 2005*, SPIE, t. 5747, San Diego, CA, USA, fév. 2005, p. 2067–2074. DOI : 10.1117/12.593399.
- [156] Myriam Servières, **Nicolas Normand**, Yves J. Bizais, Jeanpierre Guédon, « Noise behavior for spline Mojette FBP reconstruction », dans *SPIE Medical Imaging 2005*, SPIE, t. 5747, San Diego, CA, USA, fév. 2005, p. 2100–2109. DOI : 10.1117/12.593390.
- [9] Marie Babel, Benoît Parrein, Olivier Déforges, **Nicolas Normand**, Jeanpierre Guédon, Joseph Ronsin, « Secured and progressive transmission of compressed images on the internet : application to telemedicine », dans *Color Imaging X : Processing, Hardcopy, and Applications*, G. G. ESCHBACH Reiner ; Marcu, éd., SPIE, t. 5670, San Jose, CA, jan. 2005, p. 126–136. DOI : 10.1117/12.589191.
- [71] Jeanpierre Guédon, Myriam Servières, Stéphane Beaumont, **Nicolas Normand**, « Medical software control quality using the 3D Mojette projector », dans *ISBI*, avr. 2004, p. 836–839. DOI : 10.1109/ISBI.2004.1398668.
- [160] Myriam Servières, **Nicolas Normand**, Peggy Subirats, Jeanpierre Guédon, « Some links between continuous and discrete Radon transform », dans *SPIE Medical Imaging 2004*, SPIE, t. 5370, fév. 2004, p. 1961–1971. DOI : 10.1117/12.533472.
- [168] Peggy Subirats, Myriam Servières, **Nicolas Normand**, Jeanpierre Guédon, « Angular assessment of the Mojette Filtered Back Projection », dans *SPIE Medical Imaging 2004*, A. A. AMINI et A. MANDUCA, édés., SPIE, t. 5370, fév. 2004, p. 1951–1960. DOI : 10.1117/12.533483.
- [157] Myriam Servières, **Nicolas Normand**, Jeanpierre Guédon, « A Mojette transform approach to discrete medical tomography », dans *First Malaysia-France Regional Workshop on Image Processing in Vision*, 2003.
- [99] **Nicolas Normand**, « Convex structuring element decomposition for single scan binary mathematical morphology », dans *Discrete Geometry for Computer Imagery 2003*, I. NYSTRÖM, G. SANNITI DI BAJA et S. SVENSSON, édés., sér. Lecture Notes in Computer Science, t. 2886, Naples, Italy : Springer Berlin / Heidelberg, nov. 2003, p. 154–163, ISBN : 3-540-20499-7. DOI : 10.1007/978-3-540-39966-7\_14, ERA 43320 C.
- [104] **Nicolas Normand**, Olivier Déforges, Patrick Le Callet, Aurélien Chiron, « Algorithme de transformation en distance généralisée en un balayage, mises en œuvre séquentielle et vectorielle », dans *GRETSI*, sept. 2003. DOI : 2042/13753.
- [152] Myriam Servières, Jeanpierre Guédon, **Nicolas Normand**, « A discrete tomography approach to PET reconstruction », dans *Fully 3D Reconstruction in Radiology and Nuclear Medicine*, Y. J. BIZAIS, éd., Saint Malo, juil. 2003.

- [129] Benoît Parrein, Jeanpierre Guédon, **Nicolas Normand**, « Multimedia packet transport : multiple layers or description ? », dans *Packet Video*, Nantes, avr. 2003.
- [65] Jeanpierre Guédon, **Nicolas Normand**, « Spline Mojette transform. applications in tomography and communications », dans *EUSIPCO 2002*, t. II, Toulouse, sept. 2002, p. 407–410.
- [128] Benoît Parrein, Lionel Chatelier, **Nicolas Normand**, Jeanpierre Guédon, « FEC comparison for multimedia wireless distribution », dans *ASWN 2002*, IEEE, Paris, juil. 2002, p. 140–144.
- [89] Franck Lusteau, Dominique Guillerme, Antoine Ciftsi, **Nicolas Normand**, Jeanpierre Guédon, « MUSHAN : multimedia ship area network », dans *Internet Multimedia Management Systems III*, SPIE, t. 4862, Boston, juil. 2002, p. 296–304. DOI : 10.1117/12.473045.
- [69] Jeanpierre Guédon, **Nicolas Normand**, Pierre Verbert, Benoît Parrein, Florent Autrusseau, « Load-balancing and scalable multimedia distribution using the Mojette transform », dans *ITCOM*, SPIE, t. 4519, Denver, août 2001, p. 226–234. DOI : 10.1117/12.434272.
- [138] Benoît Parrein, Pierre Verbert, **Nicolas Normand**, Jeanpierre Guédon, « Scalable multiple descriptions on packets networks via the n-dimensional Mojette transform », dans *ITCOM*, SPIE, t. 4524, Denver, août 2001, p. 243–252. DOI : 10.1117/12.434360.
- [134] Benoît Parrein, **Nicolas Normand**, Jeanpierre Guédon, « Multiple description coding using exact discrete Radon transform », dans *Data Compression Conference*, IEEE, Snowbird, mar. 2001, p. 508. DOI : 10.1109/DCC.2001.10031.
- [35] Olivier Déforges, **Nicolas Normand**, « A generic systolic processor for real time grayscale morphology », dans *International Conference on Acoustic, Speech and Signal Processing*, t. 6, juin 2000, p. 3331–3334. DOI : 10.1109/ICASSP.2000.860113.
- [68] Jeanpierre Guédon, **Nicolas Normand**, Benoît Parrein, Christophe Pouliquen, « Distributed image transmission and storage on internet system », dans *ACIDCA 2000*, Monastir, mar. 2000, p. 164–169.
- [48] Dietmar Eggeman, Jeanpierre Guédon, **Nicolas Normand**, « Transferts de documents multimédias sur réseaux à qualité de service », dans *CORESA*, Lannion, 1998.
- [37] Olivier Déforges, **Nicolas Normand**, « Recursive morphological operators for gray image processing. application in granulometry analysis », dans *ICIP*, t. 2, Santa Barbara, CA, USA, oct. 1997, p. 672–675. DOI : 10.1109/ICIP.1997.638585.
- [36] —, « A systolic preprocessor for online morphologic operations », dans *SPIE conf. on Parallel and Distributed Methods for Image Processing*, t. 3166, sept. 1997, p. 212–223. DOI : 10.1117/12.279619.
- [63] Jeanpierre Guédon, **Nicolas Normand**, « Mojette transform : applications for image analysis and coding », dans *Visual Communications and Image Processing*, t. 3024, fév. 1997, p. 873–884. DOI : 10.1117/12.263299.
- [110] **Nicolas Normand**, Jeanpierre Guédon, Olivier Philippé, Dominique Barba, « Controlled redundancy for image coding and high-speed transmission », dans *Visual Communications and Image Processing*, t. 2727, Orlando, FL, mar. 1996, p. 1070–1081. DOI : 10.1117/12.233180.
- [111] **Nicolas Normand**, Jeanpierre Guédon, Francois Terrien, Dominique Barba, « Source-channel coding for medical imaging », dans *PCS*, Melbourne, mar. 1996, p. 329–334.

- [121] **Nicolas Normand**, Christian Viard-Gaudin, « A background based adaptive page segmentation algorithm », dans *ICDAR*, Montréal, août 1995, p. 138–141. DOI : 10.1109/ICDAR.1995.598961.
- [122] —, « A two-dimensional bar code reader », dans *International Conference on Pattern Recognition*, IAPR, Jerusalem, oct. 1994, p. 201–203. DOI : 10.1109/ICPR.1994.577158.
- [182] Christian Viard-Gaudin, **Nicolas Normand**, Dominique Barba, « A bar code location algorithm using a two-dimensional approach », dans *ICDAR*, oct. 1993, p. 45–48. DOI : 10.1109/ICDAR.1993.395785.





# Chapitre 1

## Introduction

L'objectif de ce chapitre est de donner une vue globale de mes travaux de recherche en insistant sur les thèmes qui ne sont pas développés dans la suite du manuscrit. Le positionnement scientifique dans la communauté de la géométrie discrète est décrit dans la section 1.1. La section 1.2 résume les résultats obtenus et les approches adoptées. Les choix de rédaction sont expliqués dans la section 1.3.

### 1.1 Positionnement de ce travail

Mes thèmes de recherche, la tomographie discrète et les distances discrètes, sont introduits ici après une courte description du domaine plus général de la géométrie discrète.

#### 1.1.1 Géométrie discrète

Ce travail se situe dans le domaine de la géométrie discrète, à l'interface entre géométrie et informatique. À l'étranger, le domaine de recherche « Discrete Geometry » s'entend généralement comme la branche des mathématiques qui s'intéresse à l'étude des objets géométriques représentables par des structures discrètes : listes, arbres, graphes, treillis... Les polytopes sont par exemple des objets géométriques qu'on peut représenter de façon discrète (bien qu'ils soient continus).

En France, le domaine « Géométrie Discrète » prend un sens plus restreint, les chercheurs qui se reconnaissent dans cette communauté ont pour objectif la conception d'algorithmes capables de manipuler les objets géométriques, le plus souvent dans un contexte de traitement d'images. La conférence *Discrete Geometry for Computer Imagery* est la plus représentative de ce domaine de recherche et de cette communauté.

#### 1.1.2 Tomographie discrète

La tomographie est une technique d'imagerie dont l'objectif est de restituer une surface ou un volume à partir de mesures externes qui représentent ses projections dans un ensemble de directions. Elle est utilisée en imagerie médicale, en astrophysique, en contrôle non destructif, lorsque l'observation directe de l'objet d'étude est impossible (intrusive, inaccessible...). J. RADON a établi qu'il était possible de reconstruire une image à partir de ses projections, c'est-à-dire d'effectuer la transformation de J. RADON inverse, la transformée de J. RADON directe étant la

formulation des projections à partir de l'image  $f (\mathbb{R}^2 \rightarrow \mathbb{R})$  :

$$[Rf](t, \theta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \delta(t - x \cos \theta - y \sin \theta) dx dy \quad (1.1)$$

où  $\delta$  est la distribution de DIRAC et  $t - x \cos \theta - y \sin \theta = 0$  l'équation de la ligne intégrale de paramètres  $\theta$  et  $t$ .

Les approches classiques en tomographie discrète, s'attachent en général à reconstruire un ensemble de points, ce qui est équivalent à une image binaire, à partir d'un nombre très limité de projections, 2 ou 4 par exemple. En termes d'information, il est généralement nécessaire d'apporter de l'a priori par des contraintes géométriques ou topologiques sur les formes à reconstruire. Les algorithmes sont complexes (souvent NP-difficiles) et déterminer l'existence d'une solution ou son unicité est également difficile.

La transformation Mojette qui a été développée dans notre équipe à partir de 1995 est une version discrète et exacte de la transformation de Radon. Bien que selon PETER GRITZMANN lui-même, elle ait sa place à part entière dans la famille de la tomographie discrète, elle diffère d'autres approches par l'utilisation d'hypothèses moins restrictives sur les formes à reconstruire. Les algorithmes que nous avons développés ont une complexité polynomiale qui permet leur exploitation dans des applications avec de fortes contraintes de temps, par exemple pour le codage à effacement dans un système de fichiers distribué.

### 1.1.3 Distances discrètes

Mesurer des distances, par exemple l'écart entre des formes ou l'épaisseur des objets est très courant en analyse d'images. La plupart du temps, il ne suffit pas simplement d'estimer la distance entre deux points, le résultat attendu est une carte dense de mesures des distances depuis chaque point vers un ensemble : c'est la transformation en distance qui est fortement liée à la morphologie mathématique sur les ensembles binaires. Ces méthodes conduisent à de nombreuses applications : en analyse d'images mais aussi en codage [19, 141] et en synthèse [46, 76, 193].

Ce thème sera largement développé dans les chapitres 2 à 5.

## 1.2 Contributions personnelles

### 1.2.1 Éléments structurants à deux pixels

Pendant et depuis ma thèse de doctorat « Représentation d'images et distances discrètes basées sur les éléments structurants à deux pixels », mes thèmes de recherche de prédilection ont été la tomographie et les distances discrètes. Ce qui les lie, l'élément structurant à deux pixels, est relatif à la notion même d'angle discret, utilisé tantôt comme direction de projection, tantôt comme vecteur de déplacement dans un chemin [102]. Formellement, l'élément structurant à deux pixels correspondant au vecteur  $\vec{v}$  est le couple de points  $\{O, O + \vec{v}\}$ .

Le lien entre les distances et la morphologie mathématique sera explicité et exploité dans les chapitres 3 et 5. En particulier, nous verrons comment décomposer un élément structurant convexe pour ensuite effectuer l'opération d'érosion ensembliste très efficacement. Les opérations d'érosion et de dilatation ensemblistes d'une forme  $X$  par élément structurant à deux pixels  $E = \{O, O + \vec{v}\}$  se définissent simplement comme l'intersection ou la réunion de la forme  $X$  avec

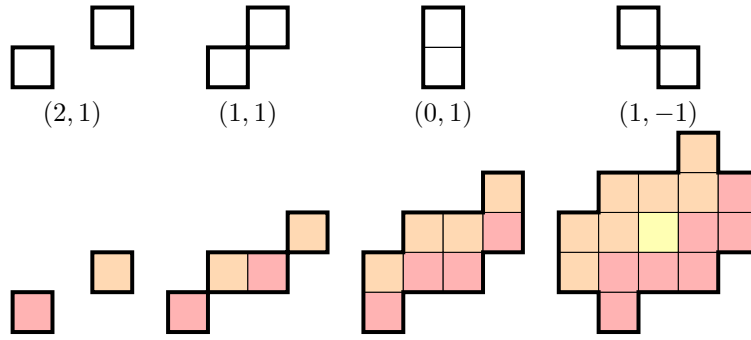


FIGURE 1.1 – Dilatation d’éléments structurants à deux pixels correspondant aux vecteurs  $(2, 1)$ ,  $(1, 1)$ ,  $(0, 1)$  et  $(-1, 1)$ .

une version translatée d’elle-même :

$$\begin{aligned} X \ominus \check{E} &= X \cap (X)_{-\vec{v}} \\ X \oplus \check{E} &= X \cup (X)_{-\vec{v}} \end{aligned}$$

où  $\oplus$  est la somme de MINKOWSKI (définition 2.41 page 48),  $(X)_p = \{x + p : x \in X\} = X \oplus \{p\}$  est le translaté de  $X$  par  $p$ , et  $\check{X} = -X = \{-x : x \in X\}$  est le *symétrique*, ou *transposé*, de  $X$  par rapport à l’origine.

### 1.2.2 Transformée Mojette

La transformation Mojette est une version discrète et exacte de la transformation de Radon. Une direction de projection est définie par un couple d’entiers  $(p, q)$  premiers entre eux. Chaque projection est composée de *bins*. Le bin de coordonnées  $b$  correspond à la droite discrète d’équation  $b + kq - lp = 0$  et sa valeur est la somme des valeurs des pixels sur cette droite :

$$[Mf](b, p, q) = \sum_{k=0}^{P-1} \sum_{l=0}^{Q-1} f(k, l) \Delta(b + kq - lp). \quad (1.2)$$

Ici,  $\Delta$  désigne le delta de KRONECKER :  $\Delta(0) = 1, \forall k \neq 0 \Delta(k) = 0$ , équivalent discret de la distribution de DIRAC, l’image discrète  $f$  est définie sur le domaine entier  $[0 \dots P-1] \times [0 \dots Q-1]$ .

La figure 1.2 montre un exemple de transformée Mojette pour l’ensemble de directions  $S = \{(2, 1); (1, 1); (0, 1); (-1, 1)\}$ . Les projections sont calculées modulo 4.

Le problème de la reconstruction consiste à séparer les différentes contributions qui s’additionnent dans chaque valeur de *bin*. Le caractère linéaire de la transformation rend possible sa résolution sous la forme d’un système  $Ax = y$  où  $x$  est le vecteur constitué des pixels de l’image,  $y$  contient les valeurs des *bins* des projections et  $A$  est la matrice de projection. Cependant la taille de ce système augmente rapidement avec celle de l’image. La solution n’est pas unique lorsque l’image peut recéler un fantôme, c’est-à-dire une forme dont les projections sont nulles pour les directions considérées [83]. Le critère de M. B. KATZ permet de déterminer si un ensemble de directions est suffisant pour une image rectangulaire, en fonction de sa taille et indépendamment de son contenu [83].

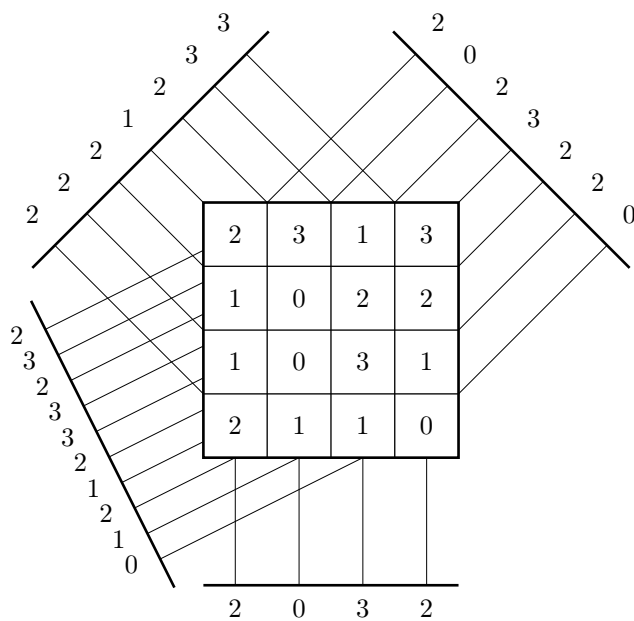


FIGURE 1.2 – Projection Mojette pour l’ensemble de directions  $S = \{(2, 1); (1, 1); (0, 1); (-1, 1)\}$ . Les projections sont calculées modulo 4.

**Algorithme de transformation Mojette inverse** Nous avons opté pour un algorithme simple (algorithme 1.3) qui, de manière répétée, identifie un *bin* reconstructible (repère 1) puis recopie sa valeur dans l’image. Un bin  $(p, q, b)$  est dit reconstructible lorsqu’il est aligné avec un seul pixel  $(k, l)$  de l’image, autrement dit lorsque  $(k, l)$  est le seul point dans le support de l’image tel que  $b = -qk + pl$  (repère 2). Après reconstruction du pixel, sa contribution est soustraite de toutes les projections (repère 3). Le problème à résoudre a été réduit d’un pixel, rendant souvent d’autres pixels reconstructibles. Avec JEANPIERRE GUÉDON, nous avons montré que si le problème admettait une solution unique, alors cet algorithme simple permettait de la trouver, la propriété de « reconstructibilité » se propageant alors de pixel en pixel jusqu’à leur épuisement [109]. La preuve de la validité de l’algorithme, géométrique, construit le plus petit fantôme pour cet ensemble de projections par dilatation des éléments structurants à deux pixels associés aux vecteurs de projection. De plus, elle généralise le critère de reconstruction à toute forme convexe. La figure 1.1 illustre un fantôme construit par dilatation d’éléments structurants à deux pixels.

**Complexité algorithmique et optimisation** L’algorithme 1.3 n’est cependant complet que lorsque l’on précise comment sont identifiés les bins reconstructibles et comment sont localisés les pixels correspondants. Une manière de répondre à la première question consiste à tenir le compte du nombre de points du support de l’image qui contribuent à chaque bin. Sont reconstructibles les bins pour lesquels ce nombre vaut 1. En pratique, il suffit de projeter la fonction indicative du support de l’image, c’est-à-dire une image unitaire de mêmes dimensions. Un stagiaire allemand, DIETMAR EGGEMAN, avait fait remarquer que la seconde question pouvait être résolue de la même manière en projetant les coordonnées des pixels. Lorsqu’un bin devient reconstructible pour le pixel  $(k, l)$ , les trois projections annexes,  $\text{proj}_1$ ,  $\text{proj}_k$  et  $\text{proj}_l$  recueillent respectivement les valeurs 1,  $k$  et  $l$  (cf. figure 1.4). Les opérations élémentaires de l’algorithme 1.3 ayant un coût constant, on en déduit que sa complexité temporelle est d’ordre  $O(IN)$  où  $I$  est le nombre de

---

**Algorithme 1.3:** Mojette inverse [109]

---

**Données :**  $S = \{(p_i, q_i), 1 \leq i \leq I\}$ , ensemble des directions de projection  
**Données :**  $\text{proj}(b, p_i, q_i)$ , projections  
**Données :**  $X$ , ensemble des points à reconstruire  
**Résultat :**  $f$ , image reconstruite

```
1 tant que  $\exists(p, q, b)$  restructurable faire
2   //  $(p, q, b)$  provient du pixel  $k, l : b = -qk + pl$ 
    $\exists!(k, l) \in X : b = -qk + pl$ 
    $f(k, l) \leftarrow P_{p, q}(b)$ 
    $X \leftarrow X \setminus \{(k, l)\}$ 
   pour chaque  $(p_i, q_i) \in S$  faire
3      $b_i \leftarrow -q_i k + p_i l$ 
      $P_{p_i, q_i}(b_i) \leftarrow P_{p_i, q_i}(b_i) - f(k, l)$ 
   fin
fin
```

---

projections et  $N = P \times Q$  le nombre de pixels de l'image de dimensions  $P \times Q$ .

En observant comment la restructurabilité se propage du bord de l'image vers l'intérieur, avec ANDREW KINGSTON et PIERRE EVENOU, nous avons proposé un algorithme qui évite le recours aux projections annexes. Le principe est d'anticiper le choix des bins restructurables et des pixels correspondants en calculant un chemin parmi les points de l'image. Le graphe présenté sur la figure 1.5 illustre la propagation de la reconstruction (en supposant une reconstruction de gauche à droite sur chaque ligne horizontale). Ce travail a été présenté lors de la conférence *Discrete Geometry for Computer Imagery 2006* [112].

**Reconstruction à partir de données bruitées** Pour fonctionner correctement, l'algorithme Mojette inverse doit disposer de données exactes, exemptes de bruit. Dans le cas contraire, les erreurs s'amplifient pendant la reconstruction. Avec la thèse de MYRIAM SERVIÈRES, nous avons exploré des approches utilisées en tomographie classique, plus adaptées aux données bruitées, telles que la rétroprojection filtrée, le gradient conjugué... Les propriétés de l'opérateur de rétroprojection  $M^*$  ont été étudiées. Nous avons établi que la composition  $M^*M$  a une structure Toeplitz bloc Toeplitz, exploitable dans les méthodes itératives [154, 159]. Lorsque le nombre de projection est suffisant pour que chaque couple de points apparaisse dans un des bins alors la rétroprojection en un pixel fait apparaître la somme des pixels de l'image (aisée à estimer) et un multiple de la valeur du pixel en question. Une formule exacte et peu sensible au bruit, semblable à une rétroprojection en découle :

$$f(k, l) = \frac{1}{I-1} \left[ \tilde{f}(k, l) - \frac{1}{I} \sum_i \sum_b \text{proj}(b, p_i, q_i) \right] \quad (1.3)$$

où la rétroprojection  $\tilde{f}$  en  $(k, l)$  s'écrit :

$$\tilde{f}(k, l) = \sum_{k=0}^{P-1} \sum_{l=0}^{Q-1} \text{proj}(b, p, q) \Delta(b + kq - lp) . \quad (1.4)$$

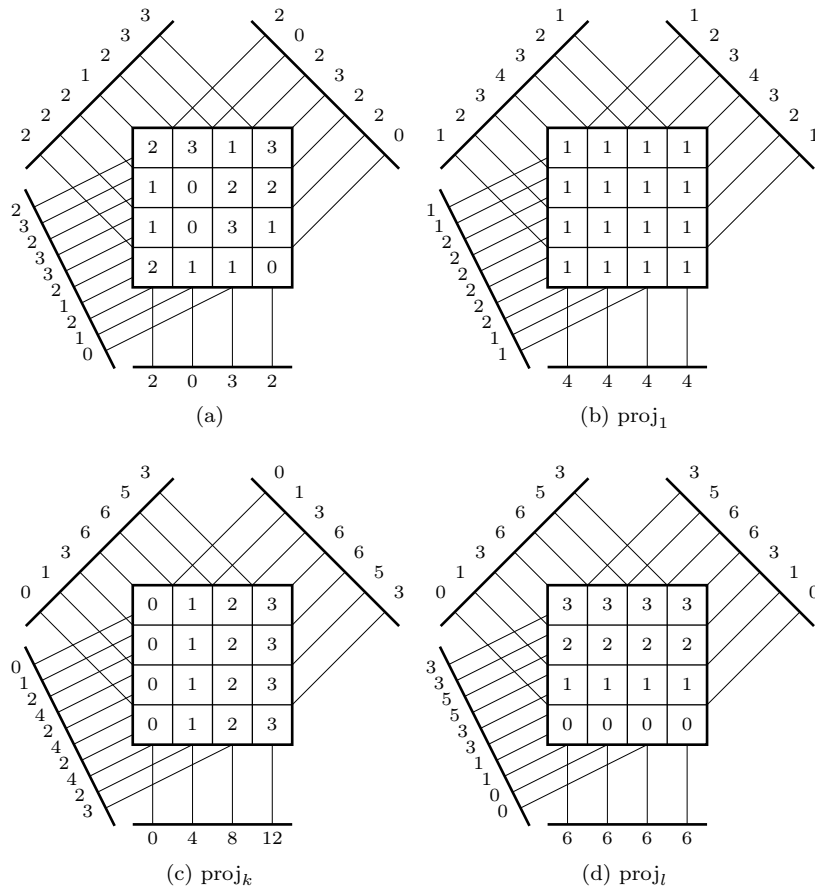


FIGURE 1.4 – Projection de l'image (a) et projections annexes utilisées dans l'algorithme de reconstruction Mojette pour le suivi du nombre de pixels (b) et des coordonnées des pixels projetés (c, d).

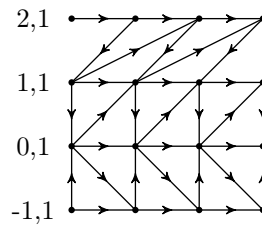


FIGURE 1.5 – Graphe de dépendance des points dans la reconstruction Mojette d'une image  $4 \times 4$ .

**Application : codage à effacement** L'application au codage à effacement est apparue assez tôt. Il suffisait en effet de remarquer qu'en calculant des projections supplémentaires et donc redondantes, au-delà d'un ensemble suffisant, la reconstruction était assurée à partir d'un sous-ensemble de directions [110]. Le nombre d'angles calculés et le nombre d'angles nécessaires constituant les paramètres d'un code capable de compenser une perte de données (code à effacement). Dans ce domaine, les codes REED-SOLOMON sont très populaires car ils sont optimaux en termes de quantité d'information, propriété appelée *Maximum Distance Separable* (MDS) [17]. Avec BENOÎT PARREIN et JEANPIERRE GUÉDON, nous avons montré que la transformée Mojette est un code  $(1 + \epsilon)$ -MDS, autrement dit un code qui tend asymptotiquement vers l'optimum quand la taille des données augmente. En combinant plusieurs paramètres de redondance dans un *buffer géométrique* unique, nous avons construit un schéma de codage à protection inégale (codage prioritaire) qui, couplé avec un codage source scalable, devient un système de codage à descriptions multiples [134, 135, 138]. Parallèlement, nous avons étudié l'application de la transformation Mojette au stockage distribué [68–70, 127, 129]. J'ai participé à l'encadrement du mémoire ingénieur CNAM de P. ÉVENOU qui a approfondi cette question. Depuis, P. ÉVENOU a fondé l'entreprise *Fizians* qui a développé la solution de stockage distribué *Rozo*. La transformation Mojette est au cœur de cette technologie en permettant de se prémunir contre les indisponibilités de serveurs (pannes matérielles, accès réseau rompu...) sans augmenter de manière déraisonnable le coût de stockage (contrairement aux solutions qui répliquent les données).

**Quantité d'information** Par rapport aux approches classiques en tomographie discrète, l'approche Mojette peut sembler moins efficace par le plus grand nombre de projections utilisées. Par exemple, l'ensemble des directions  $\{(1, 0); (1, 1); (1, -1); (3, 1); (-2, -1); (4, 1); (3, -1)\}$  suffisant pour reconstruire une image  $15 \times 15$  avec la Mojette produit 322 *bins*. En se contentant des quatre directions  $\{(1, 0); (0, 1); (1, 1); (1, -1)\}$  seuls 94 *bins* sont utilisés. En réalité dans cet exemple, les *bins* Mojette restent binaires et chacun peut avoir une représentation quatre fois plus petite : finalement, avec moins d'information la transformée Mojette permet de reconstruire de manière certaine, sans contrainte sur le contenu et avec une complexité algorithmique polynomiale.

**Comparaison avec l'approche algébrique de L. HAJDU et R. TIJDEMAN** Indépendamment des approches géométriques, L. HAJDU et R. TIJDEMAN ont proposé une approche algébrique de la tomographie discrète [74]. L'image  $f$  est représentée par sa fonction génératrice, c'est-à-dire le polynôme de LAURENT  $\sum f(k, l)x^k y^l$ . Dans ce formalisme, un produit de polynômes représente une convolution d'images et la projection dans une direction discrète correspond au reste d'une division de polynômes. Un fantôme simple dans la direction  $(a, b)$  a pour fonction génératrice :

$$f_{(a,b)}(x, y) = \begin{cases} x^a y^b - 1, & \text{if } a > 0, b > 0, \\ x^a - y^{-b}, & \text{if } a > 0, b < 0, \\ x - 1, & \text{if } a = 1, b = 0, \\ y - 1, & \text{if } a = 0, b = 1. \end{cases} \quad (1.5)$$

La figure 1.6 montre quelques fantômes et leurs fonctions génératrices, leur produit engendre la fonction génératrice du fantôme minimal pour l'ensemble des directions. Le résultat est bien sûr équivalent à celui obtenu par la convolution dans le domaine image des fantômes simples. Pour

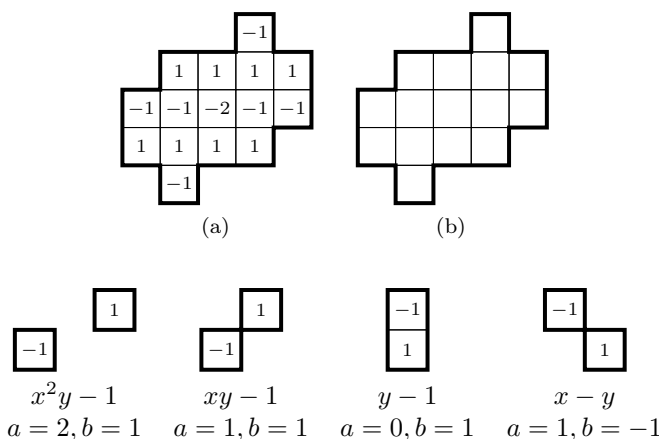


FIGURE 1.6 – Quelques fantômes simples et leurs fonctions génératrices.

l'ensemble  $S = \{(2, 1); (1, 1); (0, 1); (-1, 1)\}$ , la fonction génératrice du fantôme est :

$$\begin{aligned}
 F_S(x, y) &= (x^2y - 1)(xy - 1)(x - y)(y - 1) \\
 &= -x \\
 &\quad + y + xy + x^2y + x^3y \\
 &\quad - y^2 - xy^2 - 2x^2y^2 - x^3y^2 - x^4y^2 \\
 &\quad + xy^3 + x^2y^3 + x^3y^3 + x^4y^3 \\
 &\quad - x^3y^4.
 \end{aligned}$$

Récemment, L. HAJDU a décrit des ensembles de vecteurs [73] qui reconstruisent des images binaires à support rectangulaire dont la taille excède la limite de M. B. KATZ. Pour une taille  $15 \times 15$ , l'ensemble proposé est  $\{(1, 2), (1, 3), (1, 7), (1, -2)\}$ . Ces directions reconstruisent n'importe quel sous-ensemble du carré  $15 \times 15$  mais bien que ce dernier excède les limites de M. B. KATZ, le critère en termes d'information reste très favorable à la Mojette avec 322 bins binaires contre 274 bins multivalués. Ceci est dû au caractère  $(1 + \epsilon)$ -MDS de la Mojette. Dans ce cas précis, le nombre de bins est supérieur de 97 au nombre de pixels, ce surplus ne change pas lorsque la largeur de l'image augmente, et son importance relative décroît jusqu'à devenir négligeable.

### 1.2.3 FRT

J'ai rencontré IMANTS SVALBE pour la première fois à Naples, lors de la conférence *Discrete Geometry for Computer Imagery 2003*. IMANTS SVALBE a ensuite correspondu avec l'équipe IVC et JEANPIERRE GUÉDON l'a invité à passer un mois en à Nantes en marge de la conférence *Discrete Geometry for Computer Imagery 2005*. IMANTS SVALBE travaillait depuis 2000 sur une transformation proche de la Mojette, dénommée *Finite Radon Transform* (FRT). La différence principale réside dans la géométrie périodique de l'image  $p \times p$  avec des droites qui s'enroulent sur les bords de l'image (cf. figure 1.7).



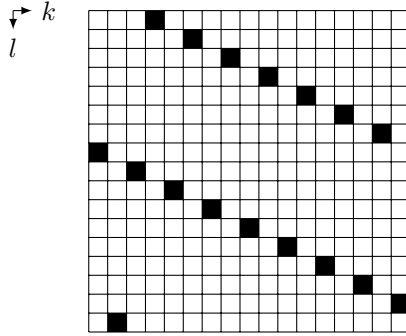


FIGURE 1.7 – Droite  $k - ml \equiv t \pmod{p}$  dans la géométrie FRT (ici  $m = 2$ ,  $t = 3$ ,  $p = 17$ ).

$$[\text{FRT}_p f](t, m) = \begin{cases} \sum_{k=0}^{p-1} f(k, t) & \text{si } m = p \\ \sum_{l=0}^{p-1} \sum_{k=0}^{p-1} f(k, l) \cdot \Delta(k - ml \equiv t \pmod{p}) & \text{sinon} \end{cases}$$

Les paramètres dans le domaine FRT sont  $m$  pour la direction de projection et  $t$  pour la position du bin.  $m$  représente le vecteur de projection  $(m, 1)$  pour  $0 \leq m < p$  et la projection horizontale est notée  $m = p$ .

Tout comme la Mojette, la FRT peut être reconstruite par une rétroprojection filtrée exacte. Une équation similaire à (1.3) était déjà connue en 1993 [93]. Lorsque  $p$  est premier, il existe exactement  $p + 1$  directions de droites distinctes.

### 1.2.4 Distances discrètes

Les distances discrètes auxquelles nous consacrerons les chapitres 2 à 5 sont intimement liées à la notion de chemin et correspondent à la manière la plus intuitive de les définir. Ce parti pris exclut d'emblée la distance euclidienne. Alors qu'elle est liée aux chemins (en ligne droite) dans le domaine continu, elle existe indépendamment de tout chemin dans le domaine discret. La distance euclidienne, bien qu'ayant de très bonnes propriétés d'invariance en rotation, correspond davantage à la discrétisation d'une distance continue qu'à une distance discrète à part entière.

Les algorithmes présentés sont rapides (complexité temporelle linéaire avec le nombre de pixels), peu consommateurs de mémoire, adaptés à des traitements à la volée. Plusieurs mises en œuvres matérielles capables de traiter les images à la cadence vidéo ont été développées et présentées [34–38, 104]. Pour compléter, la thèse d'AURORA ARLICOT, en cours de préparation, décrit une méthode de squelettisation à la volée.

### 1.2.5 Séjour en Australie

À l'invitation d'IMANTS SVALBE, j'ai passé un an dans la *School of Physics, Monash University*, Melbourne. Notre objectif était d'étudier la FRT et de développer un schéma de codage à effacement, à la manière de ce qui avait été fait avec la Mojette. Nous voulions d'abord mieux comprendre la transformation, ses propriétés, son espace nul c'est-à-dire les fantômes FRT.

**Effet dans le domaine FRT d'une transformation affine** J'ai commencé par m'intéresser à exprimer dans le domaine FRT l'effet d'une transformation affine dans le domaine image. Soit

la transformation affine :

$$\begin{bmatrix} k' \\ l' \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & k_0 \\ c & d & l_0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} k \\ l \\ 1 \end{bmatrix}$$

que l'on peut également écrire :

$$\begin{bmatrix} b_0 - a_0 & c_0 - a_0 & a_0 \\ b_1 - a_1 & c_1 - a_1 & a_1 \\ 0 & 0 & 1 \end{bmatrix}$$

Les points  $(0,0)$ ,  $(1,0)$  et  $(0,1)$  sont respectivement transformés en  $A : (a_0, a_1)$ ,  $B : (b_0, b_1)$  et  $C : (c_0, c_1)$ . Il existe exactement  $p^2(p^2 - 1)(p^2 - p)$  transformations affines inversibles  $(\text{mod } p)$ . Le point  $A$  peut être choisi librement parmi  $p \times p$  points. Le point  $B$  doit être distinct de  $A$  (un point exclu parmi  $p \times p$ ) et le point  $C$  ne doit pas être aligné avec  $A$  et  $B$  (ceci exclut  $p$  points). Une transformation affine préservant les alignements de points, elle a pour effet de permuter les coefficients dans le domaine FRT (les valeurs de la transformée sont déplacées mais inchangées). De plus, parce qu'elle préserve les parallèles, les lignes de coefficients FRT sont permutées et les coefficients sont permutés à l'intérieur des lignes mais les directions de projection ne se mélangent pas.

Prenons l'exemple des transformations de  $f$  et de sa transposée  $f^\top : f^\top(k, l) = f(l, k)$ . On note  $m' \equiv m^{-1} \pmod{p}$  l'inverse de  $m$  modulo  $p$  lorsqu'il existe, *i.e.* lorsque  $m \not\equiv 0 \pmod{p}$ .

$$\forall m \in [0, p[, [\text{FRT}_p f](t, m) = \sum_k \sum_l f(k, l) \cdot \Delta(k - ml - t \equiv 0 \pmod{p})$$

- Cas  $m \in ]0, p[ :$

$$k - ml \equiv t \pmod{p} \Leftrightarrow l - m'k + m't \equiv 0 \pmod{p} \text{ (en multipliant chaque terme par } -m').$$

$$\text{Donc } \Delta(k - ml - t \equiv 0 \pmod{p}) \Leftrightarrow \Delta(l - m'k + m't \equiv 0 \pmod{p}).$$

Et

$$[\text{FRT}_p f](t, m) = [\text{FRT}_p f^\top](-m't, m').$$

- Cas  $m \in \{0, p\} :$

$$\begin{aligned} [\text{FRT}_p f](t, 0) &= \sum_k \sum_l f(k, l) \cdot \Delta(k - t \equiv 0 \pmod{p}) \\ &= \sum_k f(t, k) = \sum_k f^\top(k, t) = [\text{FRT}_p f^\top](t, p). \end{aligned}$$

$$[\text{FRT}_p f](t, p) = \sum_k f(k, t) = \sum_k f^\top(t, k) = [\text{FRT}_p f^\top](t, 0).$$

Pour résumer, les directions 0 et  $p$  sont échangées, la direction  $m$  devient son inverse modulo  $p$ .

**Fantômes FRT** Les fantômes FRT élémentaires sont calculés de la même manière que les fantômes Mojette, par convolution des fantômes simples des directions de projection. Cependant, la périodicité de l'espace permet superpose parfois leurs valeurs pour les annuler. IMANTS SVALBE a qualifié de minimal un fantôme qui s'annule dans  $n$  directions tout en ne possédant que  $2n$  coefficients égaux à  $+1$  ou  $-1$  et les a étudiés sous l'angle combinatoire. Chaque coefficient doit être aligné dans les  $n$  directions avec un coefficient de signe opposé. Cette structure peut être représentée par un carré latin.

Après avoir noté que la transformation affine d'un fantôme préservait sa qualité de fantôme et que pour tout fantôme minimal il devait exister une transformation linéaire qui permutait ses

directions, j'ai proposé une méthode de génération de fantômes par application itérative d'une transformation linéaire des coordonnées de ses points. Nous avons ainsi pu dépasser la limite de 7 directions qui semblait infranchissable et générer des fantômes minimaux de très grande taille. La transformation linéaire utilise une matrice  $2 \times 2$  qui est une racine modulo  $p$  de la matrice identité ce qui rapproche étonnamment les fantômes minimaux de la transformation de FOURIER discrète qui utilise une racine complexe de l'unité et de la transformation de FOURIER modulaire qui utilise une racine entière de l'unité. Le fait que transformée de FOURIER soit une des méthodes, non décrite ici, de reconstruction de la FRT n'y est certainement pas étranger. IMANTS SVALBE a depuis découvert d'autres jolis liens, par exemple celui-ci : la transformée de FOURIER 2D d'un fantôme minimal est un empilement de fantômes minimaux.

**Approche algébrique : FRT et VANDERMONDE** Le but de ma mission en Australie était de mettre au point une méthode de codage à effacement utilisant la FRT, sur le principe de ce qui avait été déjà fait avec la Mojette. Bien que très voisines, les deux transformations étaient reconstruites par des algorithmes fort différents. Cependant, l'équation (1.3) ayant sont équivalent FRT était encourageant. Pendant mon séjour, j'ai proposé une simplification de la FRT, appelée FRT\*, qui supprime le cas particulier de la direction horizontale et rend les espaces image et FRT identiques de taille  $(p - 1) \times p$  (initialement  $p \times p$  pour l'image et  $(p + 1) \times p$  pour la FRT. J'ai montré que la FRT était alors un opérateur MDS et proposé deux schémas de codage à effacement, non-systématique et systématique (c'est-à-dire dont le code est composé du message en clair accompagné d'information de redondance) [120]. Sur cette base, ANDREW KINGSTON a proposé une forme involutive de la transformation (égale à sa transformation inverse) non publiée à ce jour. Chaque ligne de l'espace transformé par FRT\* est une combinaison linéaire de lignes d'images ayant subi une permutation circulaire. J'ai proposé une représentation algébrique, où chaque ligne est représentée par un polynôme de LAURENT et l'image entière est un vecteur de polynômes. Notre représentation algébrique est différente de celle de L. HAJDU et R. TIJDEMAN qui représente l'image par un polynôme unique et l'espace transformé par autant de polynômes qu'il y a de projections. Dans notre cas les lignes d'images sont distinguées ainsi que les lignes de la transformation. Il y a donc une symétrie dans la représentation de l'image et de sa transformée. La transformation complète (celle de l'image) peut alors être vue comme le produit d'une matrice de projection avec le vecteur constitué des lignes d'images. Nous avons découvert que la matrice de projection FRT est une matrice de VANDERMONDE. Il y a donc une très grande proximité entre cette transformation et les codes de REED-SOLOMON.

**Distances et morphologie mathématique** En parallèle de l'étude de la transformation FRT, j'ai continué à travailler sur les distances discrètes. Les algorithmes utilisés dans la thèse d'AURORE ARLICOT ont été étendus (initialement écrits pour la distance octogonale, ils ont été généralisés à toute séquence de voisinages). L'application de ces distances à l'analyse granulométrique d'images de matériaux a commencé à être examinée. Ce travail est présenté dans le chapitre 5 et une soumission à *International Symposium on Mathematical Morphology* est en préparation [124].

**Séquences d'entiers non décroissantes, séquences inverses, séquences complémentaires** La recherche sur les distances m'a souvent conduit à manipuler des séquences de nombres entiers. Je suis redevable à l'*On-Line Encyclopedia of Integer Sequences* (OEIS) de N.J.A. SLOANE et SIMON PLOUFFE de m'avoir permis, le plus souvent de flâner dans les séquences comme on navigue dans un dictionnaire de définition en définition, et quelques fois de gagner un temps considérable en révélant immédiatement formules, problèmes et références liées à une séquence. Mon séjour à Monash m'a permis de découvrir un article fondamental de J. LAMBEK et L. MOSER

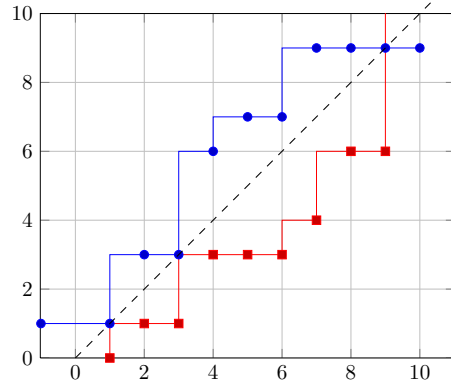


FIGURE 1.8 – Séquence non décroissante  $(0, 1, 1, 3, 3, 3, 4, 6, 6, \dots)$  et son inverse  $(1, 1, 3, 3, 6, 7, 7, 9, \dots)$

[86]. Simple, général, cet article est pourtant très peu cité. Il définit l'inverse d'une séquence d'entiers non décroissante  $f$  par :

$$f(m) < n \iff f^\dagger(n) \not\leq m . \quad (1.6)$$

La symétrie de l'expression (1.6) permet d'échanger  $f$  et  $f^\dagger$ .  $f^\dagger$  est également une séquence non décroissante et son inverse est  $f$ . La figure 1.8 montre une séquence et son inverse. De la même façon que les courbes représentatives d'une fonction et de sa réciproque sont symétriques par rapport à la première bissectrice, c'est aussi le cas des inverses de LAMBEK-MOSER. Sur la figure 1.9, sont tracées les droites d'équations  $k + l = n$  pour  $n$  entier. Chacune de ces droites rencontre exactement un point  $(k, f(k))$  ou  $(f^\dagger(k), k)$ , les séquences  $f(k) + k$  et  $f^\dagger(k) + k$  sont donc complémentaires. C'est en réalité la motivation principale de cet article, qui généralise par exemple des résultats précédents sur les suites de S. BEATTY [12]. Récemment, B. FARHI a obtenu, à partir de cette propriété, une formule close donnant le  $n^{\text{e}}$  entier naturel non-FIBONACCI [51].

Mais la portée de l'article de J. LAMBEK et L. MOSER est plus générale, il est probable que toute expression du type  $\max\{x : \text{condition sur } x\}$  puisse être réécrite en utilisant l'inverse de LAMBEK-MOSER. Sur le thème des distances discrètes, la plupart des résultats que j'ai produits peuvent être réinterprétés sous cet angle.

### 1.3 Choix de contenu et présentation

La Mojette et la FRT constituent une part importante de mon temps de comptage de petits carrés. Ce thème fort de l'équipe a déjà bénéficié d'une bonne exposition. Les premiers résultats, obtenus avec JEANPIERRE GUÉDON, ont été rendus publics dans son manuscrit d'Habilitation à Diriger des Recherches intitulé « Un tomographe discret : l'opérateur Mojette. Applications

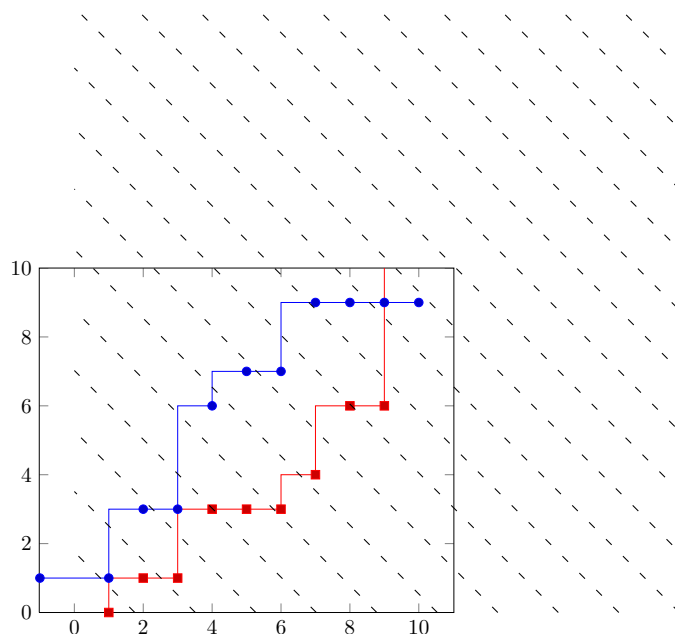


FIGURE 1.9 – Séquence non décroissante et son inverse

en image et réseaux ». En 2005, JEANPIERRE GUÉDON a donné un exposé invité lors de la conférence *Discrete Geometry for Computer Imagery* (DGCI). Un ouvrage collectif publié en 2009 a synthétisé les travaux de notre équipe et de nos collaborateurs (IMANTS SVALBE, ANDREW KINGSTON...). J'ai participé à la rédaction de six chapitres de cet ouvrage [29, 62, 64, 119, 127, 153]. J'ai également eu l'opportunité de présenter oralement les avancées plus récentes lors des journées du Groupe de Travail Géométrie Discrète à Clermont-Ferrand en octobre 2011 [101]. Notre équipe est un acteur important de la tomographie discrète et bénéficie à juste titre d'une visibilité internationale dans ce domaine. J'ai pris ma part de cet effort et continuerai à la prendre.

En parallèle, je me suis attaché à mener des travaux sur les distances discrètes. Ce document est l'occasion de montrer que notre équipe est également active dans ce domaine. À cette fin, je développe en quatre chapitres nos contributions principales sur les distances discrètes, établies depuis 2003. Cette date, correspond en effet à mon retour à la conférence DGCI (après une visite en tant que simple observateur en 1999) et à la rencontre décisive avec IMANTS SVALBE.

En gardant le lien entre les distances et les chemins, nous décrivons ainsi dans la suite du manuscrit, un procédé de construction itératif de disques similaire à la manière dont sont fabriqués les fantômes par dilatation d'éléments structurants à deux pixels (chapitre 2). Les algorithmes de transformation en distance que nous proposerons utiliseront les chemins pour propager les valeurs de distances (chapitre 3). Nous pourrions nous appuyer sur la construction des disques pour déterminer les règles d'inclusion et détecter les disques maximaux (chapitre 4). Dans le dernier chapitre, nous utiliserons les distances à des fins de mesure de la taille des objets dans les images.



## Chapitre 2

# Distances à poids locaux variables

### 2.1 Introduction

Bien qu'il ne soit pas formellement nécessaire de faire appel à la notion de chemin pour définir une distance (définition 2.1), le sens commun définit bien la distance entre deux points comme étant égale à la longueur du plus court chemin qui les sépare. Ce qui distingue alors une fonction distance d'une autre est constitué par la façon de construire ces chemins et la mesure de leur longueur.

Pour les distances discrètes de la grille carrée définies initialement par A. ROSENFELD et J. L. PFALTZ, un chemin connexe est construit par extension transitive d'une relation de voisinage entre les points, en remplacement de la notion de continuité du plan continu [148]. Le nombre de déplacements d'un point à l'un de ses voisins établit la mesure de sa longueur. Dans la grille carrée, deux types de voisinages naturels  $\mathcal{N}_4$  et  $\mathcal{N}_8$  conduisent à la définition de deux distances discrètes, les distances  $d_4$  et  $d_8$ . Ces deux distances usuelles ont le défaut d'une forte anisotropie,  $d_4$  surévaluant les distances diagonales,  $d_8$  les sous-évaluant en comparaison de la distance euclidienne.

Les généralisations classiques de ces distances combinent (au moins) les deux types de voisinage usuels de deux manières différentes et conduisent à deux classes de distances distinctes. Pour les unes, les distances à séquence de voisinages, plusieurs types de voisinage différents sont alternés (*e.g.* distance octogonale) [72, 186]. Pour les autres, les distances de chanfrein, différents voisinages sont utilisés simultanément avec des poids différents appelés distances locales [22]. Les distances à séquence de voisinages peuvent être vues comme des distances pour lesquelles le poids local d'un vecteur peut prendre la valeur 1 lorsque ce vecteur appartient au voisinage utilisé pour cette étape de la construction du chemin, ou une valeur arbitrairement élevée dans le cas contraire.

La proposition faite ici est de combiner ces deux approches avec des poids variables selon la distance parcourue sans restriction sur leurs valeurs (qui devra néanmoins être strictement positive).

Après un rappel de définitions et de propriétés des distances discrètes dans la section 2.2.1, la section 2.3 introduit cette nouvelle classe de distances à poids variables. Les propriétés des disques de ces distances sont discutées dans la section 2.4.

Les résultats présentés ici ont fait l'objet d'une communication lors de la conférence *Discrete Geometry for Computer Imagery 2011*, d'un article en révision dans *Computer Vision and Image Understanding* et d'une soumission en cours à *Image Processing On Line* [99, 116–118].

## 2.2 Préliminaires

### 2.2.1 Distances discrètes

**Définition 2.1** (Distance et norme). Soit  $d$  une fonction  $d : \mathbb{Z}^n \times \mathbb{Z}^n \mapsto \mathbb{N}$  et les propriétés suivantes  $\forall x, y, z \in \mathbb{Z}^n, \forall \lambda \in \mathbb{Z}$  :

$$\text{séparation} \quad d(x, y) = 0 \Leftrightarrow x = y, \quad (2.1a)$$

$$\text{symétrie} \quad d(x, y) = d(y, x), \quad (2.1b)$$

$$\text{inégalité triangulaire} \quad d(x, z) \leq d(x, y) + d(y, z), \quad (2.1c)$$

$$\text{invariance par translation} \quad d(x, y) = d(x + z, y + z), \quad (2.1d)$$

$$\text{homogénéité} \quad d(\lambda x, \lambda y) = |\lambda| \cdot d(x, y). \quad (2.1e)$$

$d$  est appelée *fonction distance discrète*, ou *distance discrète*, si elle possède les propriétés (2.1a) à (2.1c), (en anglais *metric*) et induit une *norme* si elle possède également les propriétés (2.1d) et (2.1e).

Dans la suite, nous abandonnerons la propriété (2.1b) voire (2.1c) pour définir des « pseudo-distances asymétriques ».

**Définition 2.2** (Disque). Soit  $d$  une fonction distance discrète, le *disque fermé*  $D_{\leq}$  et le *disque ouvert*  $D_{<}$  de centre  $c$  et de rayon  $r$  sont les ensembles de points de  $\mathbb{Z}^n$  tels que :

$$\begin{aligned} D_{<}(c, r) &= \{p : d(c, p) < r\}; \\ D_{\leq}(c, r) &= \{p : d(c, p) \leq r\}. \end{aligned} \quad (2.2)$$

Les disques sont généralement dénommés *boules* en dimension supérieure à 2. Nous utiliserons indifféremment l'un ou l'autre terme, sans changer la notation,  $D$ . Puisque le codomaine de  $d$  est  $\mathbb{N}$ , nous avons :

$$\forall r \in \mathbb{N}, d(c, p) \leq r \iff d(c, p) < r + 1.$$

Donc :

$$\forall r \in \mathbb{N}, D_{\leq}(c, r) = D_{<}(c, r + 1). \quad (2.3)$$

La propriété de séparation implique que le disque ouvert de rayon nul est vide  $D_{<}(c, 0) = \emptyset$  et le disque ouvert de rayon 1 ne contient que son centre  $D_{<}(c, 1) = \{c\}$ . De plus, la définition 2.2 implique que la suite de disques de  $d$  est croissante par inclusion ensembliste :

$$\forall r \in \mathbb{N}, D_{\leq}(c, r) \subseteq D_{\leq}(c, r + 1) \text{ and } D_{<}(c, r) \subseteq D_{<}(c, r + 1). \quad (2.4)$$

Cette propriété est très importante, car c'est elle qui donne un sens à la transformation en distance (définition 3.1) qui sera étendue à d'autres familles d'ensembles croissants par la suite.

### Distances engendrées par des chemins

Parmi les distances discrètes, les distances engendrées par des chemins occupent une place particulière. Historiquement, ce sont les premières distances discrètes introduites par A. ROSENFELD et J. L. PFALTZ conjointement à l'étude des images discrètes sous l'angle de la topologie [148] qui les rapproche des distances définies dans les structures de graphes. La notion de chemin fournit tout à la fois une manière de définir ces distances et une manière de les calculer par propagation des longueurs de chemins. Ces distances constituent un domaine de recherche à part entière qui a vu leur généralisation sous divers aspects, par la variété des définitions de chemin, différents façons de mesurer leurs longueurs et leurs applications à différents types de structures de  $\mathbb{Z}^2$  ou  $\mathbb{Z}^n$  [e.g. 21, 31, 95, 97, 165].



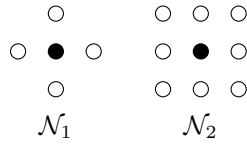


FIGURE 2.6 – 1- et 2-voisinages dans la grille carrée.

**Définition 2.3** (Distance engendrée par des chemins). Soit  $\mathcal{P}(p, q)$  l'ensemble des chemins de  $p \in \mathbb{Z}^n$  vers  $q \in \mathbb{Z}^n$ . La fonction  $d : \mathbb{Z}^n \times \mathbb{Z}^n \mapsto \mathbb{N}$  est une *distance engendrée par des chemins* si :

- (i)  $\forall (p, q), d(p, q) = \min \{ \mathcal{L}(P), P \in \mathcal{P}(p, q) \}$ ,
- (ii)  $d$  est une distance,

où  $\mathcal{L}(P)$  est la longueur (ou le coût) du chemin  $P$ . Un chemin  $P \in \mathcal{P}(p, q)$  de longueur  $d(p, q)$  est appelé chemin minimal, il n'est en général pas unique.

La définition précédente est incomplète, il reste à définir ce qu'est un chemin  $P \in \mathcal{P}(p, q)$  de  $p$  vers  $q$  et comment mesurer sa longueur,  $\mathcal{L}(P)$ . Différentes distances se distinguent par des choix de  $\mathcal{P}$  et  $\mathcal{P}$  distincts mais la réciproque est fautive : la même distance peut être définie avec plusieurs définitions de  $\mathcal{P}$  et  $\mathcal{L}$ .

**Définition 2.4** (points  $k$ -voisins). Dans une grille carrée, cubique ou hypercubique en dimension  $n$ , on appelle points  $k$ -voisins deux points dont les cellules partagent une face de dimension  $n - k$  ou supérieure, autrement dit, s'ils n'ont pas plus de  $k$  coordonnées qui diffèrent et si celles-ci ne diffèrent pas plus d'une unité.

Les points  $p$  et  $q$  sont  $k$ -voisins si et seulement si :

$$\sum_{i=1 \dots n} |p_i - q_i| \leq k \quad (2.5)$$

$$\max_{i=1 \dots n} \{ |p_i - q_i| \} \leq 1$$

Deux points  $k$ -voisins pour  $k$  positif sont également  $(k - 1)$ -voisins. Tout point est 0-voisin avec lui-même. On nomme points  $k$ -voisins stricts deux points  $k$ -voisins mais pas  $(k + 1)$ -voisins.

**Définition 2.5** (voisinage). Le *voisinage* d'un point  $p$ , noté  $\mathcal{N}(p)$  est l'ensemble des voisins de  $p$ .

Lorsque la relation de voisinage est invariante par translation (c'est le cas dans la suite), nous appellerons simplement *voisinage*, noté également  $\mathcal{N}$ , l'ensemble des vecteurs de déplacement qui ont un point pour origine et un de ses voisins pour extrémité.

Dans la grille carrée 2D, par exemple, les voisinages usuels sont le 1-voisinage pour lequel chaque point possède quatre voisins, dits voisins directs, situés au-dessus, au-dessous, à gauche et à droite (une arête commune) et le 2-voisinage qui compte quatre points supplémentaires situés en diagonale, dits voisins indirects dont les pixels partagent un sommet (figure 2.6). Ces voisinages sont appelés 4- et 8-voisins par certains auteurs en référence au nombre de leurs éléments. Dans la grille cubique 3D, les 1-, 2- et 3-voisins sont souvent dénommés respectivement 6-, 18- et 26-voisins pour la même raison.

### Distances basées sur un voisinage unique

A. ROSENFELD et J. L. PFALTZ ont introduit en 1966 les distances  $d_4$  et  $d_8$ , dénotées  $d$  et  $d^*$  dans [148], basées respectivement sur les 1- et 2-voisinages. Pour ces distances, un chemin

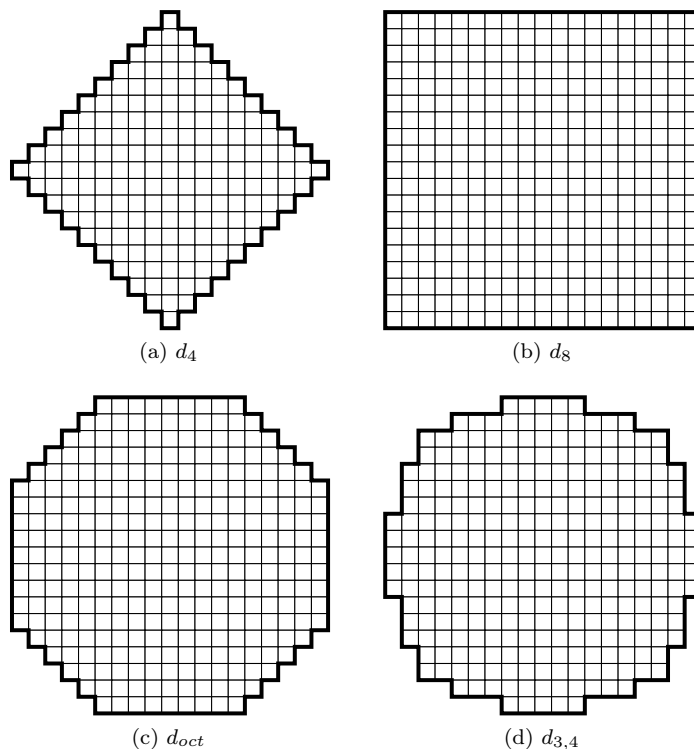


FIGURE 2.7 – Disques de quelques distances discrètes.  $d_1$  et  $d_2$ .

est une séquence de points voisins deux à deux :  $(p_0, \dots, p_r)$  où  $p_{i+1} = p_i + \vec{v}$ ,  $\vec{v} \in \mathcal{N}$ ,  $\mathcal{N}$  étant  $\mathcal{N}_1$  pour  $d_4$  et  $\mathcal{N}_2$  pour  $d_8$ . Le coût associé à un chemin est égal au nombre des déplacements :  $\mathcal{L}(p_0, \dots, p_r) = r$  [148]. Les formes fonctionnelles de ces fonctions distance sont [139, 147] :

$$\forall (k, l) \in \mathbb{Z}^2 \times \mathbb{Z}^2 \begin{cases} d_4(p, q) = |p_1 - q_1| + |p_2 - q_2| \\ d_8(p, q) = \max \{|p_1 - q_1|, |p_2 - q_2|\} \end{cases}$$

$d_4(p, q)$  et  $d_8(p, q)$  coïncident lorsque  $p$  et  $q$  sont alignés horizontalement et verticalement et sont dans ce cas égales à la distance euclidienne. Dans le cas général,  $d_4(p, q) \geq d_8(p, q)$ ,  $d_4$  surestime et  $d_8$  sous-estime les chemins diagonaux, comparativement à la distance euclidienne.

Les figures 2.7a et 2.7b montrent les disques ouverts de rayon 10 de ces distances.

Le besoin de distances moins dépendantes des rotations a conduit à imaginer de nouvelles distances qui combinent les voisinages  $\mathcal{N}_1$  et  $\mathcal{N}_2$  voire en introduisent de nouveaux, soit en modifiant la définition de chemin, soit en mesurant différemment leurs longueurs.

### Distances à séquence de voisinages

L'idée de changer de voisinage en fonction de la distance parcourue est apparue dès 1968 avec la distance octogonale  $d_{oct}$  (notée  $d_4$  dans [147]). Un chemin de  $d_{oct}$  est une séquence de points  $(p_0, \dots, p_r)$  dans laquelle les points successifs  $p_{i-1}, p_i$  sont 1-voisins si  $i$  est impair, 2-voisins sinon.

La forme fonctionnelle est [147] :

$$d_{oct}(p, q) = \max \left\{ \left\lfloor \frac{2}{3} (|p_1 - q_1| + |p_2 - q_2| + 1) \right\rfloor, d_8(p, q) \right\} \quad (2.6)$$

Plus généralement, un  $B$ -chemin où  $B$  est une séquence  $B = (b(1), b(2), \dots)$  à valeurs dans  $[1, n]$ , est une séquence de points  $(p_0, \dots, p_l)$  où  $p_{i-1}, p_i$  sont  $b(i)$ -voisins, *i.e.*,  $(p_i \in \mathcal{N}_{B(i)}(p_{i-1}))$  [31]. On parle alors de distance à séquence de voisinages.

Toutes les séquences, cependant, ne donnent pas des distances. M. YAMASHITA et N. HONDA donnent une condition de “métricité” pour les séquences périodiques de voisinages dans  $\mathbb{Z}^n$  (séquences périodiques) [188].

### Distances de chanfrein

Les distances de chanfrein utilisent simultanément plusieurs types de voisinages en leur attribuant des poids, ou distances locales. Des précurseurs de ces distances ont été introduits par U. MONTANARI [95]. H. G. BARROW, J. M. TENENBAUM, R. C. BOLLES et H. C. WOLF présentent implicitement la première distance de chanfrein  $d_{2,3}$  en décrivant l’algorithme qui les calcule H. G. BARROW, J. M. TENENBAUM, R. C. BOLLES et H. C. WOLF. Mais elles ont réellement été popularisées par G. BORGEFORS qui recommande  $d_{3,4}$  et  $d_{5,7,11}$  [22]. La géométrie des boules de chanfrein a été étudiée en profondeur par É. THIEL et É. RÉMY [144, 172, 173].

**Définition 2.8** (pondération [172]). On appelle *pondération*, notée  $M = (\vec{v}; w)$ , l’association d’un vecteur  $\vec{v}$  et d’un poids  $w$ .

Le poids  $w$ , strictement positif, représente le coût (ou distance locale) du déplacement de vecteur  $\vec{v}$ .

**Définition 2.9** (masque de chanfrein [172]). Un *masque de chanfrein* est un ensemble de pondérations qui possède une symétrie centrale et contient au moins une base de  $\mathbb{Z}^n$ .

$$\mathcal{M} = \{M_i \in \mathbb{Z}^n \times \mathbb{N}^*\}_{1 \leq i \leq m} \quad (2.7)$$

La grille  $\mathbb{Z}^n$  est symétrique relativement aux hyperplans normaux aux axes et aux bissectrices (G-symétrie). L’espace  $\mathbb{Z}^n$  est divisé en  $2^n \cdot n!$  sous-espaces (il y en a 8 dans  $\mathbb{Z}^2$  et 48 dans  $\mathbb{Z}^3$ ). Le sous-ensemble particulier  $x_n \geq \dots \geq x_1 \geq 0$  est appelé *cône générateur* ou simplement *générateur* noté  $\mathcal{G}$ . Pour chaque point de  $\mathbb{Z}^n$ , on peut déterminer son point G-symétrique unique dans  $\mathcal{G}$  en ordonnant les valeurs absolues de ses composantes par ordre décroissant.

Inversement, on dérive les points G-symétriques de  $p \in \mathcal{G}$  en avec toutes les  $n_1!$  permutations possibles des composantes de  $p$  et toutes les  $2^{n_2}$  combinaisons de de signes où  $n_1$  et  $n_2$  sont respectivement le nombre de valeurs absolues différents des composantes de  $p$  et le nombre de composantes non nulles.

Les masques de chanfrein étant généralement G-symétriques (C. FOUARD et G. MALANDAIN donnent des contre-exemples pour des grilles anisotropes [54]), les pondérations sont données uniquement dans  $\mathcal{G}$  et on note :

$$\mathcal{M}_{\mathcal{G}} = \{M_i \in \mathcal{G} \times \mathbb{N}^*\}_{1 \leq i \leq m} .$$

Les vecteurs dans  $\mathcal{G}$  sont usuellement ordonnés dans l’ordre lexicographique de leurs composantes :  $p$  précède  $q$  si  $\exists i \geq 0, \forall j < i, p_j < q_j$  et  $p_i = q_i$ . Selon cet ordre, les premiers points dans le cône générateur 2D sont :  $(0, 0), (1, 0), (1, 1), (2, 0), (2, 1), (2, 2) \dots$ . Chaque vecteur est nommé par une lettre de l’alphabet latin, en excluant le déplacement nul interdit dans les masques de

chanfrein, et les vecteurs multiples qui sont généralement superflus. Dans la grille 2D :  $\vec{a} = (1, 0)$ ,  $\vec{b} = (1, 1)$ ,  $\vec{c} = (2, 1)$ ,  $\vec{2c} = (4, 2)$ , et les poids correspondants sont désignés par  $a, b, c, 2c...$  La méthode de construction de ces points, par les séquences d'HAROS-FAREY, a été décrite dès le début du XIX<sup>e</sup> siècle [50, 75].

**Définition 2.10** (distance de chanfrein [172]). Soit le masque de chanfrein  $\mathcal{M} = \{(\vec{v}_i, w_i) \in \mathbb{Z}^n \times \mathbb{N}^*\}_{1 \leq i \leq m}$ . La distance de chanfrein (ou distance pondérée) qui sépare deux points  $x$  et  $y$  est :

$$d(x, y) = \min \left\{ \sum \lambda_i w_i : x + \sum \lambda_i \vec{v}_i = y, \lambda_i \in \mathbb{N}, 1 \leq i \leq m \right\} . \quad (2.8)$$

Les distances de chanfrein ont de fortes similitudes avec les distances  $k$ -orientées [45, p. 180].

Les chemins entre les points  $x$  et  $y$  sont produits par une séquence de déplacements selon les vecteurs du masque  $\mathcal{M}$ . Le coût total (ou longueur) d'un chemin est la somme des poids associés aux déplacements et la distance entre  $x$  et  $y$  est la longueur du plus court chemin entre  $x$  et  $y$ .

Tout masque de chanfrein définit une distance (en anglais : *metric*) [180]. En effet, le masque contenant une base de  $\mathbb{Z}^n$ , tout point peut être atteint par une combinaison des vecteurs de cette base, ou de leurs symétriques également présents dans le masque. Il existe donc un chemin entre tout couple de points de  $\mathbb{Z}^n$ . Les poids étant strictement positifs, toute distance est positive excepté pour un chemin nul, de longueur nulle, d'un point  $x$  quelconque à lui-même. Considérons trois points  $x, y$  et  $z$ . Il existe au moins un chemin de longueur  $d(x, y)$  de  $x$  à  $y$  et au moins un chemin de longueur  $d(y, z)$  de  $y$  à  $z$ . La concaténation de ces deux chemins forme un chemin valide de  $x$  à  $z$  de longueur  $d(x, y) + d(y, z)$ . Puisque  $d(x, z)$  est la longueur du plus court chemin de  $x$  à  $z$ ,  $d(x, z)$  ne peut excéder  $d(x, y) + d(y, z)$  et l'inégalité triangulaire est vérifiée.

Cependant un masque de chanfrein n'engendre une norme que lorsque des conditions particulières sur les vecteurs du masque et leurs poids associés permet un triangulation de la boule rationnelle en cônes d'influence [144, 172]. Lorsqu'un masque engendre une norme, alors toutes ses boules sont convexes.

## 2.2.2 Séquences non décroissantes d'entiers

### Séquences d'entiers définies sur $\mathbb{N}^*$

Soit  $f$  une fonction de  $\mathbb{N}^*$  dans  $\mathbb{N} \cup \{\infty\}$  qui définit la séquence d'entiers  $(f(1), f(2), \dots)$ . Par abus de langage, nous appellerons également séquence, la fonction  $f$  elle-même :

$$\begin{aligned} f : \mathbb{N}^* &\rightarrow \mathbb{N} \cup \{\infty\} \\ m &\mapsto f(m) \\ &\forall m > 0, f(m) \leq f(m+1) . \end{aligned} \quad (2.9)$$

**Définition 2.11** (séquence complémentaire). Deux séquences sont dites *complémentaires* si et seulement si chaque entier naturel apparaît une seule fois dans une séquence et est absent de l'autre.

### Séquences de BEATTY

**Définition 2.12** (séquence de BEATTY). On appelle *séquence de BEATTY*, la séquence  $(f(1), f(2), \dots)$  engendrée par la fonction  $f(n) = \lfloor n\alpha \rfloor$  où  $\alpha > 1$  est un irrationnel.

S. BEATTY a observé que si  $X$  est un irrationnel positif et  $X \times Y = 1$  alors les deux séquences d'entiers définies par :

$$f(m) = \lfloor m(X + 1) \rfloor , \quad (2.10)$$

$$g(n) = \lfloor n(Y + 1) \rfloor , \quad (2.11)$$

$n$	1	2	3	4	5	6	7	8	9	10	11	12	13	14
$\lfloor \sqrt{2}n \rfloor$	1	2	4	5	7	8	9	11	12	14	15	16	18	19
$\lfloor \frac{\sqrt{2}}{\sqrt{2}-1}n \rfloor$	3	6	10	13	17	20	23	27	30	34	37	40	44	47

Tableau 2.13 – Deux séquences de BEATTY complémentaires.

sont complémentaires [12]. Plusieurs preuves ont été données par A. OSTROWSKI, J. HYSLOP et A. C. AITKEN [126]. Le tableau 2.13 donne un exemple de paire de séquences de BEATTY complémentaires.

### Inverse de LAMBEK-MOSER

**Définition 2.14** (inverse de LAMBEK-MOSER). Soit  $f$  une fonction de  $\mathbb{N}^*$  dans  $\mathbb{N} \cup \{\infty\}$  qui définit une séquence d'entiers  $(f(1), f(2), \dots)$  non décroissante (autrement dit, croissante au sens large) :  $\forall n \in \mathbb{N}^*, f(n+1) \geq f(n)$ .  $f$  admet une séquence inverse unique au sens de J. LAMBEK et L. MOSER, notée  $f^\dagger$  et définie par [86] :

$$f(m) < n \iff f^\dagger(n) \not\leq m . \quad (2.12)$$

La symétrie de l'expression (2.12) permet d'échanger  $f$  et  $f^\dagger$ .  $f^\dagger$  est également une séquence non décroissante et son inverse est  $f$ . Le tableau 2.15 donne l'exemple de deux séquences inverses l'une de l'autre.

La quantité  $f^\dagger(n)$  peut être interprétée comme le nombre de valeurs de la séquence  $f$  inférieures à  $n$  :

$$f^\dagger(n) = |\{m : f(m) < n\}| ,$$

le rang de la dernière valeur dans la séquence  $f$  inférieure à  $n$  :

$$= \max \{m : f(m) < n\} \text{ ou } 0 \text{ si } f(1) \geq n \text{ ou } \infty \text{ si } \forall m, f(m) < n , \quad (2.13)$$

le rang qui précède la première valeur supérieure ou égale à  $n$  dans la séquence  $f$  :

$$= \min \{m : f(m) \geq n\} - 1 \text{ ou } \infty \text{ si } \forall m, f(m) < n . \quad (2.14)$$

Une propriété notable de la séquence  $f$  et de son inverse  $f^\dagger$  est que, en ajoutant le rang de chaque terme à celui-ci, on obtient deux séquences complémentaires  $f(m) + m$  et  $f^\dagger(n) + n$  [86]. Ce résultat est plus général que celui donné par OSTROWSKI, HYSLOP et AITKEN [126] sur les séquences de BEATTY [12].

On déduit de (2.12) que l'inverse de la séquence  $f(m) = \lfloor \tau m \rfloor$  où  $\tau$  est un réel, est  $f^\dagger(n) = \lceil \frac{n}{\tau} - 1 \rceil$  donc  $f(m) + m = \lfloor (1 + \tau)m \rfloor$  et  $f^\dagger(n) + n = \lceil (1 + \frac{1}{\tau})n - 1 \rceil$  sont deux séquences complémentaires. Si  $\tau$  est irrationnel, ces séquences sont des séquences de BEATTY et, quel que soit  $n$  un entier positif,  $\lceil (1 + \frac{1}{\tau})n - 1 \rceil$  est égal à  $\lfloor (1 + \frac{1}{\tau})n \rfloor$  conformément à [12].

**Itérer une séquence non décroissante**  $f^\dagger(n)$  étant l'indice du plus grand terme de  $f$  inférieur à  $n$  (ou 0),  $f(f^\dagger(n))$  est donc la plus grande valeur inférieure à  $n$  dans la séquence  $f$  (si  $f^\dagger(n) > 0$ ).

De la même façon,  $f(f^\dagger(n) + 1)$  est la plus petite valeur de  $f$  supérieure ou égale à  $n$ .

$n$	1	2	3	4	5	6	7	8	9	10	11	12	13	14
$f(n)$	1	3	3	6	7	9	9	12	13	15	15	18	19	21
$f^\dagger(n)$	0	1	1	3	3	3	4	5	5	7	7	7	8	9
$f^\dagger(f(n) + 1) + 1$	2	4	4	5	6	8	8	9	10	12	12	13	14	16

Tableau 2.15 – Exemple de séquence non décroissante et de son inverse au sens de J. LAMBEK et L. MOSER.  $f$  est la séquence cumulée de la séquence périodique  $(1, 2, 0, 3)$ ,  $f^\dagger$  son inverse.  $f^\dagger(f(n) + 1) + 1$  identifie le rang du prochain incrément de valeur dans  $f$ . Par exemple,  $f(6) = 9$ ,  $f^\dagger(f(6) + 1) + 1 = 8$  est le rang d'apparition de la plus petite valeur supérieure à 9, en l'occurrence 12.

$n$	1	2	3	4	5	6	7	8	9	10	11	12	13	14
$u(n)$	1	2	4	5	6	8	9	10	12	13	14	16	17	18
$v(n)$	1	3	6	7	9	12	13	15	18	19	21	24	25	27

Tableau 2.16 – Itération dans une séquence d'entiers non décroissante.  $u(n)$  prend successivement les valeurs de tous les indices pour lesquels la fonction  $f$  du tableau 2.15 croît strictement.

**Proposition 2.17.**  $f^\dagger(f(m) + 1) + 1$  est l'indice du premier terme supérieur à  $m$  pour lequel  $f$  croît strictement.

$$\begin{aligned}
f^\dagger(f(m) + 1) + 1 = m' &\iff \begin{cases} f^\dagger(f(m) + 1) < m' \\ f^\dagger(f(m) + 1) \geq m' - 1 \end{cases} \\
&\iff \begin{cases} f(m') \geq f(m) + 1 \\ f(m' - 1) < f(m) + 1 \end{cases} \\
&\iff f(m') > f(m) \text{ et } f(m' - 1) \leq f(m).
\end{aligned}$$

La proposition 2.17 est illustrée par un exemple dans le tableau 2.15.

Soient  $u$  et  $v$  deux séquences telles que  $u(1) = 1$ ,  $u(n+1) = f^\dagger(v(n) + 1) + 1$ ,  $v(n) = f(u(n))$ .  $u$  prend les valeurs des indices pour lesquels  $f$  croît et  $v$  les valeurs de  $f$  pour ces indices.

### Algorithme

Le calcul de l'inverse de la séquence  $f$  dont les premiers termes sont donnés, le dernier se répétant indéfiniment, est obtenu grâce à l'algorithme 2.18. L'algorithme 2.19, pour sa part, calcule l'inverse de la séquence cumulée d'une séquence périodique avec un offset constant  $f_o$  et des valeurs négatives ramenées à 0 :  $\forall r > 1, f^\Sigma(r) = \max\{0; f_o + \sum_{s=1}^r f(s)\}$ .

## 2.3 Distances à poids variables

Avant d'introduire la classe des distances à poids variables dans la section 2.3.2 page 41, revenons sur ce qui distingue les distances discrètes classiques les unes des autres. Alors que nous avons expliqué précédemment que les distances sont déterminées à la fois par une définition de chemin et par une manière de mesurer leur longueur, nous allons mettre en évidence que la seconde suffit. Les coûts de déplacement des distances que nous introduirons combineront

---

**Algorithme 2.18:** Calcul de  $g = f^\dagger$ , inverse de LAMBEK-MOSER de la séquence non décroissante  $f$ . Les  $f_p$  premiers termes de  $f$  sont donnés, le dernier se répétant indéfiniment :  $\forall m \geq f_p, f(m) = f(f_p)$ . L'algorithme calcule les  $g_p$  premiers éléments de  $g$ , le dernier se répétant indéfiniment :  $\forall n \geq g_p, g(n) = g(g_p)$ .

---

**Données :**  $f_p, f(1) \cdots f(f_p)$   
**Résultat :**  $g_p, g(1) \cdots g(g_p)$   
 $m \leftarrow 1$   
 $n \leftarrow 1$   
**faire**  
  **tant que**  $m \leq f_p$  **et**  $f(m) = n - 1$  **faire**  $m \leftarrow m + 1$   
  **si**  $m > f_p$  **alors**  
  |  $g(n) \leftarrow \infty$   
  **sinon**  
  |  $g(n) \leftarrow m - 1$   
  **fin**  
   $n \leftarrow n + 1$   
**tant que**  $m \leq f_p$  **et**  $f(m) \neq \infty$   
 $g_p \leftarrow n - 1$

---



---

**Algorithme 2.19:** Calcul de  $g^\Sigma + g_o$ , inverse de LAMBEK-MOSER de la séquence  $f^\Sigma + f_o$ .  $f^\Sigma$  est la séquence cumulée de la séquence  $f_p$ -périodique  $f$  avec un offset constant  $f_o$  et des valeurs négatives ramenées à 0.  $\forall r > 1, f^\Sigma(r) = \max\{0; f_o + \sum_{s=1}^r f(s)\}$ .  $g^\Sigma$  est la séquence cumulée de la séquence  $g_p$ -périodique  $g$  avec un offset constant  $g_o$  et des valeurs négatives ramenées à 0 :  $\langle a \rangle_b$  représente le reste de la division entière de  $a$  par  $b$ .

---

**Données :**  $f_p, f(1) \cdots f(f_p)$  et  $f_o$   
**Résultat :**  $g_p, g(1) \cdots g(g_p)$  et  $g_o$   
 $g_p \leftarrow \sum_{i=1}^{f_p} f(i);$   
// Dans la suite,  $f(m)$  désigne  $f(\langle m - 1 \rangle_{f_p} + 1)$  et  $g(n)$  désigne  $g(\langle n - 1 \rangle_{g_p} + 1)$   
 $n = f_o + 1;$   
 $m_0 \leftarrow f_p;$   
**pour**  $m \leftarrow 1$  **à**  $f_p$  **faire**  
  **si**  $f(m) \neq 0$  **alors**  
  |  $m_0 \leftarrow \min\{m_0; m\};$   
  **fin**  
   $g(n) \leftarrow g(n) + 1;$   
   $n \leftarrow n + f(m);$   
**fin**  
 $g_o \leftarrow m_0 - 1 - g(f_o + 1);$   
**pour**  $n \leftarrow 1$  **à**  $-f_o$  **faire**  
   $g_o \leftarrow g_o + g(n + f_o);$   
**fin**  
**pour**  $n \leftarrow 0$  **à**  $-f_o$  (**pas**  $-1$ ) **faire**  
   $g_o \leftarrow g_o - g(n + f_o);$   
**fin**

---

finalemt les deux paramètres de variabilité des distances classiques : le vecteur de déplacement emprunté et la distance déjà parcourue.

### 2.3.1 Variabilité des coûts de déplacements dans les distances

#### Distances basées sur un voisinage unique

Considérons une distance simple  $d$  engendrée par le voisinage  $\mathcal{N}$ .

$$\mathcal{L}(p, p_1, \dots, q) = r \implies \forall \vec{v} \in \mathcal{N}, \mathcal{L}(p, p_1, \dots, q, q + \vec{v}) = r + 1, \quad (2.15)$$

A. ROSENFELD et J. L. PFALTZ définissent les distances  $d_4$  et  $d_8$  (appelées  $d$  et  $d^*$ ) tout en précisant que les seuls chemins permis ne contiennent que des points distincts [section 4.1 p. 481 148]. Cette restriction est inutile pour la définition de la distance car un chemin dans lequel un point apparaît plus d'une fois ne peut être minimal. De la même manière, ils excluent explicitement le vecteur nul des 4 et 8-voisinages, interdisant par construction la répétition d'un point causée par un déplacement nul. Encore une fois, l'incidence sur la fonction distance est nulle. Notons qu'en termes de distance, interdire un chemin équivaut à lui affecter une longueur infinie de sorte qu'il ne peut être minimal. La distance simple engendrée par le voisinage  $\mathcal{N}$  peut donc être définie comme la longueur du plus court chemin, entendu au sens large, avec la mesure de longueur suivante :

$$\mathcal{L}(p, p_1, \dots, q) = r \implies \forall \vec{v}, \mathcal{L}(p, p_1, \dots, q, q + \vec{v}) = r + c_{\vec{v}},$$

avec

$$c_{\vec{v}} = \begin{cases} 1 & \text{si } \vec{v} \in \mathcal{N} \\ \infty & \text{sinon} \end{cases}$$

#### Distances à séquence de voisinages

Pour une distance caractérisée par la séquence de voisinages  $B$  :

$$\mathcal{L}(p, p_1, \dots, q) = r \implies \forall \vec{v} \in \mathcal{N}_{B(r)}, \mathcal{L}(p, p_1, \dots, q, q + \vec{v}) = r + 1, \quad (2.16)$$

ou de manière équivalente,

$$\mathcal{L}(p, p_1, \dots, q) = r \implies \forall \vec{v}, \mathcal{L}(p, p_1, \dots, q, q + \vec{v}) = r + c_{\vec{v}}^B(r),$$

dans lequel l'incrément local de longueur  $c_{\vec{v}}^B(r)$  est unitaire pour un vecteur présent dans le voisinage désigné par  $B(r)$  et infini dans le cas contraire :

$$c_{\vec{v}}^{B(r)} = \begin{cases} 1 & \text{si } \vec{v} \in \mathcal{N}_{B(r)} \\ \infty & \text{sinon} \end{cases} \quad (2.17)$$

#### Distances de chanfrein

Dans le cas d'une distance pondérée de masque  $\mathcal{M} = \{M_k \in \mathbb{Z}^n \times \mathbb{N}^*\}_{1 \leq k \leq m}$ , l'incrément de longueur dépend uniquement du vecteur de déplacement mais pas de la longueur déjà parcourue :

$$\mathcal{L}(p, p_1, \dots, q) = r \implies \forall \vec{v}, \mathcal{L}(p, p_1, \dots, q, q + \vec{v}) = r + c_{\vec{v}}, \quad (2.18)$$

$$c_{\vec{v}} = \begin{cases} w_k & \text{si } (\vec{v}_k; w_k) = M_k \in \mathcal{M} \\ \infty & \text{sinon} \end{cases} \quad (2.19)$$



Pour résumer, l'incrément de longueur pour un vecteur  $\vec{v}$  et une distance déjà parcourue  $r$ , est 1 ou  $\infty$ , indépendamment de  $r$  dans le cas des distances simples, il est égal à 1 ou  $\infty$  selon que le vecteur  $\vec{v}$  appartient ou non au voisinage  $\mathcal{N}_{B(r)}$  pour une distance à séquence de voisinage, il est dans  $\mathbb{N}^* \cup \{\infty\}$  conformément au masque de chanfrein et indépendamment de  $r$  pour une distance pondérée.

Dans la suite, nous proposons d'utiliser un incrément de distance, noté  $c_{\vec{v}}(r)$ , à valeurs dans  $\mathbb{N}^* \cup \{\infty\}$  qui dépende à la fois du vecteur de déplacement  $\vec{v}$  et de la distance déjà parcourue  $r$ . Conformément aux remarques précédentes, l'incrément de distance associé au déplacement nul sera toujours unitaire :

$$\forall r \in \mathbb{N}, c_{\vec{0}}(r) = 1. \quad (2.20)$$

Notons que M. YAMASHITA et N. HONDA incluent explicitement le vecteur nul dans le voisinage [188].

### 2.3.2 Définition des distances à poids variables

**Définition 2.20** (Chemin). On appelle *chemin de  $p$  vers  $q$* , toute séquence finie de points  $P = (p = p_0, p_1, \dots, p_n = q)$  contenant au moins un point, et  $\mathcal{P}(p, q)$ , l'ensemble de ces chemins.

Notons que cette définition de chemin est indépendante d'une quelconque relation d'adjacence entre les points. La séquence  $P = (p)$  est admise comme chemin sans déplacement de  $p$  vers lui-même. Ce chemin est distinct du chemin  $P = (p, p)$  de  $p$  vers lui-même avec un déplacement de vecteur nul.

**Définition 2.21** (Longueurs partielle et totale d'un chemin). Soit  $\mathcal{N} = \{\vec{v}_i\}$  un ensemble de vecteurs contenant le vecteur nul  $\vec{0}$  et les coûts de déplacement positifs  $c_{\vec{v}_i}$  (avec en particulier  $c_{\vec{0}}(r) = 1$  et  $c_{\vec{v} \notin \mathcal{N}}(r) = \infty$ ). La longueur totale du chemin  $P = (p_0, p_1, \dots, p_n)$  est :

$$\mathcal{L}(P) = \mathcal{L}_n(P), \quad (2.21)$$

où  $\mathcal{L}_i(P)$  est la longueur partielle du chemin tronqué à ses  $i + 1$  premiers points, donc à ses  $i$  premiers déplacements (cf. figure 2.22) :

$$\mathcal{L}_0(P) = \mathcal{L}(p_0) = 0, \quad (2.22)$$

$$\mathcal{L}_{i+1}(P) = \mathcal{L}(p_0, \dots, p_{i+1}) = \mathcal{L}_i(P) + c_{\overrightarrow{p_i p_{i+1}}}(\mathcal{L}_i(P)). \quad (2.23)$$

**Définition 2.23.** On note  $C_{\vec{v}_i}(r) = r + c_{\vec{v}_i}(r)$ .  $c_{\vec{v}_i}(r)$  est le coût relatif (ou incrément de longueur) du déplacement  $\vec{v}_i$  effectué après avoir parcouru la longueur  $r$ .  $C_{\vec{v}_i}(r)$  représente la longueur partielle du chemin après ce déplacement (ou coût absolu de ce déplacement) :

$$\mathcal{L}_{i+1}(P) = \mathcal{L}_i(P) + c_{\overrightarrow{p_i p_{i+1}}}(\mathcal{L}_i(P)) = C_{\overrightarrow{p_i p_{i+1}}}(\mathcal{L}_i(P)). \quad (2.24)$$

**Définition 2.24.** La (pseudo-)distance induite par  $(\{\vec{v}_i\}, c_{\vec{v}_i})$  est définie par :

$$d(p, q) = 0 \iff p = q$$

$$d(p, q) = \min_{P \in \mathcal{P}(p, q)} \{\mathcal{L}(P)\}.$$

Dans la construction d'un chemin, les poids associés aux différents vecteurs du voisinage sont déterminés uniquement selon la distance déjà parcourue, indépendamment de la combinaison de déplacements empruntés pour parvenir à cette distance. On associe au déplacement nul un poids constant unitaire, autrement dit,  $\mathcal{L}(p, p) = 1$ . Cela ne signifie nullement que la distance d'un point à lui-même soit non nulle, car le plus court chemin d'un point  $p$  à lui-même reste le chemin de longueur nulle,  $\mathcal{L}(p) = 0$ .

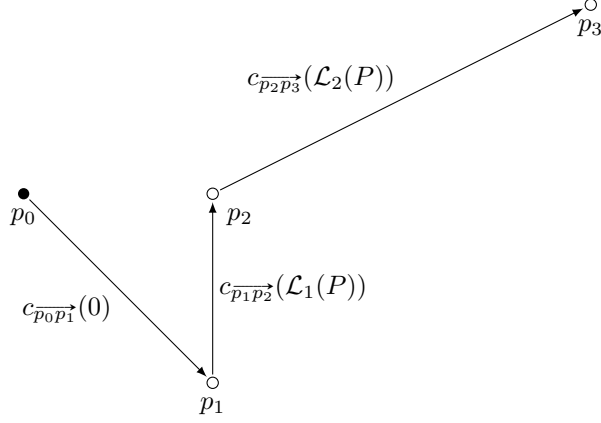


FIGURE 2.22 – Coût total d'un chemin  $P = (p_0, p_1, p_2, p_3)$ . Les coûts de déplacement  $\overrightarrow{p_0p_1}$ ,  $\overrightarrow{p_1p_2}$  et  $\overrightarrow{p_2p_3}$  dépendent des coûts partiels  $\mathcal{L}_0(P) = 0$ ,  $\mathcal{L}_1(P) = c_{\overrightarrow{p_0p_1}}(0) + 0$  et  $\mathcal{L}_2(P) = c_{\overrightarrow{p_1p_2}}(\mathcal{L}_1(P)) + \mathcal{L}_1(P)$ . Le coût total de  $P$  est  $c_{\overrightarrow{p_2p_3}}(\mathcal{L}_2(P)) + \mathcal{L}_2(P) = C_{\overrightarrow{p_2p_3}}(\mathcal{L}_2(P))$ .

**Définition 2.25.** On appelle *coût relatif* (resp. *absolu*) *minimal de déplacement*, noté  $\hat{c}$  (resp.  $\hat{C}$ ), la quantité  $\hat{c}_{\vec{v}}(r) = \min \{c_{\vec{v}}(s) + s - r, \forall s \geq r\}$  (resp.  $\hat{C}_{\vec{v}}(r) = \min \{C_{\vec{v}}(s), \forall s \geq r\}$ ).

**Proposition 2.26.** Soit un ensemble de vecteur  $\mathcal{N} = \{\vec{v}_i\}$  et les incréments de distance  $c_{\vec{v}}(r)$ . Il existe un chemin  $P$  de  $p$  vers  $q$  de longueur  $\mathcal{L}(P) = r$  mesuré avec les coûts de déplacement  $c_{\vec{v}}(r)$  si et seulement s'il existe un chemin  $P'$  de  $p$  vers  $q$  de longueur  $\mathcal{L}'(P') = r$  mesuré avec les coûts minimaux de déplacement  $\hat{c}_{\vec{v}}(r)$ .

*Démonstration.* Notons  $\mathcal{L}_i(P)$  la longueur du chemin partiel  $P$  après  $i$  déplacements,  $\mathcal{L}_i(P) = \mathcal{L}_i(p_0, p_1, \dots, p_i)$ ,  $m_0 = 1, m_{0 < i \leq n} = 1 + \mathcal{L}_i(P) - \mathcal{L}_{i-1}(P) - \hat{c}_{\overrightarrow{p_{i-1}p_i}}(\mathcal{L}_{i-1}(P)) = 1 + c_{\overrightarrow{p_{i-1}p_i}}(\mathcal{L}_{i-1}(P)) - \hat{c}_{\overrightarrow{p_{i-1}p_i}}(\mathcal{L}_{i-1}(P))$  et  $M_i = \sum_{j=0}^i m_j$  somme cumulée des  $m_i$ . Clairement, si  $\mathcal{L}(P)$  est finie alors chaque  $m_i$  est fini et strictement positif car  $\hat{c}_{\vec{v}}(r)$  est inférieur ou égal à  $c_{\vec{v}}(r)$  par construction. On pose  $P'$  le chemin (fini) formé par  $m_i$  occurrences de chaque point  $p_i$  :

$$P' = (p_0, \underbrace{p_1 \dots p_1}_{m_1}, \dots, \underbrace{p_i \dots p_i}_{m_i}, \dots, \underbrace{p_n \dots p_n}_{m_n}).$$

Prenons comme hypothèse de récurrence que la longueur du chemin partiel  $P'$  après  $m_i$  occurrences de  $p_i$ ,  $\mathcal{L}'_{M_{i-1}}(P')$ , est égale à  $\mathcal{L}_i(P)$ . Cette hypothèse est vérifiée pour  $i = 0$  car  $\mathcal{L}'_{M_0-1}(P') = \mathcal{L}'_{m_0-1}(P') = \mathcal{L}'_0(P') = 0 = \mathcal{L}_0(P)$ . Si l'hypothèse est vraie au rang  $i - 1$ , alors la longueur du chemin partiel  $P'$ , après la première occurrence de  $p_i$  vaut  $\mathcal{L}'_{M_{i-1}}(P') = \mathcal{L}_{i-1}(P) + \hat{c}_{\overrightarrow{p_{i-1}p_i}}(\mathcal{L}_{i-1}(P))$  et, après  $m_i - 1$  répétitions de  $p_i$ , vaut :  $\mathcal{L}'_{M_{i-1}+m_i-1}(P') = \mathcal{L}'_{M_{i-1}}(P') = \mathcal{L}_{i-1}(P) + \hat{c}_{\overrightarrow{p_{i-1}p_i}}(\mathcal{L}_{i-1}(P)) + m_i - 1 = \mathcal{L}_{i-1}(P) + c_{\overrightarrow{p_{i-1}p_i}}(\mathcal{L}_{i-1}(P)) = \mathcal{L}_i(P)$  et l'hypothèse est vraie au rang  $i$ . Il existe donc pour tout chemin de longueur finie  $r$  mesurée par  $\mathcal{L}$ , un chemin de longueur identique mesurée par  $\mathcal{L}'$ . Ceci est illustré sur la figure 2.27.

Réciproquement, soit  $P'$  un chemin de longueur finie mesurée par  $\mathcal{L}'$  et  $\mathcal{L}'_i(P')$  sa longueur partielle après  $i$  déplacements. On construit un chemin  $P$  où chaque point de  $P'$  apparaît  $m'_i$  fois consécutivement avec  $m'_i$  tel que  $m'_i - 1 + c_{\overrightarrow{p_i p_{i+1}}}(\mathcal{L}'_i(P')) + m'_i - 1 = \hat{c}_{\overrightarrow{p_i p_{i+1}}}(\mathcal{L}'_i(P'))$ . Par définition de  $\hat{c}$ ,  $\forall r, \exists s : \hat{c}_{\vec{v}}(r) = c_{\vec{v}}(s) + s - r$ , donc  $m'_i$  existe. On pose cette fois  $M'_0 = 0$  et  $M'_{0 < i \leq n} = \sum_{j=0}^{i-1} m'_j$ , somme cumulée des termes précédents de  $m'_i$ .

L'hypothèse de récurrence est que la longueur partielle de  $P$ , mesurée par  $\mathcal{L}$ , à la première occurrence de  $p_i$ ,  $\mathcal{L}_{M'_i}(P)$ , est égale à  $\mathcal{L}'_i(P')$ . Cette hypothèse est vérifiée pour  $i = 0$  avec une

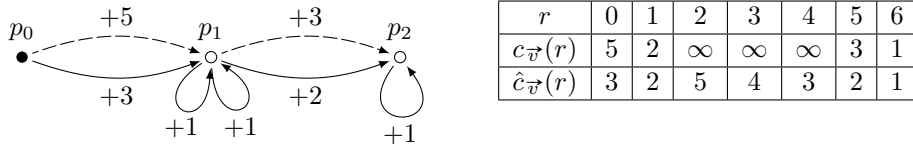


FIGURE 2.27 – Deux chemins de même longueur.  $P = (p_0, p_1, p_2)$ , donné et représenté en traits hachurés, a une longueur totale  $\mathcal{L}(P) = 8$  mesurée avec des coûts de déplacements  $\hat{c}$  tels que  $\hat{c}(r) + r$  n'est pas monotone.  $P' = (p_0, p_1, p_1, p_1, p_2, p_2)$ , représenté en traits pleins, est construit de telle sorte que  $\mathcal{L}(P) = \mathcal{L}'(P') = 8$ .

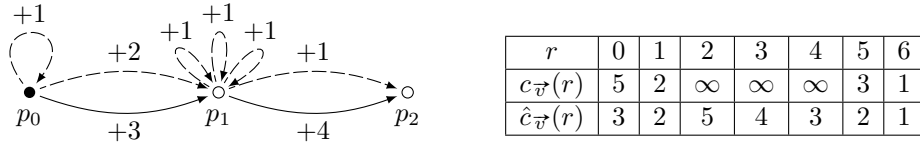


FIGURE 2.28 – Deux chemins de même longueur.  $P' = (p_0, p_1, p_2)$ , donné et représenté en traits pleins, a une longueur totale  $\mathcal{L}'(P') = 7$  mesurée avec des coûts de déplacements  $\hat{c}$  tels que  $\hat{c}(r) + r$  est monotone.  $P = (p_0, p_0, p_1, p_1, p_1, p_1, p_2)$ , représenté en traits hachurés, est construit de telle sorte que  $\mathcal{L}(P) = \mathcal{L}'(P') = 7$ .

longueur partielle nulle  $\mathcal{L}_{M'_0}(P) = \mathcal{L}_0(P) = 0 = \mathcal{L}'_0(P')$ . Si l'hypothèse est vérifiée au rang  $i$ , alors la longueur partielle de  $P$ , après  $m'_i - 1$  répétitions de  $p_i$ , est  $\mathcal{L}_{M'_i+m'_i-1}(P) = \mathcal{L}_{M'_i}(P) + m'_i - 1 = \mathcal{L}'_i(P') + m'_i - 1$ , et, à la première occurrence de  $p_{i+1}$ , vaut  $\mathcal{L}'_i(P') + m'_i - 1 + c_{\vec{p}_i \vec{p}_{i+1}}(\mathcal{L}'_i(P') + m'_i - 1) = \mathcal{L}'_i(P') + \hat{c}_{\vec{p}_i \vec{p}_{i+1}}(\mathcal{L}'_i(P')) = \mathcal{L}'_{i+1}(P')$  et l'hypothèse est aussi vérifiée au rang  $i + 1$ . Un exemple de chemin construit de cette manière est présenté sur la figure 2.28.  $\square$

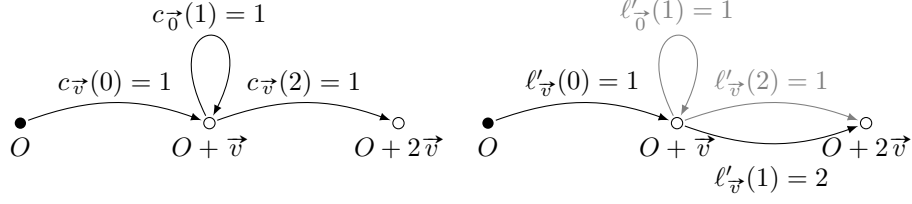
**Corollaire 2.29.** Les incréments de longueur  $c_{\vec{v}}$  et  $\hat{c}_{\vec{v}}$  induisent la même distance (ou pseudo-distance).

**Corollaire 2.30** (Monotonie de  $c_{\vec{v}}(r) + r$ ). Sans perte de généralité, c'est-à-dire sans changer la distance, il est toujours possible de définir les incréments de longueur  $c_{\vec{v}}(r)$  de telle sorte que  $c_{\vec{v}}(r) \leq c_{\vec{v}}(r + 1) + 1$ . Dans ce cas  $c_{\vec{v}}(r) + r \leq c_{\vec{v}}(r + 1) + r + 1$  et  $C_{\vec{v}}(r) = c_{\vec{v}}(r) + r$  est croissante.

Notons que les coûts minimaux de déplacements  $\hat{c}_{\vec{v}}$  sont déduits de  $c_{\vec{v}}$  indépendamment du vecteur de déplacement  $\vec{v}$  et des autres vecteurs du voisinage  $\mathcal{N}$ . La proposition 2.26 est vérifiée pour un ensemble de séquences de coûts de déplacement quels que soient les vecteurs de déplacement.

**Proposition 2.31** (Minimalité de  $\hat{c}$ ). Soit  $\hat{c}$  une séquence de coûts de déplacement relatifs minimaux. Toute séquence  $c'$  inférieure à  $\hat{c}$  (mais toujours positive) induit une (pseudo-)distance différente sur certains voisinages  $\mathcal{N}$ .

*Démonstration.* Soit un vecteur non nul  $\vec{v}$ , on associe  $\hat{c}$  à  $2\vec{v}$ . Considérons le plus petit  $r$  tel que  $c'_{\vec{v}}(r) < \hat{c}_{\vec{v}}(r)$ . On choisit  $c_{\vec{v}} = (r, \infty)$ . Alors  $d(O, O + \vec{v}) = d'(O, O + \vec{v}) = r$  et  $d'(O, O + 3\vec{v}) = r + c'_{\vec{v}}(r) < d(O, O + 3\vec{v}) = r + \hat{c}_{\vec{v}}(r)$ .  $\square$



**Proposition 2.32** (Conservation de l'ordre de la longueur de chemin par concaténation). Soient deux chemins  $P = (p_0, \dots, p_{n_P})$  et  $Q = (q_0, \dots, q_{n_Q})$  de longueurs  $\mathcal{L}(P)$  et  $\mathcal{L}(Q)$ ,  $\vec{v}$  un vecteur et  $P' = (p_0, \dots, p_{n_P}, p_{n_P} + \vec{v})$ ,  $Q' = (q_0, \dots, q_{n_Q}, q_{n_Q} + \vec{v})$  les chemins étendus par le déplacement  $\vec{v}$  et dont les longueurs sont  $\mathcal{L}(P')$  and  $\mathcal{L}(Q')$ . Alors :

$$\mathcal{L}(P) \leq \mathcal{L}(Q) \implies \mathcal{L}(P') \leq \mathcal{L}(Q'). \quad (2.25)$$

*Démonstration.* D'après (2.24),  $\mathcal{L}(P') = \hat{C}_{\vec{v}}(\mathcal{L}(P))$  et  $\mathcal{L}(Q') = \hat{C}_{\vec{v}}(\mathcal{L}(Q))$ . Par définition de  $\hat{C}_{\vec{v}}$ ,  $s \leq r \implies \hat{C}_{\vec{v}}(s) \leq \hat{C}_{\vec{v}}(r)$ , d'où (2.25).  $\square$

**Proposition 2.33.** L'insertion d'un déplacement supplémentaire ne réduit jamais la longueur d'un chemin mesurée avec  $\hat{C}$ . Soient  $P = (p_0, \dots, p_n)$  et  $Q = (q_0, q_1, \dots, q_{n+1})$  deux chemins tels que pour  $0 \leq r_0 \leq n$  :

$$\overrightarrow{q_r q_{r+1}} = \begin{cases} \overrightarrow{p_r p_{r+1}} & \text{si } r < r_0 \\ \vec{v} & \text{si } r = r_0 \\ \overrightarrow{p_{r-1} q_r} & \text{si } r > r_0 \end{cases} .$$

*Démonstration.*  $\mathcal{L}_{r_0}(Q) = \mathcal{L}_{r_0}(P)$  donc  $\mathcal{L}_{r_0+1}(Q) = \hat{C}_{\vec{v}}(\mathcal{L}_{r_0}(P)) > \mathcal{L}_{r_0}(P)$ . Par croissance de  $\hat{C}$ ,  $\forall r \geq r_0, \mathcal{L}_{r+1}(Q) \geq \mathcal{L}_r(P)$ .  $\square$

**Corollaire 2.34.** Si un chemin de longueur  $r$  mesurée avec  $\hat{C}$  est constitué de  $n_k$  déplacements  $\vec{v}_k$ ,  $1 \leq k \leq K$ , quel que soit l'ordre dans lequel ils apparaissent, alors pour toute combinaison de  $0 \leq m_k \leq n_k$ , il existe des chemins de longueur au plus  $r$  constitués de  $m_k$  déplacements  $\vec{v}_k$ .

### 2.3.3 Expression de $d$

D'après (2.20), si un incrément de distance  $c_{\vec{v}}(r)$  est fini,

$$\mathcal{L}(p_0, p_1, \dots, p_n = q) = s < r \implies \mathcal{L}(p_0, p_1, \dots, p_n = \underbrace{q, \dots, q}_{r-s}, q + \vec{v}) = r + c_{\vec{v}}(r)$$

$$d(p, q) = \min_{\vec{v}} \{ \mathcal{L}(P) + c_{\vec{v}}(\mathcal{L}(P)) \} \quad (2.26)$$

$P \in \mathcal{P}(p, q - \vec{v})$

Si  $C_{\vec{v}}(r) = c_{\vec{v}}(r) + r$  est croissante :

$$d(p, q) = \min_{\vec{v}} \{ C_{\vec{v}}(d(p, q - \vec{v})) \} \quad (2.27)$$

## 2.4 Propriétés des disques

D'après (2.20), tout chemin de  $p$  vers  $q$  de longueur inférieure à  $r$  peut être complété par des déplacements nuls pour atteindre la longueur  $r$  :

$$\mathcal{L}(p_0, p_1, \dots, p_n = q) = s < r \implies \mathcal{L}(p_0, p_1, \dots, \underbrace{p_n = q, \dots, q}_{1+r-s}) = r \quad (2.28)$$

**Proposition 2.35.** Il existe un chemin de longueur  $r$  de  $p$  vers  $q$  si et seulement si  $d(p, q) \leq r$ .

*Démonstration.* S'il existe un chemin de longueur  $r$  de  $p$  vers  $q$  alors par définition de la distance,  $d(p, q) = r$  si ce chemin est minimal,  $d(p, q) < r$  dans le cas contraire ce qui justifie l'induction. Réciproquement, si  $d(p, q) = s$  alors il existe un chemin de longueur  $s$  de  $p$  vers  $q$  qui, d'après (2.28), peut être étendu à toute longueur  $r \geq s$ .  $\square$

**Corollaire 2.36.** Autrement dit, pour toute valeur de  $r$  supérieure ou égale à  $d(p, q)$ , il existe un chemin de  $p$  vers  $q$  dont la longueur est exactement  $r$ . Le disque fermé de centre  $p$  et de rayon  $r$  est donc l'ensemble des points  $q$  pour lesquels un chemin de longueur exactement  $r$  existe.

$$\begin{aligned} q \in D_{\leq}(p, r) &\iff d(p, q) \leq r \\ &\iff \exists P \in \mathcal{P}(p, q), \mathcal{L}(P) = r. \end{aligned} \quad (2.29)$$

Donc :

$$D_{\leq}(p, r) = \{q : \exists P \in \mathcal{P}(p, q), \mathcal{L}(P) = r\}. \quad (2.30)$$

**Corollaire 2.37.** S'il existe un chemin de  $p$  à  $q$  de longueur inférieure à  $r$  et que l'incrément de longueur pour le vecteur  $\vec{v}$  et une distance parcourue égale à  $r$  est fini alors il existe un chemin de  $p$  vers  $q$  de longueur finie  $r + c_{\vec{v}}(r)$ .

### 2.4.1 Construction itérative des disques

À partir de (2.29), nous allons écrire une construction itérative des boules. Par définition de  $\mathcal{L}$ ,  $\mathcal{L}(P)$  est nulle si et seulement si  $P$  ne contient aucun déplacement. Le disque fermé de rayon nul ne contient que son centre, ce qui est conforme à (2.2) :

$$D_{\leq}(p, 0) = \{p\}.$$

Tout chemin de longueur  $r$  non nulle contient au moins deux points :

$$\begin{aligned} \forall r > 1, D_{<}(p, r) &= \{q : \exists P \in \mathcal{P}(p, q) \text{ et } \mathcal{L}(P) = r - 1\} \\ &= \{q : \exists \vec{v} \in \mathcal{N}, P \in \mathcal{P}(p, q - \vec{v}) \text{ et } C_{\vec{v}}(\mathcal{L}(P)) = r - 1\} \end{aligned} \quad (2.31)$$

selon l'invariance par translation :

$$\begin{aligned} \forall r > 1, D_{<}(p, r) &= \{q : \exists \vec{v} \in \mathcal{N}, P \in \mathcal{P}(p + \vec{v}, q) \text{ et } C_{\vec{v}}(\mathcal{L}(P)) = r - 1\} \\ &= \bigcup_{\vec{v} \in \mathcal{N}} \{q : \exists P \in \mathcal{P}(p + \vec{v}, q) \text{ et } C_{\vec{v}}(\mathcal{L}(P)) = r - 1\} \\ &= \bigcup_{\substack{\vec{v} \in \mathcal{N} \\ s: C_{\vec{v}}(s) = r - 1}} \{q : \exists P \in \mathcal{P}(p + \vec{v}, q) \text{ et } \mathcal{L}(P) = s\} \\ &= \bigcup_{\substack{\vec{v} \in \mathcal{N} \\ s: C_{\vec{v}}(s) = r - 1}} D_{<}(p + \vec{v}, s + 1). \end{aligned} \quad (2.32)$$

Soit pour des disques fermés :

$$\forall r > 0, D_{\leq}(p, r) = \bigcup_{\substack{\vec{v} \in \mathcal{N} \\ s: C_{\vec{v}}(s)=r}} D_{\leq}(p + \vec{v}, s) . \quad (2.33)$$

Pour une valeur particulière de  $r$ , les valeurs de  $s+1$  où  $s$  est tel que  $C_{\vec{v}}(s) = r-1$ , représentent les rayons des disques qui, réunis, forment le disque de rayon  $r$ . Toutes les valeurs de  $C_{\vec{v}}(s)$  étant strictement supérieures à  $s$ , il en découle que  $C_{\vec{v}}(s) = r-1 > s$  donc  $r > s$ . De plus, par construction,  $C_{\vec{v}}(s) = s+1$ , donc  $C_{\vec{v}}(s) = r-1$  est vérifié pour  $s = r-2$  et tout disque de rayon  $r \geq 2$  contient le disque précédent de rayon  $r-1$ .

Considérons la séquence de disques présentée dans le tableau 2.40 et les coûts de déplacement correspondants illustrés dans le tableau 2.38. Notons que ces derniers ne sont pas toujours croissants, même en ne tenant compte que des valeurs finies. Ils correspondent aux règles de construction de disques suivantes :

$$\begin{aligned} D_{<}(p, 1) &= \{p\} , \\ D_{<}(p, 2) &= D_{<}(p + \vec{v}_0, 1) \cup D_{<}(p + \vec{v}_1, 1) , \\ D_{<}(p, 3) &= D_{<}(p + \vec{v}_0, 2) \cup D_{<}(p + \vec{v}_1, 2) \cup D_{<}(p + \vec{v}_2, 1) \cup D_{<}(p + \vec{v}_3, 2) , \\ D_{<}(p, 4) &= D_{<}(p + \vec{v}_0, 3) , \\ D_{<}(p, 5) &= D_{<}(p + \vec{v}_0, 4) \cup D_{<}(p + \vec{v}_1, 3) , \\ D_{<}(p, 6) &= D_{<}(p + \vec{v}_0, 5) \cup D_{<}(p + \vec{v}_1, 4) \cup D_{<}(p + \vec{v}_2, 3) , \\ D_{<}(p, 7) &= D_{<}(p + \vec{v}_0, 6) \cup D_{<}(p + \vec{v}_1, 6) \cup D_{<}(p + \vec{v}_2, 5) \cup D_{<}(p + \vec{v}_2, 6) . \end{aligned}$$

Ces règles d'inclusion sont récapitulées dans une table de correspondance qui représente, pour chaque disque construit  $D_{<}(p, r)$  et chaque vecteur de déplacement  $\vec{v}$ , le rayon du disque inclus, noté  $S_{\vec{v}}(r)$  comme dans le tableau 2.39a, ou sous sa forme relative, son écart avec  $r$ , noté  $s_{\vec{v}}(r)$ , comme dans le tableau 2.39b. Lorsque, pour un vecteur de déplacement  $\vec{v}$ , plusieurs valeurs de  $C_{\vec{v}}(s)$  sont égales à  $r-1$ , plusieurs disques de centre  $p + \vec{v}$  apparaissent simultanément dans la construction du disque  $D_{<}(p, r)$ . C'est le cas dans notre exemple de la valeur 6 pour  $s \in \{4, 5\}$  et le vecteur  $\vec{v}_2$ . Les disques  $D_{<}(p + \vec{v}_2, 5)$  et  $D_{<}(p + \vec{v}_2, 6)$  participent à la construction du disque  $D_{<}(p, 7)$  mais  $D_{<}(p + \vec{v}_2, 5)$  étant inclus dans  $D_{<}(p + \vec{v}_2, 6)$ , il peut être omis. Inversement, si  $C_{\vec{v}}(s)$  n'est jamais égal à  $r-1$ , aucun disque n'apparaît dans la règle de construction de  $D_{<}(p, r)$ . Dans ce cas, par convention, la valeur 0 qui correspond à un disque vide sera utilisée pour  $S_{\vec{v}}$ , et  $-\infty$  pour  $s_{\vec{v}}$ . On note :

$$\begin{aligned} S_{\vec{v}}(r) &= \max \{0; s+1 : C_{\vec{v}}(s) = r-1\} \\ s_{\vec{v}}(r) &= \max \{-\infty; s-r : C_{\vec{v}}(s) = r-1\} \end{aligned}$$

et on a donc les règles de construction de disques :

$$D_{<}(p, r) = \bigcup_{\vec{v} \in \mathcal{N}} D_{<}(p + \vec{v}, S_{\vec{v}}(r)) \quad (2.34)$$

$$= \bigcup_{\vec{v} \in \mathcal{N}} D_{<}(p + \vec{v}, r + s_{\vec{v}}(r)) . \quad (2.35)$$

Grâce au corollaire 2.29, nous savons que les coûts de déplacement  $C_{\vec{v}}$  et  $\hat{C}_{\vec{v}}$  induisent les mêmes distances et donc les mêmes disques. Notons  $\hat{S}_{\vec{v}}(r) = \max \{0; s : \hat{C}_{\vec{v}}(s-1) < r\}$ ,  $\hat{S}_{\vec{v}}$  est

$r$	0	1	2	3	4	5	6
$C_{\vec{v}_0}(r)$	1	2	3	4	5	6	7
$C_{\vec{v}_1}(r)$	1	2	4	5	7	6	$\infty$
$C_{\vec{v}_2}(r)$	2	$\infty$	5	$\infty$	6	6	$\infty$
$C_{\vec{v}_3}(r)$	$\infty$	2	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$
(a) coûts absolus							
$r$	0	1	2	3	4	5	6
$c_{\vec{v}_0}(r)$	1	1	1	1	1	1	1
$c_{\vec{v}_1}(r)$	1	1	2	2	3	1	$\infty$
$c_{\vec{v}_2}(r)$	2	$\infty$	3	$\infty$	2	1	$\infty$
$c_{\vec{v}_3}(r)$	$\infty$	1	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$
(b) coûts relatifs							

Tableau 2.38 – Coûts de déplacement sous formes absolue (longueur du chemin après déplacement) et relative (incrément de déplacement).

une séquence d'entiers non-décroissante. Les valeurs de  $\hat{S}_{\vec{v}}$  sont appelées *indices de construction maximaux*. Par définition de l'inverse de LAMBEK-MOSER :

$$\hat{S}_{\vec{v}} = f^\dagger, \text{ avec } \forall r \geq 1, f(r) = \hat{C}_{\vec{v}}(r-1). \quad (2.36)$$

$$\forall r \geq 0, \hat{C}_{\vec{v}}(r) = \hat{S}_{\vec{v}}^\dagger(r+1). \quad (2.37)$$

Le tableau 2.39 montre les indices de construction maximaux pour la séquence de disque du tableau 2.40. Finalement :

$$\begin{aligned} D_{<}(p, r) &= D_{<}(p, r-1) \cup \bigcup_{\vec{v} \in \mathcal{N}^*} D_{<}(p + \vec{v}, \hat{S}_{\vec{v}}(r)) \\ &= \{p\} \cup \bigcup_{\vec{v} \in \mathcal{N}^*} D_{<}(p + \vec{v}, \hat{S}_{\vec{v}}(r)). \end{aligned} \quad (2.38)$$

La même construction peut être exprimée en termes de boules fermées :

$$\forall r > 1, D_{\leq}(p, r) = \{p\} \cup \bigcup_{\vec{v} \in \mathcal{N}^*} D_{\leq}(p + \vec{v}, \hat{S}_{\leq, \vec{v}}(r-1)), \quad (2.39)$$

où tous les  $\hat{S}_{\leq, \vec{v}}$  sont des séquences non décroissantes d'entiers naturels, avec en particulier pour le déplacement nul  $\forall r > 0, \hat{S}_{\leq, \vec{0}}(r) = r$ . Avec l'inverse de LAMBEK-MOSER décrit précédemment, on peut reformuler l'expression des coûts de déplacement à partir des règles de construction :

$$\forall r \geq 0, C_{\vec{v}}(r) = \hat{S}_{\leq, \vec{v}}^\dagger(r+1) + 1. \quad (2.40)$$

Notons, qu'une séquence finie de disques de rayon maximal  $l$  est, en pratique, équivalente à une séquence infinie de disques où les disques de rayon au moins  $l$  sont égaux. Dans ce cas, toutes les valeurs de  $\hat{S}_{\leq, \vec{v}}$  à l'indice  $l$  et au-delà sont égales et n'excèdent pas la valeur  $l$ . En conséquence,  $C^\dagger$  est infini pour le rayon  $l$  et au-delà.

$r$	1	2	3	4	5	6	7
$S_{\vec{v}_0}(r)$	0	1	2	3	4	5	6
$S_{\vec{v}_1}(r)$	0	1	2	0	3	4	6
$S_{\vec{v}_2}(r)$	0	0	1	0	0	3	6
$S_{\vec{v}_3}(r)$	0	0	2	0	0	0	0

(a) indices de construction absolus

$r$	1	2	3	4	5	6	7
$s_{\vec{v}_0}(r)$	$-\infty$	-1	-1	-1	-1	-1	-1
$s_{\vec{v}_1}(r)$	$-\infty$	-1	-1	$-\infty$	-2	-2	-1
$s_{\vec{v}_2}(r)$	$-\infty$	$-\infty$	-2	$-\infty$	$-\infty$	-3	-1
$s_{\vec{v}_3}(r)$	$-\infty$	$-\infty$	-1	$-\infty$	$-\infty$	$-\infty$	$-\infty$

(b) indices de construction relatifs

$r$	1	2	3	4	5	6	7
$\hat{S}_{\vec{v}_0}(r)$	0	1	2	3	4	5	6
$\hat{S}_{\vec{v}_1}(r)$	0	1	2	2	3	4	6
$\hat{S}_{\vec{v}_2}(r)$	0	0	1	1	1	3	6
$\hat{S}_{\vec{v}_3}(r)$	0	0	2	2	2	2	2

(c) indices de construction maximaux absolus

Tableau 2.39 – Table de construction d’une suite de disque.

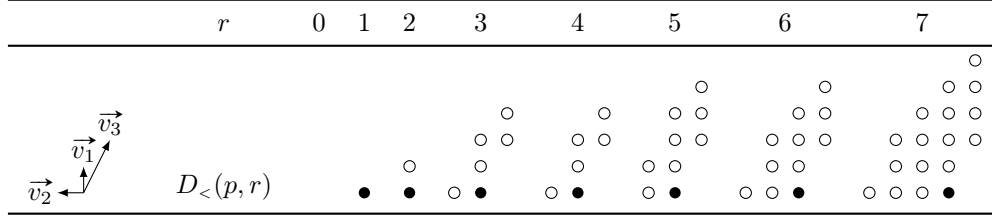


Tableau 2.40 – Construction d’une suite de disques induite par des coûts de déplacement variables selon la distance parcourue. Le “centre”  $p$  du disque est représenté par un symbole plein •.

### Représentation par somme de MINKOWSKI

**Définition 2.41** (Somme de MINKOWSKI). Soient deux ensembles  $A$  et  $B$ , la *somme de MINKOWSKI* de  $A$  et  $B$ , notée  $A \oplus B$  est l’ensemble des sommes d’un élément de  $A$  et d’un élément de  $B$  :

$$A \oplus B = \{a + b : a \in A, b \in B\} . \quad (2.41)$$

**Proposition 2.42.** Si une valeur particulière  $r$  n’apparaît jamais dans  $C_{\vec{r}}$  ailleurs qu’à l’indice  $r-1$  ( $\forall s \neq r-1, C_{\vec{r}}(s) \neq r$ ), alors le disque fermé (resp. ouvert) de rayon  $r$  (resp.  $r+1$ ) est la somme de MINKOWSKI du disque fermé de rayon  $r-1$  (resp.  $r$ ) avec l’ensemble  $\{O + \vec{v} : c_{\vec{r}}(r-1) = 1\}$ .



*Démonstration.*

$$\begin{aligned}
D_{\leq}(p, r) &= \bigcup_{\vec{v} \in \mathcal{N}: C_{\vec{v}}(r-1)=r} D_{\leq}(p + \vec{v}, r-1) \\
&= D_{\leq}(p, r-1) \oplus \bigcup_{\vec{v} \in \mathcal{N}: C_{\vec{v}}(r-1)=r} \{O\} \\
&= D_{\leq}(p, r-1) \oplus \{O + \vec{v} : C_{\vec{v}}(r-1) = r\} \\
&= D_{\leq}(p, r-1) \oplus \{O + \vec{v} : c_{\vec{v}}(r-1) = 1\} \quad \square
\end{aligned}$$

L'ensemble  $\{O + \vec{v} : c_{\vec{v}}(r-1) = 1\}$  est le disque fermé de rayon 1 obtenu par les poids relatifs  $\forall s \geq r-1, c'_{\vec{v}}(s) = c_{\vec{v}}(s-r+1)$ .

**Corollaire 2.43.** Soient  $B$  une séquence à valeurs dans  $[1; I]$ ,  $\{\mathcal{N}_1, \dots, \mathcal{N}_I\}$  un ensemble de  $I$  voisinages et pour chacun, les coûts de déplacement  $c_{\vec{v}}^i$  tels que :

$$c_{\vec{v}}^i(0) = \begin{cases} 1 & \text{si } \vec{v} \in \mathcal{N}_i \\ \infty & \text{sinon} \end{cases}.$$

La séquence de disques induite par  $c_{\vec{v}}(r) = c_{\vec{v}}^{B(r+1)}(0)$  est la séquence des sommes de MINKOWSKI des voisinages  $\mathcal{N}_{B(r)}$ .

**Proposition 2.44.** Plus généralement, si  $\forall r < r_1, C(r) \leq r_1$  alors tous les disques de rayon  $r_2$  au moins égal à  $r_1$  sont le produit de la somme de MINKOWSKI de  $D_{\leq}(p, r_1)$  avec le disque  $D'_{\leq}(O, r_2 - r_1)$  engendré par les incréments de distance  $c'_{\vec{v}}(r) = c_{\vec{v}}(r + r_1)$  (i.e.,  $C'_{\vec{v}}(r) = C_{\vec{v}}(r + r_1) - r_1$ ).

$$\forall r < r_1, C(r) \leq r_1 \implies \forall r_2 \geq r_1, D_{\leq}(p, r_2) = D_{\leq}(p, r_1) \oplus D'_{\leq}(O, r_2 - r_1) \quad (2.42)$$

*Démonstration.* (2.42) est vérifiée pour  $r_2 = r_1$  car  $D_{\leq}(p, r_1) = D_{\leq}(p, r_1) \oplus \{O\}$ . Supposons que (2.42) soit vérifiée dans l'intervalle  $[r_1, r_2]$ . On rappelle (2.33) :

$$D_{\leq}(p, r_2 + 1) = \bigcup_{\substack{\vec{v} \in \mathcal{N} \\ s: C_{\vec{v}}(s)=r_2+1}} D_{\leq}(p + \vec{v}, s).$$

Seules les valeurs de  $s$  dans l'intervalle  $[r_1, r_2]$  peuvent vérifier  $C_{\vec{v}}(s) = r_2 + 1$  et l'hypothèse s'applique pour  $D_{\leq}(p + \vec{v}, s)$ , donc :

$$\begin{aligned}
D_{\leq}(p, r_2 + 1) &= \bigcup_{\substack{\vec{v} \in \mathcal{N} \\ s: C_{\vec{v}}(s)=r_2+1}} D_{\leq}(p + \vec{v}, r_1) \oplus D'_{\leq}(O, s - r_1) \\
&= D_{\leq}(p + \vec{v}, r_1) \oplus \bigcup_{\substack{\vec{v} \in \mathcal{N} \\ s: C_{\vec{v}}(s)=r_2+1}} D'_{\leq}(O, s - r_1) \\
&= D_{\leq}(p + \vec{v}, r_1) \oplus \bigcup_{\substack{\vec{v} \in \mathcal{N} \\ s: C'_{\vec{v}}(s)=r_2-r_1+1}} D'_{\leq}(O, s) \\
&= D_{\leq}(p + \vec{v}, r_1) \oplus D'_{\leq}(O, r_2 - r_1 + 1).
\end{aligned}$$

Ce qui valide l'hypothèse pour l'intervalle  $[r_1, r_2 + 1]$ . □

Lorsque pour un rayon  $r$  et pour tous les vecteurs de déplacements  $\vec{v}$ , les valeurs définies de  $S_{\vec{v}}(r)$  sont toutes égales à  $r - 1$ , le disque  $D_{<}(p, r)$  est le produit de la somme de MINKOWSKI de  $D_{<}(p, r - 1)$  avec l'ensemble des vecteurs de déplacements dont le coût absolu est  $r - 1$ .

Lorsque pour un rayon  $r$ , les incréments de distance  $c_{\vec{v}}(r)$  de tous les vecteurs de déplacement  $\vec{v} \in \mathcal{N}$  sont unitaires, alors le disque de rayon  $r + 1$  est la somme de MINKOWSKI du disque de rayon  $r$  avec le voisinage  $\mathcal{N}$ .

La proposition 2.44 fournit une condition suffisante (mais pas nécessaire) pour construire des séquences de disques par somme de MINKOWSKI. En particulier, quand les coûts relatifs de déplacement sont soit 1 soit  $\infty$  comme dans (2.17), alors la proposition 2.44 s'applique à toutes les valeurs positives de rayon. Les disques des distances à séquence de voisinages sont donc produits par des somme de MINKOWSKI, ce qui est bien connu. Dans le chapitre 3, la proposition 2.44 sera exploitée avec (2.39) et (2.40) pour construire des séquences de disques où les sommes de MINKOWSKI sont décomposées en plusieurs étapes.

## 2.5 Conclusion

Ce chapitre a décrit une classe de distances généralisées dont les coûts de déplacement varient à la fois en fonction du vecteur de déplacement et avec la distance déjà parcourue. Ce schéma est suffisamment général pour englober les distances à séquence de voisinage, les distances de chanfrein et les distances généralisées produites par somme de MINKOWSKI. Nous avons montré qu'un ensemble de coûts de déplacement pouvait être donné sous une forme minimale, où chaque vecteur de déplacement se voit associer une séquence non-décroissante de coûts, sans altérer les valeurs de distances.

# Chapitre 3

## Transformation en distance

### 3.1 Introduction

Les opérations de morphologie mathématique ensembliste ont un coût de calcul important lorsque les éléments structurants sont de grande taille. De nombreuses méthodes ont été présentées pour réduire ce coût. Les approches sont principalement de deux ordres. La première consiste à décomposer un élément structurant de grande taille en éléments plus petits. L'application de l'élément structurant initial consiste à utiliser successivement les éléments de la décomposition. Le coût total est donc directement lié au nombre d'opérations et dépend donc de la taille de l'élément initial. La seconde consiste à binariser une carte de distance qui, elle, peut être calculée en un nombre de balayage fixe, avec un coût constant par pixel, indépendamment de la taille de l'élément structurant. Cependant, ce dernier doit pouvoir être exprimé comme le disque d'une distance ce qui restreint fortement les formes disponibles.

La méthode proposée ici utilise la transformation en distance généralisée avec une distance à poids variables telle qu'elle a été présentée dans le chapitre 2. Elle permet de décomposer un polygone convexe quelconque en une séquence de disques obtenus par des réunions ensemblistes y compris lorsque celui-ci n'est pas décomposable en sommes de MINKOWSKI.

La complexité temporelle d'une érosion morphologique avec une forme convexe est constante par pixel, indépendamment de la taille de l'élément structurant. La complexité spatiale est réduite à une ou quelques lignes d'image alors que les méthodes classiques par transformation en distance nécessitent la mémorisation complète de l'image. La structure des algorithmes permet leur mise en œuvre dans une chaîne de traitements avec un calcul à la volée, ligne par ligne, des résultats.

La section 3.2 rappelle des travaux antérieurs sur le calcul de la transformation en distance par propagation pour les distances définies par des chemins.

### 3.2 Préliminaires

**Définition 3.1** (transformation en distance). La *transformation en distance*  $DT_X$  du sous-ensemble  $X \in \mathbb{Z}^n$  est une fonction  $\mathbb{Z}^n \mapsto \mathbb{N}$  qui associe à chaque point  $p$  sa distance depuis le plus proche point du complémentaire de  $X$  :

$$\begin{aligned} DT_X : \mathbb{Z}^n &\rightarrow \mathbb{N} \\ DT_X(p) &= \min \{d(q, p) : q \in \mathbb{Z}^n \setminus X\}. \end{aligned} \tag{3.1}$$

De manière équivalente, puisque tous les points dont la distance à  $p$  est inférieure à  $DT_X(p)$  appartiennent à  $X$  ( $\check{D}(p, DT_X(p) - 1) \subset X$ ) et au moins un point hors de  $X$  a une distance vers

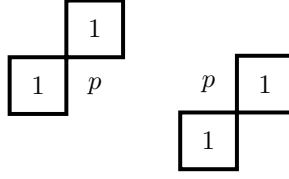


FIGURE 3.2 – Séparation du voisinage  $\mathcal{N}_4$  en  $\mathcal{N}_4^+$  et  $\mathcal{N}_4^-$  pour la propagation de la distance  $d_4$  en deux balayages séquentiels [148].

$p$  égale à  $DT_X(p)$  ( $\check{D}(p, DT_X(p)) \not\subset X$ ),  $DT_X$  peut s'exprimer en termes d'inclusions de disques dans  $X$  :

$$\begin{aligned} DT_X(p) &= \max \{r : \check{D}(p, r-1) \subset X\} \\ &= \max \{r : \check{D}_<(p, r) \subset X\}, \end{aligned} \quad (3.2)$$

soit encore :

$$\begin{aligned} DT_X(p) \geq r &\iff \check{D}(p, r-1) \subset X \\ &\iff \check{D}_<(p, r) \subset X. \end{aligned} \quad (3.3)$$

Ici,  $\check{D} = -D$  représente le symétrique de  $D$  par rapport à l'origine. La carte de distance est généralement définie comme la distance *au* fond, ce qui est équivalent à la distance *depuis* le fond par symétrie. Cette égalité est perdue quand on relâche la contrainte de symétrie (2.1b), et la définition 3.1 correspond mieux au mode de calcul de la carte de distance qui propage les longueurs de chemins depuis les points du fond.

### 3.2.1 Transformation en distance par propagation

Dans leur article fondateur, A. ROSENFELD et J. L. PFALTZ ont à la fois introduit la notion de distance discrète en définissant les distances  $d_4$ ,  $d_8$  et donné un algorithme séquentiel pour en calculer les transformations en distance [148].  $DT_X$  peut être calculée par l'application séquentielle de deux opérateurs (les notations diffèrent de l'article original) :

$$DT_X^1(p) = \begin{cases} \min \{1 + DT_X^1(p - \vec{v}), \vec{v} \in \mathcal{N}_4^+\} & \text{si } p \in X \\ 0 & \text{sinon} \end{cases} \quad (3.4)$$

$$DT_X(p) = \min \{DT_X^1(p); DT_X(p - \vec{v}), \vec{v} \in \mathcal{N}_4^-\}. \quad (3.5)$$

Le premier opérateur est appliqué dans un ordre de balayage séquentiel classique, de haut en bas et de gauche à droite, le second dans un ordre inversé.  $\mathcal{N}_4^+$  contient les vecteurs de  $\mathcal{N}_4$  dirigés vers l'avant (donc compatibles avec le premier balayage) et  $\mathcal{N}_4^-$  ceux dirigés en arrière (figure 3.2).

Cet algorithme a connu plusieurs évolutions. U. MONTANARI propose plusieurs tailles de voisinages et remplace les incréments constants par les longueurs des vecteurs [95] (figure 3.3). H. G. BARROW, J. M. TENENBAUM, R. C. BOLLES et H. C. WOLF présentent cet algorithme avec les poids 2 et 3 correspondant à la distance de chanfrein 2, 3 [10] (figure 3.4). G. BORGEFORS recommande l'utilisation des poids entiers 3 et 4 pour un masque  $3 \times 3$ , ou 5, 7, 11 pour un masque  $5 \times 5$  en 2D et ne préconise pas de masque  $7 \times 7$  (figure 3.5).

Le cas des distances à séquence de voisinages est moins direct. En effet, un chemin minimal d'une distance de chanfrein reste valide et minimal si on change l'ordre des vecteurs de déplacement, d'après (2.8). Pour que (3.4) donne un résultat correct au point  $p$ , il faut que pour tout point  $q$  d'origine, il existe un chemin minimal de  $q$  vers  $p$  constitué d'un groupe de déplacements de  $\mathcal{N}^+$

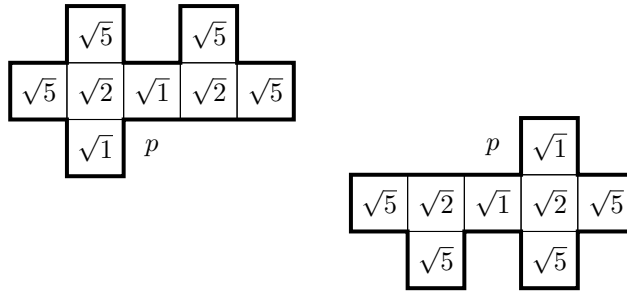


FIGURE 3.3 – Séparation du masque de U. MONTANARI utilisé pour la propagation d’une distance euclidienne approchée en deux balayages séquentiels [95].

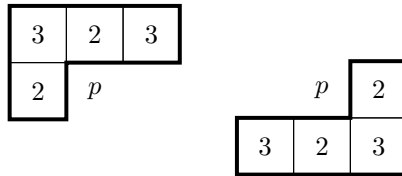


FIGURE 3.4 – Séparation d’un masque à valeurs entières utilisé pour la propagation d’une distance de chanfrein en deux balayages séquentiels [10].

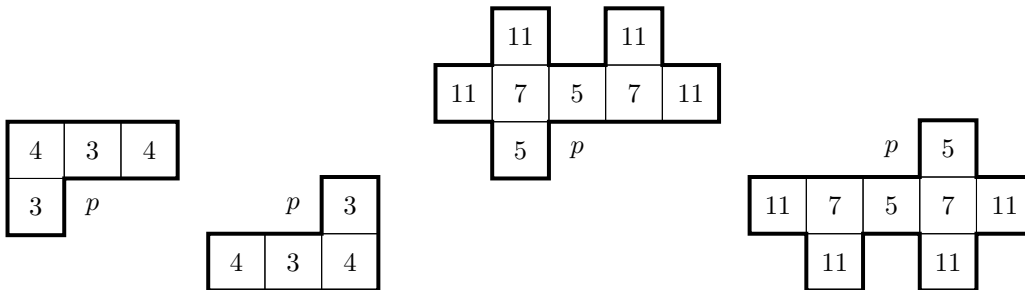


FIGURE 3.5 – Masques de chanfrein recommandée par G. BORGEFORS dans le plan discret [22].

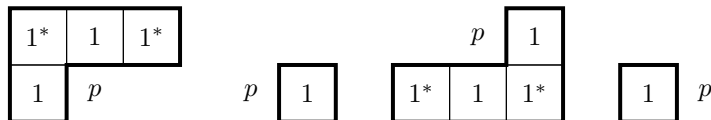


FIGURE 3.6 – Voisinages de MONTANARI pour le calcul de la distance octogonale.

suivis d'un ensemble de déplacements de  $\mathcal{N}^-$ . Cette décomposition n'est pas toujours possible pour les distances à séquence de voisinages.

L'algorithme *Sequential Octagonal Distance mapping* de P.-E. DANIELSSON utilise deux balayages principaux avec deux sous-balayages chacun [30] (voir figure 3.6). R. STRAND, B. NAGY, C. FOUARD et G. BORGEFORS étudient les conditions de validité de masques et d'ordres de parcours dans différents types de grilles et proposent notamment un algorithme en trois balayages dans la grille carrée [166]. S. CHEN et R. M. HARALICK appellent transformation en érosion récursive l'application de (3.4) à un voisinage quelconque [27]. X. WANG et G. BERTRAND vont plus loin en définissant une distance généralisée (non symétrique) avec un voisinage ou deux voisinages alternés qui, sous condition, peut se calculer en un balayage unique de l'image.

La propagation séquentielle en un nombre fixé de balayages n'est pas le seul moyen de calculer une transformation en distance. A. ROSENFELD et J. L. PFALTZ ont proposé un algorithme par érosions morphologiques successives qui s'applique aux distances  $d_4$ ,  $d_8$  et  $d_{oct}$  [147]. La valeur de distance est l'indice de l'itération à laquelle un pixel disparaît, mécanisme généralisé par S. YOKOI, J.-I. TORIWAKI et T. FUKUMURA [189].

Plutôt que de parcourir l'image dans un ordre prédéterminé, B. J. H. VERWER utilise une file de pixels ordonnée par valeur de distance [179, 181], principe repris par exemple dans des algorithmes d'amincissement [142].

La distance euclidienne, sort de ce cadre d'algorithmes de propagation, n'étant pas, dans les réseaux de points et contrairement aux espaces euclidiens continus, définie par des chemins. Des algorithmes de calcul exact avec complexité linéaire existent néanmoins [94, 149].

### 3.2.2 Morphologie mathématique ensembliste

**Définition 3.7** (dilatation ensembliste). Soient  $A$  et  $B$  deux sous-ensembles de  $\mathbb{R}^n$ . On appelle *transformation de Serra* ou *dilatation ensembliste*, ou encore simplement *dilatation*, l'ensemble [90, 151] :

$$A \oplus \check{B} = \{p : (B)_p \cap A \neq \emptyset\}. \quad (3.6)$$

où  $\oplus$  est la somme de MINKOWSKI (définition 2.41 page 48),  $(B)_p = \{x + p : x \in B\} = B \oplus \{p\}$  est le translaté de  $B$  par  $p$ , et  $\check{B} = -B = \{-x : x \in B\}$  est le *symétrique*, ou *transposé*, de  $B$  par rapport à l'origine.

La dilatation possède un élément neutre,  $\{O\}$ , mais pas d'opérateur inverse.

**Définition 3.8** (érosion ensembliste). Soient  $A$  et  $B$  deux sous-ensembles de  $\mathbb{R}^n$ . On appelle *érosion ensembliste*, ou *érosion*, l'ensemble [90, 151] :

$$A \ominus \check{B} = \{p : (B)_p \subseteq A\}. \quad (3.7)$$

où  $\overline{X} = \mathbb{R}^n \setminus X$  dénote le complémentaire de  $X$ .

L'érosion est l'opérateur dual de la dilatation :

$$\overline{A \ominus \check{B}} = \overline{A} \oplus \check{B}.$$

**Définition 3.9** (ouverture ensembliste). L'*ouverture ensembliste* de  $A$  par  $B$  est définie par :

$$\phi_B(A) = A \circ B = (A \ominus \check{B}) \oplus B.$$

L'ouverture ensembliste est croissante ( $B \subseteq C \implies A \circ B \subseteq A \circ C$ ), idempotente ( $(A \circ B) \circ B = A \circ B$ ) et anti-extensive ( $A \circ B \subseteq A$ ). Par extension toute fonction qui possède ces trois propriétés est appelée *ouverture algébrique*.

**Définition 3.10** (fermeture ensembliste). La *fermeture ensembliste* de  $A$  par  $B$  est :

$$A \bullet B = (A \oplus \check{B}) \ominus B .$$

La fermeture ensembliste est l'opérateur dual de l'ouverture ensembliste :

$$\overline{A \circ B} = \overline{A} \bullet B ,$$

elle est croissante, idempotente et extensive ( $A \bullet B \supseteq A$ ). Toute fonction partageant ces trois propriétés est une *fermeture algébrique*.

### Distance et érosion ensembliste

La transformation en distance est liée à l'érosion morphologique (3.7) par (3.3) : l'ensemble des points de  $X$  dont la distance depuis le fond est au moins  $r$  est l'érodé de  $X$  par le symétrique du disque ouvert de rayon  $r$  :

$$X \ominus D_{<}(O, r) = \{p : DT_X(p) \geq r\} . \quad (3.8)$$

Éroder un ensemble de points à partir de sa carte de distance consiste donc uniquement à effectuer une binarisation par seuillage dont le coût est négligeable. Le coût total dépend donc directement de celui de la transformation en distance. Les algorithmes en nombre fixe de balayages [22, 147, 189] d'image permettent donc d'appliquer une érosion avec un coût constant par pixel quelle que soit la taille de l'élément structurant. Par un déplacement du centre des disques, le coût peut être réduit à un balayage unique de l'image [27, 185].

Les disques des distances discrètes usuellement utilisées ont une variété de formes limitée comme en témoignent les figures 2.7a à 2.7d page 34. Les algorithmes de transformation en distance par propagation peuvent néanmoins être adaptés à des séquences de disques construits par dilations morphologique avec un élément structurant quelconque [27, 185] voire avec une alternance de deux éléments structurants [186].

### Décomposition morphologique

D'autres méthodes de réduction de complexité utilisent la propriété qu'une série d'érosions avec un ensemble d'éléments structurants est équivalent à une érosion unique avec la somme de MINKOWSKI des éléments :

$$(A \ominus \check{B}) \ominus \check{C} = A \ominus (\check{B} \oplus \check{C}) . \quad (3.9)$$

De nombreuses méthodes utilisant ce principe sont adaptées à des architectures matérielles capables d'effectuer une érosion en cycle d'horloge. La forme des éléments structurants de base dépend des capacités de l'architecture. Des méthodes de décomposition de polygones convexes ont été présentées pour des éléments structurants de base linéaires [56], ou limités à des voisinages locaux pour des machines 4- ou 8-connectées [187]. Ces décompositions donnent des mises en œuvre optimales sur des architectures parallèles ou *pipeline*, mais la complexité dépend néanmoins de la taille des éléments structurants.

Dans toutes ces méthodes, la limite est la décomposition par somme de MINKOWSKI. Lorsqu'une forme n'est pas décomposable, une option consiste à effectuer une réunion ensembliste de formes plus petits comme le présente la figure 3.11.d [82]. Dans ce cas, la décomposition initiale peut être obtenue par une transformation en distance généralisée en un balayage. La complexité de la dernière étape dépend cependant de la taille de l'élément structurant.

D'autres méthodes encore utilisent un nombre fixe de balayages, mais sont limitées à des formes telles que des lignes [78], des rectangles [178] ou requièrent une combinaison d'opérations pour des éléments plus généraux [162]. Pour les éléments structurants arbitraires, des méthodes combinatoires ou génétiques ont été proposées [3].

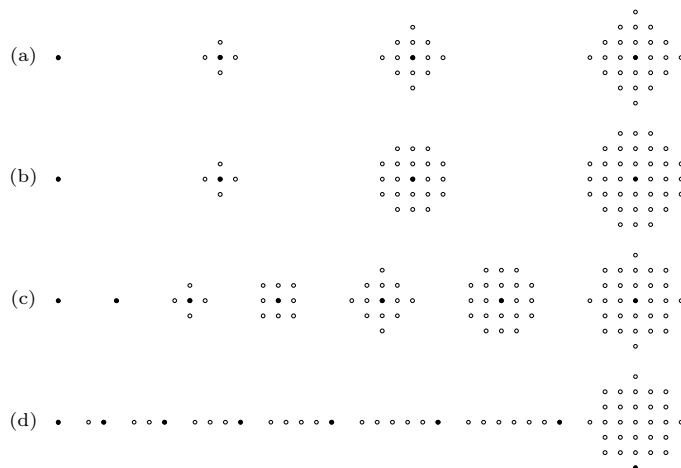


FIGURE 3.11 – Exemples de constructions d'éléments structurants utilisant les distances. a : disques de la distance  $d_4$  (première rangée). b : disques de la distance octogonale (rangée 2). c : disques de la distance de chanfrein  $d_{2,3}$  (rangée 3). d : les segments obtenus par une transformation en distance généralisée sont réunis dans une étape finale pour former un élément structurant arbitraire (dernière rangée)

### 3.2.3 Convexité et polytopes

#### Convexité

**Définition 3.12** (ensemble convexe). Un sous-ensemble  $K$  de  $\mathbb{R}^n$  est convexe si pour tout couple de points  $x$  et  $y$  de  $K$ , le segment  $[x, y]$  est contenu dans  $K$ . Autrement dit :

$$x, y \in K \times K, 0 \leq \lambda \leq 1 \implies \lambda x + (1 - \lambda)y \in K . \quad (3.10)$$

**Définition 3.13** (fonction d'appui). La fonction d'appui d'un convexe  $K$  est la fonction :

$$h(K, \vec{v}) = \sup \{ x \cdot \vec{v} : x \in K \} . \quad (3.11)$$

**Proposition 3.14.** La fonction d'appui de la somme de MINKOWSKI est la somme des fonctions d'appui. Soient  $K_1$  et  $K_2$  sont deux convexes non vides et  $\vec{v}$  un vecteur non nul [voir par exemple : 58].

$$h(K_1 \oplus K_2, \vec{v}) = h(K_1, \vec{v}) + h(K_2, \vec{v}) . \quad (3.12)$$

**Définition 3.15** (hyperplan d'appui). Soit un convexe non vide  $K$  et un vecteur  $\vec{v}$  non nul tel que  $h(K, \vec{v})$  soit fini. On appelle *hyperplan d'appui* de normale extérieure  $\vec{v}$ , l'hyperplan :

$$H(K, \vec{v}) = \{ x \in \mathbb{R}^n : x \cdot \vec{v} = h(K, \vec{v}) \} . \quad (3.13)$$

#### Polytopes

**Définition 3.16** (polyèdre convexe). Un polyèdre convexe est l'intersection d'un nombre fini de demi-plans [par exemple 192].

**Définition 3.17** (polytope). Un polytope est l'enveloppe convexe d'un nombre fini de points.



**Théorème 3.18** (Weyl-Minkowski). Une partie d'un espace euclidien est un polytope si et seulement si elle est un polyèdre convexe fini.

On peut donc représenter un polytope de  $\mathbb{R}^n$  comme l'enveloppe convexe de ses  $k$  sommets ( $\mathcal{V}$ -représentation) ou par un ensemble de  $l$  demi-plans ( $\mathcal{H}$ -représentation) :

$$P = \text{conv}(\{p_i\}_{1 \leq i \leq k}) = \left\{ \sum_{i=1}^k \lambda_i p_i, \lambda_i \geq 0 \text{ et } \sum_{i=1}^k \lambda_i = 1 \right\}, \quad (3.14)$$

$$P = \{x : Ax \leq y\}, \quad (3.15)$$

où  $A$  est une matrice  $l \times n$ ,  $y$  un vecteur de  $n$  valeurs dénommées  $\mathcal{H}$ -coefficients de  $P$ . Soient deux vecteurs  $\vec{u}$  et  $\vec{v}$ , on note  $\vec{u} \leq \vec{v}$  si et seulement si  $\forall i, \vec{u}_i \leq \vec{v}_i$ .

**Définition 3.19** (face). L'intersection d'un polytope  $P$  avec un des ses hyperplans d'appui est appelée *face* de  $P$ . On note :

$$F(P, \vec{v}) = P \cap H(P, \vec{v}). \quad (3.16)$$

Si la dimension de  $P$  est  $n$ , une face de dimension 0 est un *sommet*, une face de dimension 1 est une *arête*, une face de dimension  $n - 1$  est une *facette*.

**Définition 3.20** (polytope discret). Un *polytope discret* est l'intersection d'un polytope  $\mathcal{P}$  de  $\mathbb{R}^n$  avec le réseau de points  $\mathbb{Z}^n$  (discrétisation de GAUSS de  $\mathcal{P}$ ).

**Représentation Minimale** De nombreuses opérations sur les polytopes dans  $\mathbb{R}^n$ , qu'elles s'appuient sur la  $\mathcal{V}$  or  $\mathcal{H}$ -représentation, nécessitent une représentation minimale. La suppression de redondance (*redundancy removal*) consiste à éliminer les points superflus, car intérieurs, de la  $\mathcal{V}$ -représentation, et les inégalités inutiles dans la  $\mathcal{H}$ -représentation. Puisque l'objectif est ici de construire des séquences de polytopes qui possèdent les mêmes directions de facettes, il est préférable de conserver l'intégralité des vecteurs normaux de sorte que les  $\mathcal{H}$ -polytopes partagent la même matrice, bien que certains soient redondants pour les petites tailles.

En revanche, certaines opérations imposeront d'utiliser une  $\mathcal{H}$ -représentation minimale en termes de coefficients.

**Définition 3.21** (Représentation à coefficients minimaux). Une  $\mathcal{H}$ -représentation à coefficients minimaux d'un polytope discret  $P$ , notée  $\hat{\mathcal{H}}$ -représentation est une  $\mathcal{H}$ -représentation  $P = \{x : Ax \leq y\}$  telle que  $y$  est minimal :

$$P = \{x \in \mathbb{Z}^n : Ax \leq y\} \text{ and } \forall i \in [1 \dots l], \exists x \in \mathbb{Z}^n : A_i x = y_i, \quad (3.17)$$

où  $A_i$  désigne la  $i^{\text{e}}$  ligne de la matrice  $A$ . Le vecteur  $\hat{y}$  des coefficients minimaux est donné par la fonction d'appui :

$$\hat{y}_i = h(P, A_i).$$

La figure 3.22 montre deux représentations du même polygone  $P$  dans  $\mathbb{Z}^2$ , une  $\mathcal{H}$ -représentation :

$$\begin{pmatrix} -1 & 1 \\ 0 & -1 \\ 14 & 2 \\ 13 & 5 \\ 11 & 9 \end{pmatrix} \cdot x \leq \begin{pmatrix} 0 \\ 0 \\ 95 \\ 95 \\ 95 \end{pmatrix}$$

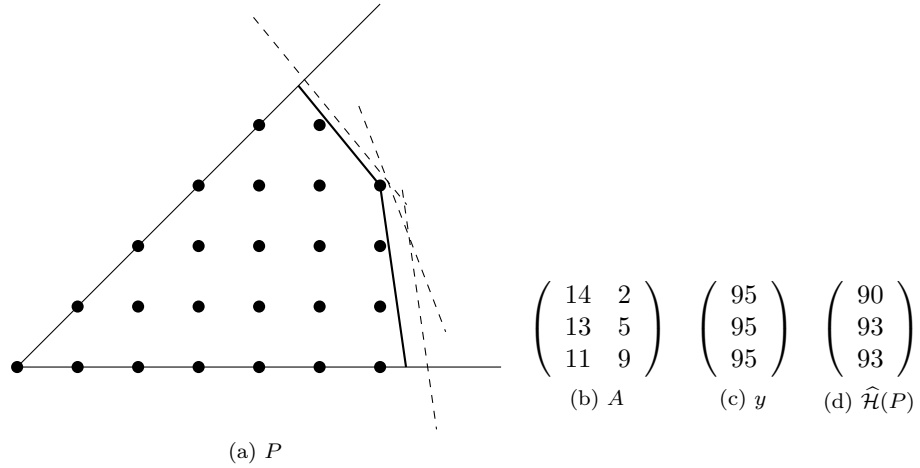


FIGURE 3.22 –  $\mathcal{H}$ -représentations d'un polytope discret  $P$ . Lignes discontinues :  $\mathcal{H}$ -représentation de  $P$ . Lignes continues :  $\widehat{\mathcal{H}}$ -représentation de  $P$ . Dans le cas minimal  $\widehat{\mathcal{H}}$ , trois égalités sont vérifiées pour le point  $(6, 3)$ . Bien que les coefficients soient minimaux, la représentation n'est pas minimale car une des inégalités de la  $\widehat{\mathcal{H}}$ -représentation de  $P$  peut être éliminée.

et la  $\widehat{\mathcal{H}}$ -représentation de  $P$  :

$$\begin{pmatrix} -1 & 1 \\ 0 & -1 \\ 14 & 2 \\ 13 & 5 \\ 11 & 9 \end{pmatrix} \cdot x \leq \begin{pmatrix} 0 \\ 0 \\ 90 \\ 93 \\ 93 \end{pmatrix} .$$

**Somme de MINKOWSKI de  $\mathcal{H}$ -polytopes** Soient  $P$  et  $Q$  sont deux polytopes non vides et  $\vec{v}$  un vecteur non nul. Les polytopes étant des convexes, (3.12) s'applique à  $P$  et  $Q$ . De plus, les faces de  $P \oplus Q$  sont les sommes des faces correspondantes de  $P$  et  $Q$  [58, p. 317] :

$$\forall \vec{v} \neq \vec{0}, F(P \oplus Q, \vec{v}) = F(P, \vec{v}) \oplus F(Q, \vec{v}) . \quad (3.18)$$

Soit le  $\mathcal{H}$ -polytope  $P = \{x : Ax \leq y\}$ . Le translaté de  $P$  par  $\vec{v}$ , qui est aussi la somme de MINKOWSKI de  $P$  et  $\{\vec{v}\}$ , découle directement de (3.12) :

$$(P)_{\vec{v}} = P \oplus \{\vec{v}\} = \{x + \vec{v} : Ax \leq y\} = \{x : Ax \leq y + A\vec{v}\} . \quad (3.19)$$

La translation d'une représentation minimale donne une représentation minimale.

**Test d'inclusion** Soient  $P = \{x : Ax \leq y\}$  et  $Q = \{x : Ax \leq z\}$  deux polytopes représentés par la même matrice  $A$  et les vecteurs de coefficients  $y$  et  $z$ .  $P$  est un sous-ensemble de  $Q$  si  $y \leq z$  (condition suffisante) :

$$y \leq z \implies P \subseteq Q . \quad (3.20)$$

De plus, si  $y$  est représenté par ses coefficients minimaux, la condition est aussi nécessaire :

$$\hat{y} \leq z \iff P \subseteq Q . \quad (3.21)$$

### 3.3 Transformation en distance à latence minimale

Dans [185], X. WANG et G. BERTRAND proposent une distance généralisée asymétrique avec des vecteurs de déplacements et un ordre de balayage tels que quand le point  $p$  de l'image est visité tous ses voisins l'ont déjà été. L'algorithme nécessite donc un seul balayage. Ils étendent ensuite ce résultat à une séquence où deux voisinages avec la même condition de causalité sont alternés strictement (*i.e.*,  $B = (1, 2)$ ) [186].

Dans la suite, nous introduisons une méthode qui calcule une transformation en distance en un seul balayage avec un nombre de voisinages quelconques et une séquence quelconque, périodique ou non. Elle peut être utilisée pour extraire une version translatée (afin de respecter la condition de causalité sur les vecteurs de déplacement) de la transformation en distance à séquence de voisinages en 2D et 3D. Le coût de calcul, bien que dépendant du nombre de vecteurs de déplacements, est constant par pixel, autrement dit la complexité temporelle est linéaire avec le nombre de pixels, ce qui est optimal. La complexité spatiale est très faible car seule une ligne d'image en 2D ou un plan dans le cas 3D est nécessaire. Cet algorithme peut être utilisé sur des images très grandes en hauteur, voire de hauteur infinie (par exemple acquises continûment), et fournit une ligne de résultat pour chaque ligne d'image en entrée.

L'algorithme qui permet de recentrer ce résultat sera également donné. Sa complexité spatiale est supérieure car il doit repositionner les centres de disque en arrière dans l'image. Cependant, il ne rompt pas le flot général des données : les images sont toujours fournies ligne à ligne. Il n'y a donc pas non nécessité de stocker une quelconque image intermédiaire. Nous verrons dans le chapitre 5 un exemple d'application qui utilise directement cette transformation en distance translatée.

#### 3.3.1 Transformation en distance généralisée

**Proposition 3.23.** La transformation en distance d'une image  $X$  avec la distance induite par le voisinage  $\mathcal{N}$  et les coûts de déplacement  $C_{\vec{v}}$  est telle que :

$$\text{DT}_X(p) = \begin{cases} 0 & \text{si } p \notin X \\ \min \{ \hat{C}_{\vec{v}}(\text{DT}_X(p - \vec{v})), \vec{v} \in \mathcal{N}^* \} & \text{sinon} \end{cases} . \quad (3.22)$$

où  $\hat{C}_{\vec{v}}$  représente les déplacements absolus minimaux correspondant à  $C_{\vec{v}}$  (définition 2.25) et  $\mathcal{N}^* = \mathcal{N} \setminus \vec{0}$ , l'ensemble des vecteurs de déplacements non nuls.

*Démonstration.* Le cas  $p \notin X$  découle immédiatement des définitions 3.1 et 2.24. Supposons maintenant que  $p \in X$  donc tout chemin de  $q \notin X$  vers  $p$  possède au moins un vecteur de déplacement. La proposition 2.26 établit que les distances induites par les coûts  $(\{\vec{v}_i\}, C_{\vec{v}_i})$  et les coûts minimaux  $(\{\vec{v}_i\}, \hat{C}_{\vec{v}_i})$  sont égales. Nous pouvons donc admettre sans perte de généralité que les coûts utilisés sont des coûts minimaux pour lesquels la proposition 2.32 est vérifiée. Donc si  $P = (q = p_0, \dots, p_n = p - \vec{v})$  est un chemin minimal de  $q$  vers  $p - \vec{v}$  alors  $P' = (q = p_0, \dots, p_n, p + \vec{v})$  possède un coût minimal — parmi les chemins de  $q$  vers  $p$  où l'avant-dernier point est  $p - \vec{v}$  — égal à  $\hat{C}_{\vec{v}}(\mathcal{L}(P))$ .  $\hat{C}_{\vec{v}}(\text{DT}_X(p - \vec{v}))$  est donc la plus courte distance du point  $q \notin X$  vers  $p$  via  $p - \vec{v}$ . Les chemins dont le dernier déplacement  $\vec{v}$  n'est pas dans  $\mathcal{N}$  ont un coût infini et ne peuvent donc pas être minimaux, ce qui prouve (3.22).  $\square$

Quand tous les vecteurs de  $\mathcal{N}^*$  sont compatibles avec le sens de balayage (condition de causalité), (3.22) propage les chemins depuis le fond en un balayage. Une transformation en distance généralisée avec des voisinages  $\mathcal{N}_1 \dots \mathcal{N}_n$  quel que soit leur nombre, sélectionnés par une

séquence  $B, B(i) \in [1, n]$  découle directement de (2.17, 3.22) et les coûts minimaux sont donnés par :

$$\hat{C}_{\vec{v}}(r) = \min \{s : s > r \text{ et } \vec{v} \in \mathcal{N}_{B(s)}\}. \quad (3.23)$$

Soit  $\chi_{\vec{v}}(r)$  la fonction caractéristique de l'ensemble  $\mathcal{N}_{B(r)}$  (i.e.,  $\chi_{\vec{v}}(r) = 1$  si  $\vec{v} \in \mathcal{N}_{B(r)}$ ; 0 sinon) et  $\chi_{\vec{v}}^{\Sigma}(r)$  sa somme cumulée ( $\chi_{\vec{v}}^{\Sigma}(r) = \sum_{s \leq r} \chi_{\vec{v}}(s)$ ). Alors selon la proposition 2.17 :

$$\hat{C}_{\vec{v}}(r) = [\chi_{\vec{v}}^{\Sigma}]^{\dagger}(\chi_{\vec{v}}^{\Sigma}(r) + 1) + 1. \quad (3.24)$$

L'algorithme 3.24 produit une transformation en distance généralisée avec une séquence quelconque de voisins causaux ( $\mathcal{N}$  est leur union), avec les coûts de déplacement donnés par (3.24). Cet algorithme a été présenté pour la première fois en 2003 pour la décomposition de polygones convexes [99].

---

**Algorithme 3.24:** Transformation en distance généralisée asymétrique calculée en un balayage

---

**Données :**  $X$  : un ensemble de points

**Données :**  $\mathcal{N}$  : voisinage causal

**Données :**  $\hat{C}_{\vec{v}}$  : coûts absolus minimaux de déplacement

**Résultat :**  $DT_X$  : transformation en distance généralisée de  $X$

**pour chaque**  $p$  *dans le domaine de DT*, *dans un ordre de balayage compatible avec  $\mathcal{N}$*  **faire**

**si**  $p \notin X$  **alors**

    |  $DT_X(p) \leftarrow 0$

**sinon**

    |  $l \leftarrow \infty$

**pour chaque**  $\vec{v}$  *dans  $\mathcal{N}$*  **faire**

      |  $l \leftarrow \min \{l; \hat{C}_{\vec{v}}(DT_X(p - \vec{v}))\}$

**fin**

    |  $DT_X(p) \leftarrow l$

**fin**

**fin**

---

### 3.3.2 Distances à séquence de voisinages

La séquence des boules d'une distance à séquence de voisinages induite par la séquence  $B$  est produite par la somme de MINKOWSKI itérative des voisinages :

$$D_{\leq}(p, 0) = \{p\}, \quad D_{\leq}(p, r) = D_{\leq}(p, r - 1) \oplus \mathcal{N}_{B(r)}.$$

La notation  $\mathbf{1}_B(r)$  pour  $\mathcal{N}_1$ , plus généralement,  $\mathbf{j}_B(r)$  pour  $\mathcal{N}_j$ , est utilisée pour compter le nombre d'apparitions d'un voisinage dans la séquence  $B$  jusqu'au rayon  $r$  :

$$\mathbf{j}_B(r) = \text{Card}(\{i : b(i) = j, 1 \leq i \leq r\}).$$

Nous appliquons à chaque voisinage  $\mathcal{N}_j$  une translation de vecteur  $\vec{t}_j$  de façon à rendre le voisinage translaté causal  $\mathcal{N}'_j = \mathcal{N}_j \oplus \{\vec{t}_j\}$ . Dans un ordre de balayage invariant aux translations,

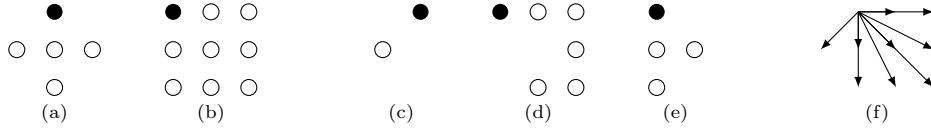


FIGURE 3.25 – Voisinage utilisés pour la transformation en distance à séquence de voisinages asymétrique. (a), et (b) sont respectivement les voisinages translattés de type 1 et 2,  $\mathcal{N}'_1$  et  $\mathcal{N}'_2$ . (c), (d) et (e) sont respectivement  $\mathcal{N}'_1 \setminus \mathcal{N}'_2$ ,  $\mathcal{N}'_2 \setminus \mathcal{N}'_1$  et  $\mathcal{N}'_1 \cap \mathcal{N}'_2$ , chaque ensemble est associé à des coûts de déplacement différents. (f) est l'ensemble de tous les vecteurs de déplacement,  $\mathcal{N}'_1 \cup \mathcal{N}'_2$ , utilisés.

$\vec{t}_j$  déplace le premier point visité de  $\mathcal{N}_j$  vers l'origine  $O$ . Pour un ordre de balayage classique en  $nD$  :

$$\vec{t}_j = (\underbrace{0, \dots, 0}_{n-j}, \underbrace{1, \dots, 1}_j) \quad (3.25)$$

Les voisinages translattés  $\mathcal{N}'_1$  et  $\mathcal{N}'_2$  obtenus avec  $\vec{t}_1 = (0, 1)$  et  $\vec{t}_2 = (1, 1)$  sont illustrés sur la figure 3.25a et figure 3.25b. Les vecteurs de  $\mathcal{N}'_1 \cap \mathcal{N}'_2$  sont utilisés pour tous les rayons  $r$  donc  $\forall \vec{v} \in \mathcal{N}'_1 \cap \mathcal{N}'_2, \chi_{\vec{v}}(r) = 1$  et  $\chi_{\vec{v}}^{\Sigma}(r) = r$ .  $\mathbf{1}_B$ , respectivement  $\mathbf{2}_B$ , est la somme cumulée de la fonction caractéristique de  $\mathcal{N}_{B(r)}$  pour un vecteur de  $\mathcal{N}'_1 \setminus \mathcal{N}'_2$ , respectivement  $\mathcal{N}'_2 \setminus \mathcal{N}'_1$ . On a donc, d'après (3.24) :

$$\hat{C}_{\vec{v}}(r) = \begin{cases} \hat{C}_{\vec{v}}^1(r) = \mathbf{1}_B^\dagger(\mathbf{1}_B(r) + 1) + 1 & \text{if } \vec{v} \in \mathcal{N}'_1 \text{ et } \vec{v} \notin \mathcal{N}'_2 \\ \hat{C}_{\vec{v}}^2(r) = \mathbf{2}_B^\dagger(\mathbf{2}_B(r) + 1) + 1 & \text{if } \vec{v} \notin \mathcal{N}'_1 \text{ et } \vec{v} \in \mathcal{N}'_2 \\ \hat{C}_{\vec{v}}^{12}(r) = r + 1 & \text{if } \vec{v} \in \mathcal{N}'_1 \text{ et } \vec{v} \in \mathcal{N}'_2 \end{cases}$$

Les fonctions  $\mathbf{1}_B$  et  $\mathbf{2}_B$ , et par conséquent, les coûts de déplacement sont liés à la façon de définir la séquence de voisinage  $B$ .

**Séquence périodique.** Si  $B$  est une distance périodique, les coûts minimaux relatifs  $\hat{c}_{\vec{v}}$  sont également des séquences périodiques. Prenons par exemple la séquence périodique de la distance octogonale,  $B = (\overline{1, 2})$ , alors  $\mathbf{1}_B(r)_{r \geq 0} = (0, 1, 1, 2, \dots)$ ,  $\mathbf{1}_B^\dagger(r)_{r > 0} = (0, 2, 4, \dots)$ ,  $\hat{C}_{\vec{v}}^1(r)_{r \geq 0} = (1, 3, 3, 5, \dots)$  et  $\hat{c}_{\vec{v}}^1(r)_{r \geq 0} = (1, 2, 1, 2, \dots)$ . De la même façon,  $\mathbf{2}_B(r)_{r \geq 0} = (0, 0, 1, 1, 2, \dots)$ ,  $\mathbf{2}_B^\dagger(r)_{r > 0} = (1, 3, \dots)$ ,  $\hat{C}_{\vec{v}}^2(r)_{r \geq 0} = (2, 2, 4, \dots)$  et  $\hat{c}_{\vec{v}}^2(r)_{r \geq 0} = (2, 1, 2, 1, \dots)$ .

**Séquence de BEATTY.** Supposons maintenant que la séquence de voisinages soit définie par une séquence de BEATTY (comme le proposent A. HAJDU et L. HAJDU [72]) ou une séquence de BEATTY rationnelle :  $B(r) = \lfloor \tau r \rfloor - \lfloor \tau(r-1) \rfloor$ , avec  $\tau \in [1, 2]$  de sorte que  $B(r) \in \{1, 2\}$ .  $\mathbf{1}_B$  et  $\mathbf{2}_B$  sont respectivement les sommes cumulées de  $2 - B(r) = \lceil (2 - \tau)r \rceil - \lceil (2 - \tau)(r-1) \rceil$  et  $B(r) - 1 = \lfloor (\tau - 1)r \rfloor - \lfloor (\tau - 1)(r-1) \rfloor$ . Alors  $\mathbf{1}_B(r) = \lceil (2 - \tau)r \rceil$ ,  $\mathbf{2}_B(r) = \lfloor (\tau - 1)r \rfloor$ ,  $\mathbf{1}_B^\dagger(r) = \lfloor \frac{r-1}{2-\tau} \rfloor$  et  $\mathbf{2}_B^\dagger(r) = \lceil \frac{r}{\tau-1} - 1 \rceil$ . Ceci permet de calculer  $\hat{C}_{\vec{v}}^1$  et  $\hat{C}_{\vec{v}}^2$  à la volée. Pour la distance octogonale,  $\tau = 3/2$ ,  $\mathbf{1}_B(r) = \lceil r/2 \rceil$ ,  $\mathbf{2}_B(r) = \lfloor r/2 \rfloor$ ,  $\mathbf{1}_B^\dagger(r) = 2r - 2$  and  $\mathbf{2}_B^\dagger(r) = 2r - 1$ .

Un exemple produit par l'algorithme 3.24 pour la distance octogonale asymétrique est présenté sur la figure 3.26b. Les coûts de déplacement sont obtenus indifféremment par la séquence périodique  $B = (1, 2)$  ou le taux  $\tau = 3/2$ .

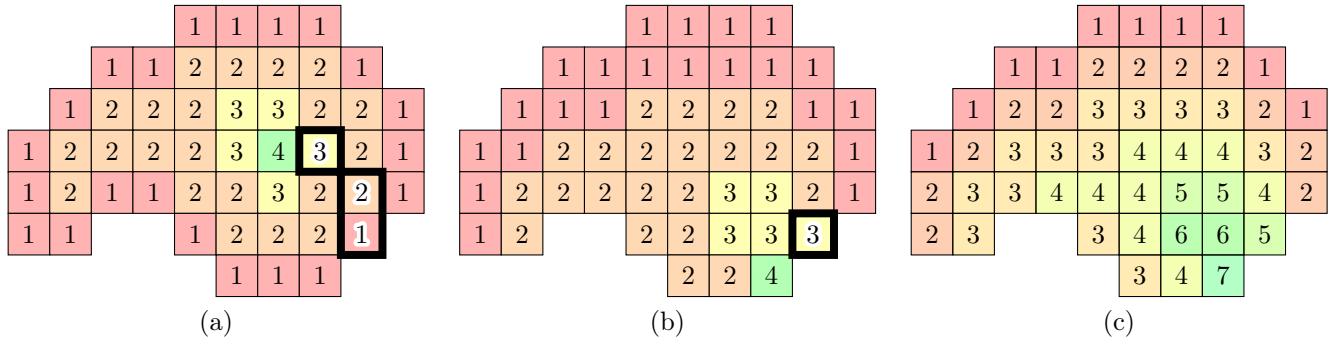


FIGURE 3.26 – (a) Transformation en distance octogonale. (b) Transformation en distance octogonale asymétrique. Les points entourés (a) sont des centres de disques tous translatés au même point (b) avec la valeur 3. (c) Transformation en distance octogonale asymétrique avec des disques intermédiaires.

### 3.3.3 Recentrage de la transformation en distance asymétrique

Soit  $\{\vec{t}(r), r \in \mathbb{N}^*\}$  une séquence de vecteurs de translation tels que les disques translatés  $D'(p, r) = D_{\leq}(p + \vec{t}(r), r)$  et  $\check{D}'(p, r) = \check{D}(p - \vec{t}(r), r)$  sont croissants par inclusion ensembliste. Pour une séquence de disques produits par des voisinages translatés définis par (3.25), les vecteurs de translations sont :

$$\begin{aligned} \vec{t}(r) &= \vec{t}(r-1) + \overrightarrow{t_{B(r)}} \\ &= \sum_j \mathbf{j}_B(r) \vec{t}_j \\ &= \left( \sum_{j=n}^n \mathbf{j}_B(r), \dots, \sum_{j=1}^n \mathbf{j}_B(r) \right) \end{aligned}$$

En particulier en 2D :

$$\vec{t}(r) = (\mathbf{2}_B(r), \mathbf{1}_B(r) + \mathbf{2}_B(r)) = (\mathbf{2}_B(r), r). \quad (3.26)$$

Les valeurs de  $\text{DT}'_X$  ont des équivalences avec les valeurs de  $\text{DT}_X$  :

$$\begin{aligned} \text{DT}_X(p) \geq r &\iff \check{D}(p, r-1) \subseteq X \\ &\iff \check{D}'(p + \vec{t}(r-1), r-1) \subseteq X \\ &\iff \text{DT}'_X(p + \vec{t}(r-1)) \geq r. \end{aligned} \quad (3.27)$$

Donc :

$$\begin{aligned} \text{DT}_X(p) = r &\iff \text{DT}_X(p) \geq r \text{ and } \text{DT}_X(p) < r+1 \\ &\iff \text{DT}'_X(p + \vec{t}(r)) \leq r \leq \text{DT}'_X(p + \vec{t}(r-1)). \end{aligned} \quad (3.28)$$

Connaissant  $\text{DT}'_X(p)$  et  $\text{DT}'_X(p + \vec{t})$ , on peut déduire les valeurs de  $\text{DT}_X(p - \vec{t}(r-1))$  pour tous les rayons  $r$  entre  $\text{DT}'_X(p + \vec{t})$  et  $\text{DT}'_X(p)$  pour lesquels  $\vec{t}(r) = \vec{t}(r-1) + \vec{t}$ , i.e.,  $\vec{t} = t_{B(r)}$ . L'algorithme 3.32 restaure les rayons  $r$  des disques symétriques en sélectionnant tous les  $r$  dans

$[DT'_X(p + \vec{t}_j), DT'_X(p)]$  tels que  $B(r) = j$ . Itérer les valeurs de  $r$  pour lesquelles  $B(r) = j$  utilise la proposition 2.17 de l'inverse de LAMBEK-MOSER. Les valeurs de  $DT'_X$  deviennent disponibles avant que l'image soit entièrement traitée. Par exemple, dans un balayage standard, dès que la ligne  $y$  est traitée, toutes les lignes de  $DT'_X$  au-dessus de  $y - r_{\max}$  sont pleinement reconstruites ( $r_{\max}$  est la valeur maximale de  $DT'$  sur la ligne  $y$ ).

### 3.3.4 Disques intermédiaires

L'ensemble complet des voisins utilisés dans l'algorithme 3.24 est la réunion de tous les vecteurs de déplacements qui ont des coûts finis (à l'exception du déplacement nul). En 2D, cet ensemble comporte 9 vecteurs, ce qui est du même ordre que les 8 vecteurs utilisés par les algorithmes classiques. Ce nombre augmente rapidement avec la dimension : 32, 107, 350 respectivement pour les dimensions 3, 4 and 5. La suite semble suivre avec un décalage constant de  $-1$ , la séquence A126184 de l'encyclopédie en ligne des séquences de nombres entiers de SLOANE et PLOUFFE [44]). Dans cette section, nous allons montrer comment réduire fortement le nombre de vecteurs de déplacement en décomposant les voisinages par des unions ensemblistes et des translations.

Soient  $B$  une séquence de valeurs dans  $[1, n]$  et  $c_{\vec{v}}^j, j \in [1, n]$  les  $n$  sous-séquences de longueur finies correspondant à la construction des  $n$  ensembles  $\mathcal{N}_j, j \in [1, n]$ . Par simplicité, mais sans perdre en généralité, nous supposons que toutes les sous-séquences ont la même longueur  $l$ . On construit une séquence étendue des coûts relatifs du vecteur  $\vec{v}$  par la concaténation des sous-séquences  $c_{\vec{v}}^1 \dots c_{\vec{v}}^n$  dans un ordre déterminé par la séquence principale  $B$  :

$$\forall \vec{v}, \forall r > 0, c_{\vec{v}}(r) = c_{\vec{v}}^{B(\lfloor r/l \rfloor + 1)}(\langle r \rangle_l)$$

où  $\langle r \rangle_l$  est le reste de la division entière de  $r$  par  $l$ .

Par la proposition 2.44, les boules engendrées par ces déplacements sont tels que :

$$\forall k > 0, D_{\leq}(O, kl) = D_{\leq}(O, (k-1)l) \oplus \mathcal{N}_{B(k)} .$$

La  $i^{\text{e}}$  sous-séquence occupe les rayons  $(i-1)l$  à  $il-1$  dans la séquence étendue et inversement, le rayon  $r$  correspond à la sous-séquence d'indice  $i = \lfloor r/l \rfloor + 1$  qui commence au rayon  $r - \langle r \rangle_l$ . Un coût minimal pour le rayon  $r$ , nécessaires au calcul de la transformation en distance, peut être égal au coût minimal dans la sous-séquence correspondante quand celui-ci est fini ou déduit de la valeur initiale d'une sous-séquence qui suit. La quantité  $\hat{C}_{\vec{v}}(r)$  est soit la valeur à l'indice  $\langle r \rangle_l$  dans la sous-séquence  $i : \hat{c}_{\vec{v}}^{B(i)}(\langle r \rangle_l) + r$ , ou à la valeur initiale de la prochaine sous-séquence  $\hat{c}_{\vec{v}}^j$  où le voisinage  $\vec{v}$  est utilisé :  $\hat{c}_{\vec{v}}^j(0) + l(\mathbf{j}_B^\dagger(\mathbf{j}_B(i) + 1))$ , selon le minimum obtenu conformément à la définition 2.25.

$$\hat{C}_{\vec{v}}(r) = \min \left\{ \hat{c}_{\vec{v}}^{B(i)}(\langle r \rangle_l) + r; \right. \quad (3.29a)$$

$$\left. \hat{c}_{\vec{v}}^j(0) + l\mathbf{j}_B^\dagger(\mathbf{j}_B(i) + 1), j \in [1, n] \right\} \quad (3.29b)$$

où  $i = \lfloor r/l \rfloor + 1$ .

Clairement, si un vecteur  $\vec{v}$  n'est pas utilisé dans la décomposition du voisinage  $j$ , alors  $\hat{c}_{\vec{v}}^j$  est infini et le vecteur peut être omis de (3.29b). Inversement, quand un vecteur est utilisé dans tous les voisinages, (3.29) se simplifie en :

$$\forall j, \hat{c}_{\vec{v}}^j(0) \neq \infty \Rightarrow \hat{C}_{\vec{v}}(r) = \min \left\{ \hat{c}_{\vec{v}}^{B(i)}(\langle r \rangle_l) + r; c_{\vec{v}}^{B(i+1)}(0) + li, j \in [1, n] \right\} \quad (3.30)$$

Une simplification supplémentaire est possible lorsque les coûts de déplacement sont toujours finis pour un vecteur  $\vec{v}$ , on peut alors omettre (3.29b) :

$$\forall j, \hat{c}_{\vec{v}}^j(l) \neq \infty \Rightarrow \forall r \geq 1, \hat{C}_{\vec{v}}(r) = \hat{c}_{\vec{v}}^{B(i)}(\langle r \rangle_l) + r \quad (3.31)$$

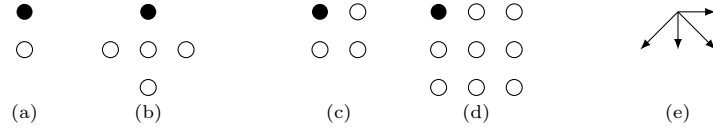


FIGURE 3.27 – Décomposition des voisinages utilisés pour la transformation en distance à séquence de voisinages asymétrique. (b), et (d) sont respectivement les voisinages translattés de types 1 et 2,  $\mathcal{N}'_1$  et  $\mathcal{N}'_2$ . (a) et (c) sont les disques intermédiaires. Seuls 4 vecteurs de déplacement (e) sont nécessaires au lieu de 9 sans cette décomposition.

$r$	1	2	$r$	1	2
$\hat{S}^1_{\leq, (1,0)}(r)$	0	0	$\hat{S}^2_{\leq, (1,0)}(r)$	1	2
$\hat{S}^1_{\leq, (-1,1)}(r)$	0	1	$\hat{S}^2_{\leq, (-1,1)}(r)$	0	0
$\hat{S}^1_{\leq, (0,1)}(r)$	1	2	$\hat{S}^2_{\leq, (0,1)}(r)$	1	2
$\hat{S}^1_{\leq, (1,1)}(r)$	0	1	$\hat{S}^2_{\leq, (1,1)}(r)$	1	2

Tableau 3.28 – Tables de construction pour la décomposition des deux voisinages  $\mathcal{N}'_1$  et  $\mathcal{N}'_2$  en 2D.

Si  $B$  est périodique, alors  $\hat{c}_{\vec{T}}$  l'est aussi avec une période multipliée par  $l$ . Avec (3.29), on peut éviter le recours à l'expansion effective de la séquence  $B$  par concaténation de sous-séquences. Alternativement,  $\hat{C}_{\vec{T}}$  peut être calculé à la volée pour les séquences longues ou, *a fortiori* les séquences apériodiques. Cependant, pour de courtes séquences il peut être souhaitable de précalculer  $\hat{c}_{\vec{T}}$  une fois pour toutes.

### Cas 2D

Considérons la décomposition suivantes des voisinages en 2D :

$$\begin{aligned}
D^1_{\leq}(O, 1) &= D^1_{\leq}(O, 0) \cup D^1_{\leq}((0, 1), 0), \\
D^1_{\leq}(O, 2) &= D^1_{\leq}(O, 1) \cup D^1_{\leq}((0, 1), 1) \cup D^1_{\leq}((-1, 1), 0) \cup D^1_{\leq}((1, 1), 0), \\
D^2_{\leq}(O, 1) &= D^2_{\leq}(O, 0) \cup D^2_{\leq}((0, 1), 0) \cup D^2_{\leq}((1, 0), 0) \cup D^2_{\leq}((1, 1), 0), \\
D^2_{\leq}(O, 2) &= D^2_{\leq}(O, 1) \cup D^2_{\leq}((0, 1), 1) \cup D^2_{\leq}((1, 0), 1) \cup D^2_{\leq}((1, 1), 1).
\end{aligned}$$

où  $D^1_{\leq}(O, 2)$  et  $D^2_{\leq}(O, 2)$  sont égaux aux voisinages translattés  $\mathcal{N}'_1$  et  $\mathcal{N}'_2$ . Les disques intermédiaires  $D^1_{\leq}(\hat{O}, 1)$  et  $D^2_{\leq}(\hat{O}, 1)$  sont illustrés sur la figure 3.27a et la figure 3.27c. Cette décomposition se résume aux valeurs de construction  $\hat{S}^1_{\leq}$  et  $\hat{S}^2_{\leq}$ , telles qu'elles sont définies dans (2.39), et montrées dans le tableau 3.28 ( $\hat{S}^1_{\leq, (0,0)}$  est omis). Notons que (3.30) est valide pour les vecteurs  $(1, 1)$  et  $(0, 1)$  de même que (3.31) pour le vecteur  $(0, 1)$ .

Ces deux séquences permettent d'obtenir n'importe quelle combinaison de sommes de MINKWOSKI de  $\mathcal{N}_1$  et  $\mathcal{N}_2$ . Prenons par exemple la distance octogonale, c'est-à-dire la séquence de voisinages  $B = (\bar{1}, 2)$ . L'application directe de (3.29) donne les valeurs  $\hat{c}_{\vec{T}}$  présentées dans le tableau 3.29c. Un exemple de transformation en distance translattée  $DT''$  issu de ces valeurs est montré dans la figure 3.26c.

Revenir à  $DT'$  à partir de  $DT''$  est très simple lorsque toutes les sous-séquences partagent la même longueur  $l$  :

$$DT'_X(p) = \left\lfloor \frac{DT''_X(p) + l - 1}{l} \right\rfloor.$$



$r$	0	1
$c_{(1,0)}^1(r)$	$\infty$	$\infty$
$c_{(-1,1)}^1(r)$	2	$\infty$
$c_{(0,1)}^1(r)$	1	1
$c_{(1,1)}^1(r)$	2	$\infty$

(a)

$r$	0	1
$c_{(1,0)}^2(r)$	1	1
$c_{(-1,1)}^2(r)$	$\infty$	$\infty$
$c_{(0,1)}^2(r)$	1	1
$c_{(1,1)}^2(r)$	1	1

(b)

$r$	0	1	2	3
$\hat{c}_{(1,0)}(r)$	3	2	1	1
$\hat{c}_{(-1,1)}(r)$	2	5	4	3
$\hat{c}_{(0,1)}(r)$	1	1	1	1
$\hat{c}_{(1,1)}(r)$	2	2	1	1

(c)

Tableau 3.29 – Coûts de déplacements pour la décomposition de  $\mathcal{N}'_1$  and  $\mathcal{N}'_2$ , (a) and (b). (c) : coûts minimaux relatifs pour la concaténation alternée de (a) et (b) qui engendre la transformation en distance octogonale asymétrique avec des disques intermédiaires.

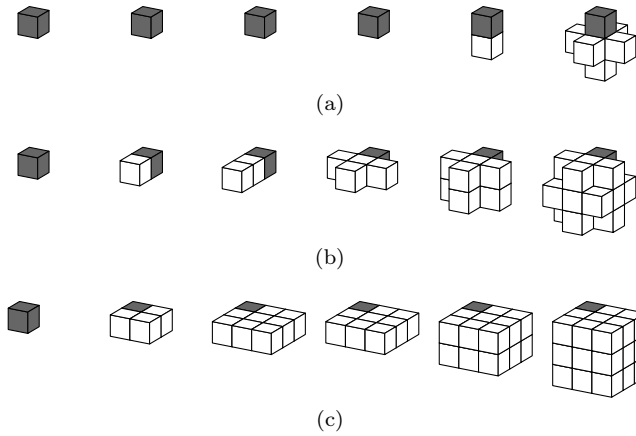












FIGURE 3.30 – Décomposition des voisinages pour la transformation en distance tradatée d'une distance à séquence de voisinages en 3D. De haut en bas,  $\mathcal{N}'_1$ ,  $\mathcal{N}'_2$  et  $\mathcal{N}'_3$ . De gauche à droite, les 6 étapes de la décomposition.

### Cas 3D

La même méthode s'applique en 3D : à partir d'une décomposition des voisinages, et une séquence  $B$ , une séquence étendue est construite pour chacun des vecteurs de déplacement. La figure 3.30 montre une décomposition possible des voisinages 3D qui utilise uniquement 9 vecteurs.

Tableau 3.31 – Indices de construction pour la décomposition des voisinages en 3D. Seules les valeurs non nulles sont montrées.

$r$		$\mathcal{N}'_1$					$\mathcal{N}'_2$					$\mathcal{N}'_3$				
		1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
(0, 0, 1)					4	5				4	5				4	5
(0, -1, 1)						4										
(0, 1, 1)						4										
(-1, 0, 1)						4					3					
(1, 0, 1)						4					3					
(0, 1, 0)							1	2				1	2			
(1, 0, 0)												1	2			
(1, 1, 0)									1			1	2			
(-1, 1, 0)									1							

---

**Algorithme 3.32:** Calcul de la transformation en distance centrée DT à partir de la transformation en distance asymétrique DT'.

---

**Données :**  $DT'_X$  : transformation en distance translétée de  $X$

**Résultat :**  $DT_X$  : transformation en distance centrée de  $X$

**pour chaque**  $p$  **dans le domaine de**  $DT'$  **dans l'ordre naturel faire**

**si**  $DT'(p) = 0$  **alors**

$DT(p) \leftarrow 0$

**sinon**

**pour chaque**  $j$  **faire**

$r \leftarrow \max \{1; DT'(p + \vec{t}_j)\}$

$r \leftarrow \mathbf{j}_B^\dagger(\mathbf{j}_B(r)) + 1$

            //  $r$  minimal tel que  $r \geq DT'(p - \vec{t}_j)$  et  $B(r) = j$

**tant que**  $r \leq DT'(p)$  **faire**

$DT(p - \vec{t}(r-1)) \leftarrow r$

$r \leftarrow \mathbf{j}_B^\dagger(\mathbf{j}_B(r) + 1) + 1$

                // Prochain  $r$  tel que  $B(r) = j$

**fin**

**fin**

**fin**

**fin**

---

## 3.4 Décomposition d'éléments structurants convexes

Dans la section 2.4.1, nous avons établi le lien entre une construction itérative de disques et une pseudo-distance et précisé le mode de calcul des coûts de déplacement à partir des relations d'inclusion de disques. Ces coûts sont directement exploitables dans l'algorithme 3.24 de transformation en distance généralisée présenté dans la section 3.3.1.

Partant d'un élément structurant convexe  $B$  donné, nous proposons maintenant de déterminer les paramètres d'une pseudo-distance dont un des disques est  $B$ . Calculer l'érosion d'un ensemble de points  $X$  avec  $B$  consistera à calculer la transformation en distance généralisée de  $X$  et à en effectuer la binarisation par seuillage avec le rayon de ce disque (3.8). Pour y parvenir, il est nécessaire d'établir un ensemble de vecteurs de déplacement et les indices de construction qui définissent la façon dont les disques sont itérativement assemblés.

La complexité de l'opération d'érosion est linéaire avec le nombre de pixels. De plus, le balayage est unique si tous les vecteurs de déplacements sont compatibles avec celui-ci. La complexité spatiale est réduite à la mémorisation d'un nombre de lignes d'image égal à la hauteur du plus grand vecteur de déplacement. Contrairement à d'autres méthodes utilisant aussi des transformations en distance généralisée pour des formes de disques particulières, la seule contrainte imposée ici est la convexité de l'élément structurant.

La structure séquentielle de l'algorithme avec un calcul à la volée, ligne par ligne, permet de l'utiliser dans une chaîne de traitements où chaque étage réalise une opération morphologique (par exemple, deux étages pour une ouverture ensembliste). Nous avons conçu et réalisé plusieurs prototypes à base de composants configurables *Field-Programmable Gate Array* (FPGA), par exemple avec une carte de prototypage qui embarque un FPGA Xilinx Spartan IIE alimenté en données par un signal vidéo PAL. Le FPGA gère la synchronisation du signal vidéo et le régénère en sortie. Au moins 8 étages utilisant des éléments structurants différents peuvent être réalisés et utilisés à la cadence vidéo sans ressources supplémentaires (seule la mémoire intégrée est utilisée). La latence correspond à une fraction de pixel à chaque étage, un délai peut être introduit pour simuler une translation du centre de l'élément structurant.

### 3.4.1 Algorithme

La méthode de décomposition d'élément structurant présentée ici évoque la croissance anisotrope d'un cristal au cours de laquelle des couches épitaxiales d'atomes sont déposées successivement sur un germe de cristal. La forme du cristal est influencée par les propriétés physiques des atomes qui contraignent les orientations des plans du cristal qui demeurent constantes durant toute la phase de croissance et la vitesse de dépôt qui peut varier d'une orientation à une autre.

La méthode permet de décomposer n'importe quel ensemble discret borné et convexe. Est considérée ici comme partie convexe de  $\mathbb{Z}^2$ , toute intersection d'un convexe de  $\mathbb{R}^2$  avec  $\mathbb{Z}^2$  sans condition de connexité. Un ensemble de points de  $\mathbb{Z}^2$  est donc convexe s'il est la discrétisation de GAUSS d'un polytope de  $\mathbb{R}^2$ , autrement dit, si et seulement si :

$$B = \text{conv}(B) \cap \mathbb{Z}^2 ,$$

$\text{conv}(B)$ , également noté  $\mathcal{P}$ , étant le plus petit polytope contenant tous les points de  $B$ . La figure 3.34 montre l'élément structurant convexe  $B$  qui servira d'exemple dans la suite et son enveloppe convexe polygonale  $\mathcal{P}$ .

Un ensemble discret convexe  $B$  étant donné, la méthode consiste à construire itérativement une séquence de disques  $D_{<}(O, r)$  selon la méthode décrite dans la section 2.4.1 et telle que pour un rayon  $r_{\max}$ ,  $D_{<}(O, r_{\max}) = B$ . Les résultats de l'algorithme sont donc constitués des paramètres de la décomposition, c'est-à-dire d'un ensemble de vecteurs de déplacements  $\mathcal{N} = \{\vec{v}_k\}$  et des

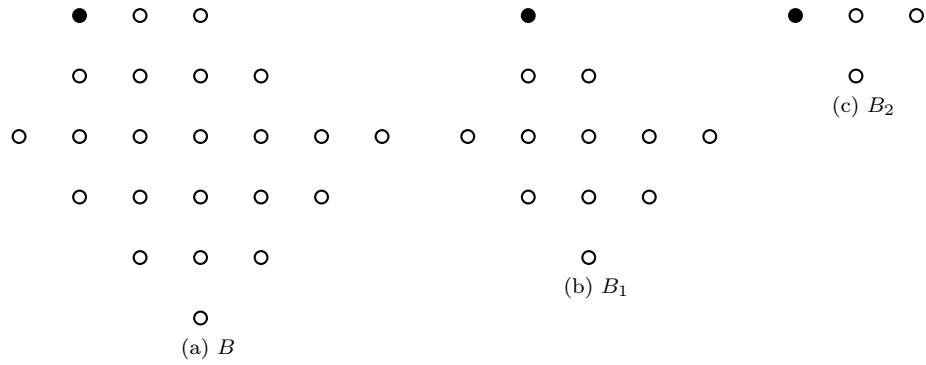


FIGURE 3.33 – Ensemble convexe discret  $B$  et les éléments de sa décomposition en somme de MINKOWSKI,  $B = B_1 \oplus B_2$ .

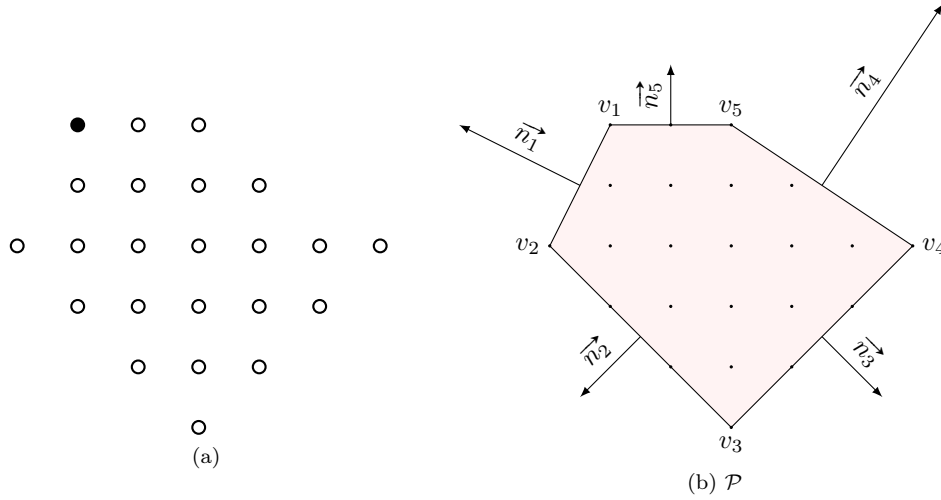


FIGURE 3.34

indices de construction  $S_{\vec{v}_k}(r)$ . Pour que la transformation en distance puisse être calculée en un balayage, il est nécessaire que chaque vecteur de déplacement respecte la condition de causalité ce qui implique que l'origine du plan soit le premier point de  $B$  dans l'ordre de balayage.

$\mathcal{P}$  occupe une place centrale dans l'algorithme, qui utilise simultanément ses  $\mathcal{V}$ - et  $\mathcal{H}$ -représentations :

$$\mathcal{P} = \text{conv}\{v_l, 1 \leq l \leq L\},$$

$$\mathcal{P} = \{x : \vec{n}_l \cdot x \leq y_l, 1 \leq l \leq L\}.$$

Les sommets de  $P$ , notés  $v_l$ , sont tels que  $v_0 = O$ ,  $v_l = v_{l+L}$ ,  $v_{l-1}$  est adjacent à  $v_l$  et  $\det(\overrightarrow{v_{l-1}v_l}, \overrightarrow{v_lv_{l+1}}) > 0$ . Le vecteur normal extérieur de l'arête  $\overrightarrow{v_lv_{l+1}}$  est noté  $\vec{n}_l$ , il est normalisé de façon à ce que ses composantes soient premières entre elles. Chaque composante de  $y$  est la valeur de la fonction d'appui de  $\mathcal{P}$  pour le vecteur normal correspondant (voir définition 3.13) :

$$\vec{n}_l \cdot \vec{v}_l = \vec{n}_l \cdot \vec{v}_{l+1} = y_l.$$

Le polygone convexe  $\mathcal{P}$  de la figure 3.34b peut être décrit par les deux représentations équivalentes :

$$\mathcal{P} = \text{conv}\{(0 \ 0), (-1 \ -2), (2 \ -5), (5 \ -2), (2 \ 0)\},$$

$$x \in \mathcal{P} \Leftrightarrow \begin{cases} -2x_1 + x_2 \leq 0 \\ -x_1 - x_2 \leq 3 \\ x_1 - x_2 \leq 7 \Leftrightarrow A \cdot x \leq y \\ 2x_1 + 3x_2 \leq 4 \\ x_2 \leq 0 \end{cases}$$

où  $A = \begin{pmatrix} -2 & 1 \\ -1 & -1 \\ 2 & 3 \\ 0 & 1 \end{pmatrix}$  et  $y = \begin{pmatrix} 0 \\ 3 \\ 7 \\ 4 \\ 0 \end{pmatrix}$  forment la  $\mathcal{H}$ -représentation minimale de  $\mathcal{P}$ .  $A$  est constituée des vecteurs normaux aux arêtes de  $\mathcal{P}$ .

### Séquence de polytopes rationnels

La forme des disques construits est contrôlée par des contraintes externes dont le but est de maintenir constantes les orientations de ses arêtes et de fixer leurs longueurs. Ces contraintes sont exprimées sous la forme d'une séquence de  $\mathcal{H}$ -polytopes rationnels  $\mathcal{P}(r)$  qui décrivent toutes les phases de la croissance. Les  $\mathcal{H}$ -description des  $\mathcal{P}(r)$  partagent la même matrice et donc les mêmes directions de facettes :

$$\mathcal{P}(r) = \{A \cdot x \leq y(r)\}.$$

Les sommets de  $\mathcal{P}$  sont ses faces de dimension nulle, c'est-à-dire les points  $v_l$  tels que, pour un vecteur  $\vec{v}$  donné,  $F(\mathcal{P}, \vec{v}) = \{v_l\}$ . Gardons ce vecteur  $\vec{v}$ , on note  $\{v_l(r)\} = F(\mathcal{P}(r), \vec{v})$  le sommet correspondant du polygone intermédiaire  $\mathcal{P}(r)$  de rayon  $r$ .

L'objectif est que chaque disque discret  $D_{<}(O, r)$  de la pseudo-distance construite soit la discrétisation de GAUSS du polygone rationnel de même paramètre  $r$  (rayon) :

$$D_{<}(O, r) = \mathcal{P}(r) \cap \mathbb{Z}^2. \quad (3.32)$$

L'algorithme consiste à déplacer les demi-plans depuis leur position initiale tangente à l'origine du plan ( $h(\mathcal{P}(0), \vec{v}) = 0$ ) jusqu'à leur position finale en s'assurant que tous les points discrets sont atteints par un chemin depuis l'origine. Une série de positions est calculée pour chacun des demi-plans comme illustré dans le tableau 3.38 pour les polygones de la figure 3.39.

### Choix des vecteurs de voisinage

Chaque paire de voisins successifs définit un cône d'influence similaire à ceux des distances de chanfrein. Dans un cône d'influence, chaque point peut être atteint par un chemin minimal qui n'utilise que les deux voisins. Cependant, contrairement aux distances de chanfrein, les sommets de l'élément structurant ne sont pas nécessairement à l'intersection de deux cônes. Le nombre  $K$  de voisins n'est donc pas nécessairement égal au nombre  $L$  des sommets de  $\mathcal{P}$ .

Deux contraintes s'appliquent sur les paires de voisins consécutifs  $(\vec{v}_k, \vec{v}_{k+1})$  :

- (i) tous les points du cône d'influence doivent être atteints par les voisins.  
Une condition suffisante est que  $\vec{v}_k$  et  $\vec{v}_{k+1}$  forment un cône unimodulaire ( $\det[\vec{v}_k, \vec{v}_{k+1}] = 1$ ) [174] ;
- (ii) tous les chemins vers un point du cône d'influence doivent être inclus dans l'élément structurant (conséquence du corollaire 2.34). Autrement dit, si  $p \in D_{<}(O, r)$  est dans le cône  $(\vec{v}_k, \vec{v}_{k+1})$  et  $p = a\vec{v}_k + b\vec{v}_{k+1}$  alors tous les points du parallélogramme  $(O, O + a\vec{v}_k, p, O + b\vec{v}_{k+1})$  doivent appartenir à  $D_{<}(O, r)$ .

L'algorithme 3.35 procède en deux étapes. Dans un premier temps, le cône formé par la première et la dernière arête est décomposé en une série de cônes unimodulaires. Dans un second temps, les sommets de  $\mathcal{P}$  sont testés un à un. Lorsqu'un sommet non atteignable selon la condition (ii) est détecté dans le cône  $(\vec{v}_k, \vec{v}_{k+1})$ , le vecteur  $\vec{v}_k + \vec{v}_{k+1}$  est inséré, divisant le cône en deux. La procédure, identique à l'insertion d'une fraction médiane dans l'arbre de STERN-BROCOT, préserve l'unimodularité.

---

**Algorithme 3.35:** Choix du voisinage

---

**Données :**  $\{v_l, 1 \leq l \leq L\}$  :  $\mathcal{V}$ -représentation du polygone à décomposer  
**Résultat :** Voisinage  $\mathcal{N} = \{\vec{v}_k, 1 \leq k \leq K\}$   
// Sélection du cône initial  
 $K \leftarrow 2$   
 $\vec{v}_1 \leftarrow \vec{v}_1 \vec{v}_2 / \gcd(\vec{v}_1 \vec{v}_2)$   
 $\vec{v}_2 \leftarrow \vec{v}_1 \vec{v}_L / \gcd(\vec{v}_1 \vec{v}_L)$   
// Étape 1 : obtention de cônes unimodulaires, condition (i)  
**tant que**  $\det(\vec{v}_{K-1}, \vec{v}_K) \neq 1$  **faire**  
    //  $(\vec{v}_{K-1}, \vec{v}_K)$  n'est pas unimodulaire  
    trouver un  $p$  tel que  $\det(\vec{v}_{K-1}, p) = 1$  (algorithme d'Euclide étendu)  
    //  $(\vec{v}_{K-1}, p + n\vec{v}_{K-1})$  est unimodulaire pour tout  $n$ , on choisit la valeur de  
     $n$  qui forme l'angle  $(\vec{v}_{K-1}, p + n\vec{v}_{K-1})$  le plus ouvert  
     $n = \left\lfloor \frac{-\det(p, \vec{v}_K)}{\det(\vec{v}_{K-1}, \vec{v}_K)} \right\rfloor$   
    // Insertion de  $p + n\vec{v}_{K-1}$  avant  $K$   
     $\mathcal{N} \leftarrow \{\vec{v}_1, \dots, \vec{v}_{K-1}, p + n\vec{v}_{K-1}, \vec{v}_K\}$   
     $K \leftarrow K + 1$   
**fin**  
// Étape 2 : insertion de voisins pour la condition (ii)  
 $k \leftarrow 1$   
**pour**  $l = 3$  à  $L - 1$  **faire**  
    // Sélection du cône  $(\vec{v}_k, \vec{v}_{k+1})$  contenant  $v_l$   
    **tant que**  $\det(\vec{v}_{k+1}, \vec{O}v_l) > 0$  **faire**  $k \leftarrow k + 1$   
    **tant que**  $v_l$  n'est pas atteignable **faire**  
        // Division du cône (insertion de  $\vec{v}_k + \vec{v}_{k+1}$ )  
         $\mathcal{N} \leftarrow \{\vec{v}_1, \dots, \vec{v}_k, \vec{v}_k + \vec{v}_{k+1}, \vec{v}_{k+1}, \dots, \vec{v}_K\}$   
         $K \leftarrow K + 1$   
    **fin**  
**fin**

---

Sur le polygone  $\mathcal{P}$  de la figure 3.34b, algorithme 3.35 donne comme vecteurs du cône initial,  $\vec{v}_1 \vec{v}_2 / \gcd(\vec{v}_1 \vec{v}_2) = (-1, -2)$ ,  $\vec{v}_1 \vec{v}_L / \gcd(\vec{v}_1 \vec{v}_L) = (1, 0)$ . Or ce cône n'est pas unimodulaire car  $\det \begin{pmatrix} -1 & -2 \\ 1 & 0 \end{pmatrix} = 2$ . Le vecteur  $(0, -1)$  est inséré et l'étape 1 se termine avec les deux cônes unimodulaires  $((-1, -2), (0, -1))$  et  $((0, -1), (1, 0))$ . Au cours de la seconde phase, le vecteur  $(1, -1)$  est inséré dans le cône  $((0, -1), (1, 0))$  car le sommet  $v_3$  n'est pas atteignable dans ce cône. Il l'est ensuite dans le cône  $((0, -1), (1, -1))$ . Finalement, le vecteur  $(2, -1)$  est inséré dans le cône  $((1, -1), (0, 1))$  pour rendre  $v_4$  atteignable. Les vecteurs de voisinage tels qu'ils sont numérotés sur la figure 3.39 sont ajoutés dans l'ordre :  $\vec{v}_1, \vec{v}_5, \vec{v}_3, \vec{v}_2$  et enfin  $\vec{v}_4$ .

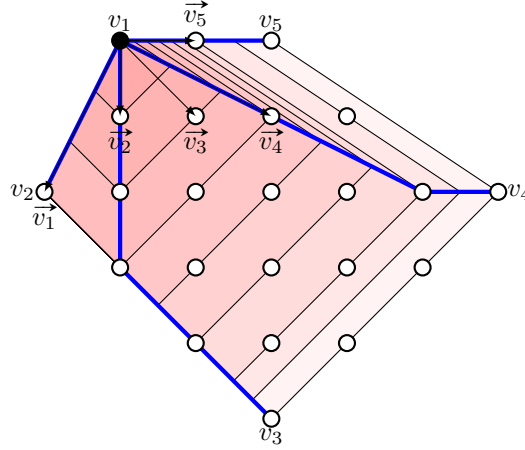


FIGURE 3.39

### Construction des polygones rationnels intermédiaires

Les vecteurs de déplacement étant connus, un chemin  $P_l$  est construit depuis l'origine vers chaque sommet  $v_l$  de  $\mathcal{P}$ . Si  $v_l$  est dans le cône  $(\vec{v}_k, \vec{v}_{k+1})$ , alors il existe un couple d'entiers naturels  $(\alpha, \beta)$  pour lequel  $\vec{Ov}_l$  s'écrit  $\alpha\vec{v}_k + \beta\vec{v}_{k+1}$ . Le chemin vers  $v_l$  est construit avec une séquence de  $\alpha$  déplacements  $\vec{v}_k$  puis  $\beta$   $\vec{v}_{k+1}$ . En pratique, ces chemins ne sont pas calculés explicitement, une fois pour toutes, mais au fur et à mesure de la construction des polygones rationnels intermédiaires (repère 1, algorithme 3.36).

Ces chemins jouent le rôle de guides pour la construction des polygones rationnels intermédiaires, les sommets de la séquence  $(v_l(r), 1 \leq r \leq r_{\max})$  étant placés soit sur les points discrets de  $P_l$  soit sur un segment qui joint deux points consécutifs de ce chemin. Deux arêtes consécutives sont liées par la position de leur sommet commun et, par conséquent, toutes les arêtes du polygone  $\mathcal{P}(r)$  sont liées. Déplacer la première arête entraîne, de proche en proche, les autres arêtes. L'algorithme 3.36 construit la séquence des polygones rationnels  $\mathcal{P}(r)$  en s'assurant que chaque point discret appartient à une face d'un polygone :

$$\forall p \in B, \exists r, \exists \vec{v}, p \in H(\mathcal{P}(r), \vec{v}) .$$

Sur la figure 3.39,  $v_3 = (2, -5)$  est dans le cône  $(\vec{v}_2, \vec{v}_3) = ((0, -1), (1, 1))$  et  $(2, -5) = 3 \times (0, -1) + 2 \times (1, 1)$ . Le chemin vers  $v_3$  est constitué de trois déplacements selon  $\vec{v}_2$  suivis de deux déplacements selon  $\vec{v}_3$ . Les coordonnées des sommets rationnels  $v_2(r)$  à  $v_4(r)$  sont présentées dans le tableau 3.37, les polygones intermédiaires sur la figure 3.39 et les coefficients de leurs  $\mathcal{H}$ -représentations dans le tableau 3.38.

### Construction des disques

Les indices de construction sont choisis pour avoir les plus grandes valeurs permises, le disque  $S_{\vec{v}_k}(r)$ , utilisé dans le voisinage  $\vec{v}_k$  pour construire le disque de rayon  $r$ , est le plus grand disque qui respecte les contraintes du polygone rationnel  $\mathcal{P}(r)$  :

$$\begin{aligned} S_{\vec{v}_k}(r) &= \max \{ s : (\mathcal{P}(s))_{\vec{v}_k} \subseteq \mathcal{P}(r) \} , \\ &= \max \{ s : \forall l, h(\mathcal{P}(s), \vec{n}_l) + h(\vec{v}_k, \vec{n}_l) \leq h(\mathcal{P}(r), \vec{n}_l) \} . \end{aligned} \quad (3.33)$$

---

**Algorithme 3.36:** Calcul des polygones rationnels intermédiaires

---

**Données :**  $\{v_l, 1 \leq l \leq L\}$ ,  $\mathcal{V}$ -représentation du polygone à décomposer  
 $\mathcal{P}(0) \leftarrow \emptyset$   
 $\mathcal{P}(1) \leftarrow \{O\}$   
 $r \leftarrow 1$   
**tant que**  $\mathcal{P}(r) \neq \mathcal{P}$  **faire**

$r \leftarrow r + 1$   
 $y_1(r) \leftarrow 0$   
 $y_L(r) \leftarrow 0$   
 $l_0 \leftarrow 1$   
**pour**  $l = 2$  à  $L - 1$  **faire**

// Voisinage utilisé dans le chemin vers le sommet  $v_l$   
**1** **si**  $\det(v_l(r), \overrightarrow{v_{k+1}}) < \det(v_l, \overrightarrow{v_{k+1}})$  **alors**  
    |  $\vec{v} \leftarrow \overrightarrow{v_k}$   
**sinon**  
    |  $\vec{v} \leftarrow \overrightarrow{v_{k+1}}$   
**fin**

// Calcul de  $y_l(r)$  indépendamment de  $y_{l-1}(r)$   
**2**  $y_l(r) \leftarrow \min\{\lfloor y_l(r-1) + 1 \rfloor; h(v_l, \vec{n}_l)\}$   
**si**  $(y_{l-1}(r) - y_{l-1}(r-1)) \times \vec{v} \cdot \vec{n}_l < (y_l(r) - y_l(r-1)) \times \vec{v} \cdot \vec{n}_{l-1}$  **alors**  
    | // Réajuster  $y_l(r)$  si sa valeur dépasse la contrainte imposée par  
         $y_l(r-1)$   
    | 
$$y_l(r) = y_l(r-1) + \frac{\vec{v} \cdot \vec{n}_l}{\vec{v} \cdot \vec{n}_{l-1}}(y_{l-1}(r) - y_{l-1}(r-1))$$
  
**fin**

**si**  $(y_{l-1}(r) - y_{l-1}(r-1)) \times \vec{v} \cdot \vec{n}_l > (y_l(r) - y_l(r-1)) \times \vec{v} \cdot \vec{n}_{l-1}$  **alors**  
    | //  $y_2(r)$  à  $y_{l-1}(r)$  sont à recalculer  
    |  $l_0 \leftarrow l$   
**fin**

**fin**  
**pour**  $l = l_0$  à  $3$  **faire** (pas -1)  
    | **si**  $(y_{l-1}(r) - y_{l-1}(r-1)) \times \vec{v} \cdot \vec{n}_l > (y_l(r) - y_l(r-1)) \times \vec{v} \cdot \vec{n}_{l-1}$  **alors**  
        | 
$$y_{l-1}(r) = y_{l-1}(r-1) + \frac{\vec{v} \cdot \vec{n}_{l-1}}{\vec{v} \cdot \vec{n}_l}(y_l(r) - y_l(r-1))$$
  
    | **fin**  
**fin**

**fin**

---



$r$	2	3	4	5	6	7	8	9
$v_2(r)$	$\left(\frac{-1}{3}, \frac{-2}{3}\right)$	$\left(\frac{-2}{3}, \frac{-4}{3}\right)$	$(-1, -2)$	$(-1, -2)$	$(-1, -2)$	$(-1, -2)$	$(-1, -2)$	$(-1, -2)$
$v_3(r)$	$(0, -1)$	$(0, -2)$	$(0, -3)$	$\left(\frac{1}{2}, \frac{-7}{2}\right)$	$(1, -4)$	$\left(\frac{3}{2}, \frac{-9}{2}\right)$	$\left(\frac{7}{4}, \frac{-19}{4}\right)$	$(2, -5)$
$v_4(r)$	$\left(\frac{2}{3}, \frac{-1}{3}\right)$	$\left(\frac{4}{3}, \frac{-2}{3}\right)$	$(2, -1)$	$\left(\frac{8}{3}, \frac{-4}{3}\right)$	$\left(\frac{10}{3}, \frac{-5}{3}\right)$	$(4, -2)$	$\left(\frac{9}{2}, -2\right)$	$(5, -2)$
$v_5(r)$	$\left(\frac{1}{6}, 0\right)$	$\left(\frac{1}{3}, 0\right)$	$\left(\frac{1}{2}, 0\right)$	$\left(\frac{2}{3}, 0\right)$	$\left(\frac{5}{6}, 0\right)$	$(1, 0)$	$\left(\frac{3}{2}, 0\right)$	$(2, 0)$

Tableau 3.37

$r$	0	1	2	3	4	5	6	7	8	9
$y_1(r)$	$-\infty$	0	0	0	0	0	0	0	0	0
$y_2(r)$	$-\infty$	0	1	2	3	3	3	3	3	3
$y_3(r)$	$-\infty$	0	1	2	3	4	5	6	$\frac{13}{2}$	7
$y_4(r)$	$-\infty$	0	$\frac{1}{3}$	$\frac{2}{3}$	1	$\frac{4}{3}$	$\frac{5}{3}$	2	3	4
$y_5(r)$	$-\infty$	0	0	0	0	0	0	0	0	0

Tableau 3.38 – Coefficients des  $\mathcal{H}$ -représentations des polygones rationnels intermédiaires.

Cette expression de  $S_{\vec{v}_k}(r)$  garantit que chaque élément structurant  $D_<(O, r)$  est un sous-ensemble de  $\mathcal{P}(r) : \forall r, D_<(O, r) \subseteq \mathcal{P}(r)$ .

Finalement, les coûts de déplacements  $\hat{C}$  sont déduits des indices de constructions selon (2.40).

### 3.4.2 Validité de l'algorithme

Par construction, les arêtes des polygones rationnels intermédiaires ont des directions identiques orthogonales aux normales des demi-plans :

$$\forall l, \forall s, \forall r, \det(\overrightarrow{v_l(s)v_{l+1}(s)}, \overrightarrow{v_l(r)v_{l+1}(r)}) = 0. \quad (3.34)$$

Les demi-plans sont positionnés de telle sorte que les arêtes des polygones rationnels intermédiaires  $\mathcal{P}(r)$  ont des longueurs croissantes.

$$\forall s > 0, \forall r \geq s, \left\| \overrightarrow{v_l(r)v_{l+1}(r)} \right\| \geq \left\| \overrightarrow{v_l(s)v_{l+1}(s)} \right\|. \quad (3.35)$$

$r$	1	2	3	4	5	6	7	8	9
$S_1(r)$	0	0	0	1	1	1	1	1	1
$S_2(r)$	0	1	2	3	3	3	3	3	3
$S_3(r)$	0	0	1	2	3	4	5	5	6
$S_4(r)$	0	0	0	1	2	3	4	4	5
$S_5(r)$	0	0	0	0	0	0	1	4	7

Tableau 3.40 – Indices de construction des disques.

$r$	0	1	2	3	4	5	6	7	8	9
$\hat{C}_{\vec{v}_1}(r)$	3	9	9	9	9	9	9	9	9	9
$\hat{C}_{\vec{v}_2}(r)$	1	2	3	9	9	9	9	9	9	9
$\hat{C}_{\vec{v}_3}(r)$	2	3	4	5	6	8	9	9	9	9
$\hat{C}_{\vec{v}_4}(r)$	3	4	5	6	8	9	9	9	9	9
$\hat{C}_{\vec{v}_5}(r)$	6	7	7	7	8	8	8	9	9	9

Tableau 3.41 – Coûts minimaux de déplacements  $\hat{C}_{\vec{v}_k}$ .

Une conséquence directe des propriétés des polygones rationnels, orientation contante et longueurs d'arêtes croissantes (3.34) et (3.35), est que ceux-ci peuvent être construits itérativement par somme de MINKOWSKI, (3.12) s'applique :

$$\mathcal{P}(r) = \mathcal{P}(s) \oplus \mathcal{Q}(s, r) ,$$

$$\mathcal{Q}(s, r) = \{x : A \cdot x \leq y(r) - y(s)\} .$$

### Inclusion des disques dans les polygones rationnels

#### Proposition 3.42.

$$\forall r \geq 0, D_{<}(O, r) \subseteq \mathcal{P}(r) . \quad (3.36)$$

*Démonstration.* La proposition est vraie pour  $r = 0$  et  $r = 1$  car, par construction,  $D_{<}(O, 0) = \mathcal{P}(0) = \emptyset$  et  $D_{<}(O, 0) = \mathcal{P}(0) = \{O\}$ . Supposons-la vraie jusqu'à  $r - 1$ . On rappelle la méthode de construction des disques (2.34) :

$$D_{<}(O, r) = \bigcup_{\vec{v}_k \in \mathcal{N}} D_{<}(O + \vec{v}_k, S_{\vec{v}_k}(r)) .$$

Puisque  $S(r) < r$ , (3.36) s'applique à  $D_{<}(O + \vec{v}_k, S_{\vec{v}_k}(r))$ . De plus, d'après (3.33),  $(\mathcal{P}(S(r)))_{\vec{v}_k} \subseteq \mathcal{P}(r)$ . Autrement dit,  $D_{<}(O + \vec{v}_k, S_{\vec{v}_k}(r)) \subseteq (\mathcal{P}(S(r)))_{\vec{v}_k} \subseteq \mathcal{P}(r)$ . Toutes les parties qui constituent  $D_{<}(O, r)$  sont incluses dans  $\mathcal{P}(r)$  donc  $D_{<}(O, r)$  l'est également. La proposition est vraie aussi au rang  $r$  et, par récurrence, à toute valeur de  $r$ . □

### Inclusion des points discrets des polygones rationnels dans les disques

**Proposition 3.43.** Si  $q \in \mathbb{Z}^n$  est un sommet de  $\mathcal{P}(s)$  et  $\vec{q}\vec{p}$  un vecteur discret tel que  $\vec{n} \cdot \vec{q}\vec{p} \geq 0$  pour tous les vecteurs normaux aux facettes qui contiennent  $p$  alors  $S_{\vec{q}\vec{p}}(r) \geq s$ .

*Démonstration.* Supposons que  $p$  soit un sommet de  $\mathcal{P}(r)$ , alors il existe  $\vec{v}$  tel que  $F(\mathcal{P}(r), \vec{v}) = \{p\}$  où  $\vec{v}$  est n'importe quelle combinaison à coefficients positifs des vecteurs normaux extérieurs des facettes contenant  $p$ . Un choix possible pour  $\vec{v}$  consiste à prendre leur somme. Selon (3.18) et par construction de  $\mathcal{P}(r)$ , si  $s \leq r$  alors  $F(\mathcal{P}(s), \vec{v}) = \{q\}$  est un sommet de  $\mathcal{P}(s)$  et  $F(\mathcal{Q}(s, r), \vec{v}) = \{p - q\}$  est un sommet de  $\mathcal{Q}(s, r)$ . On a donc,  $\mathcal{P}(s) \oplus \{p - q\} \subseteq \mathcal{P}(r)$  et  $\forall \vec{u}, h(\mathcal{P}(s), \vec{u}) + h(p - q, \vec{u}) \leq h(\mathcal{P}(r), \vec{u})$ . □

**Corollaire 3.44.** Si, en plus des conditions de la proposition 3.43,  $q \in D_{<}(O, s)$  et  $\vec{q}\vec{p}$  est un vecteur de déplacement alors  $p \in D_{<}(O, r)$ .

Pour assurer les conditions du corollaire 3.44, les polytopes rationnels sont construits de telle façon qu'il existe un chemin depuis l'origine vers chaque sommet  $v_l$  de  $\mathcal{P}$ . La séquence des positions intermédiaires du sommet  $v_l$  de  $\mathcal{P}$ ,  $\{v_{l,i}\}_{i \in [1..N]}$  contient un chemin depuis l'origine vers  $v_l$  qui emprunte les vecteurs de déplacement du cône d'influence de  $v_l$ .

Le choix des positions des demi-plans, en particulier le fait qu'elles prennent toutes les valeurs entières de la fonction d'appui (car  $y_l(r) \leq \lfloor y_l(r-1) + 1 \rfloor$ , repère 2, algorithme 3.36) garantit que chaque point discret sur les chemins  $P_l$  est un sommet d'un polygone intermédiaire. Chaque disque intermédiaire est donc assuré de contenir tous les sommets discrets du polytope rationnel de même rayon.

### 3.5 Conclusion

Dans ce chapitre, nous avons présenté un algorithme de transformation en distance généralisée pour les distances à poids variables introduites précédemment. Il procède par propagation des longueurs de chemins minimaux obtenus dans un voisinage local. En cela, son principe ne diffère pas de celui des algorithmes classiques adaptées aux distances simples, à séquence de voisinages ou de chanfrein. Son originalité réside dans la détermination des coûts de déplacements minimaux, intrinsèquement liés aux propriétés des séquences non décroissantes d'entiers naturels et à l'inverse de LAMBEK-MOSER de ses séquences.

En reprenant des idées proposées par des auteurs tels que S. CHEN et R. M. HARALICK, X. WANG et G. BERTRAND, nous calculons, en un seul balayage, une transformation en distance mais avec une variété de distances plus grande.

Le calcul des distances à séquence de voisinages, par exemple, est sensiblement accéléré, par l'application simultanée d'une translation aux vecteurs de déplacement qui rend possible ce calcul en un balayage et par la décomposition des voisinages qui permet de réduire notablement le nombre de ces vecteurs. La transformation en distance obtenue est décentrée, et il est possible de la recentrer dans un deuxième temps, lorsque c'est nécessaire. Le nombre de passes dans l'image ou le volume est réduit de trois au minimum pour la méthode de R. STRAND, B. NAGY, C. FOUARD et G. BORGEFORS [166] à deux au maximum, la dernière ne nécessitant pas de propagation peut être combinée avec la première. Le résultat centré est fourni avec une latence minimale, au fur et à mesure de l'entrée des données d'image.

Nous montrons également comment obtenir une pseudo-distance dont un des disques est une forme arbitraire convexe. Le schéma de construction des disques permet de décomposer des formes au-delà de ce qui est possible avec la somme de MINKOWSKI. Avec notre méthode, toute forme qui contient des points non visibles du plan peut être décomposée. L'application immédiate est l'érosion morphologique en un balayage par seuillage de la transformation en distance.

Dans les deux cas, les algorithmes ont une complexité temporelle linéaire avec le nombre de pixels. Le coût par pixel est faible grâce au petit nombre de vecteurs de déplacement nécessaires. Surtout, leur complexité spatiale est très limitée puisqu'ils ne requièrent que quelques lignes pour une image ou quelques plans pour un volume. Leur structure les rend utilisables dans un enchaînement d'opérations (*pipeline*) avec de faibles besoins en mémoire et une latence considérablement réduite.



# Chapitre 4

## Axe médian

### 4.1 Introduction

Nous avons vu dans le chapitre 3, la transformation en distance d'un ensemble de points  $X$ , qui associe à tout point  $p$  sa distance depuis le complément de l'ensemble, c'est-à-dire le rayon du plus grand disque ouvert centré  $p$  et contenu dans  $X$ . Un tel disque est dit maximal s'il n'est pas contenu dans un autre disque lui-même inclus dans  $X$ . L'ensemble des centres des disques maximaux, appelé *axe médian* est un outil de représentation des images binaires dont les applications vont du codage d'image à la reconnaissance de formes. Sa forme pondérée, qui associe le rayon du disque maximal à son centre, est réversible.

La détection de l'axe médian à partir de la transformation en distance nécessite d'explicitier les relations d'inclusions de disques. Pour les distances dont les disques sont construits par somme de MINKOWSKI, les distances simples  $d_4$  et  $d_8$  et les distances à séquence de voisinages, les relations d'inclusion s'expriment aisément. Le cas des distances de chanfrein est plus complexe dès lors que l'on augmente la taille des masques. En particulier, É. RÉMY, É. THIEL ont montré que le masque de chanfrein doit être distingué du voisinage de test, ensemble minimal de vecteurs qui permet la détection de toutes les inclusions.

Nous proposons une méthode en deux phases qui détermine si un masque induit une norme et qui dans ce cas, calcule les tables d'inclusion et le voisinage de test en exploitant les propriétés géométriques des boules des normes de chanfrein. La méthode s'appuie sur la représentation des disques sous forme de  $\mathcal{H}$ -polytopes, adaptée aux tests d'inclusion, et sur une décomposition de la boule rationnelle en cônes unimodulaires simpliciaux, qui permet de rechercher plus efficacement les vecteurs du voisinage de test. La stratégie d'exploration de l'espace de recherche permet de réduire le temps de calcul de plusieurs ordres de grandeur. Incidemment, nous donnons une formule close, nouvelle, qui exprime la norme de chanfrein d'un vecteur quelconque.

La première phase construit une boule rationnelle en pondérant les vecteurs de déplacement par l'inverse du poids qui leur est associé puis réalise une triangulation unimodulaire de cette boule. Conceptuellement, l'algorithme opère sur l'enveloppe convexe des vecteurs pondérés et triangule cette enveloppe. Tout en vérifiant les conditions de norme, l'algorithme produit une description de la géométrie de la boule pour le masque considéré. La seconde phase est lancée pour les normes ; à partir de la description géométrique obtenue précédemment, chaque boule (pour chaque valeur de rayon) est décrite comme l'intersection d'un ensemble de demi-espace ( $\mathcal{H}$ -représentation), ce qui facilite les tests d'inclusion et permet de calculer le voisinage de test  $\mathcal{M}_{\text{Lit}}$  et les valeurs des tables d'inclusion.

Les notions, définitions et résultats classiques sur l'axe médian, en particulier pour les distances

de chanfrein, seront rappelés dans la section 4.2.1. La section 4.3 justifie l'application du formalisme de  $\mathcal{H}$ -polytope dans notre contexte et présente les principes de la méthode proposée. Les détails de la méthode sont donnés dans les sections 4.4 et 4.5, qui présentent respectivement un algorithme de triangulation de la boule rationnelle en 3D et le calcul du voisinage de test et des tables d'inclusion. Les résultats discutés en section 4.6 montrent des gains significatifs en temps de calcul.

La méthode ainsi que ses résultats ont été présentés, pour le cas 2D, à l'occasion de la conférence *Discrete Geometry for Computer Imagery 2008* et dans une version plus détaillée et étendue au cas 3D dans *Pattern Recognition* [105, 106].

## 4.2 Préliminaires

### 4.2.1 Axe Médian

**Définition 4.1** (disque maximal). On appelle *disque maximal* dans un ensemble  $X$ , un disque  $D$  inclus dans  $X$  lorsqu'il n'existe aucun autre disque dans  $X$  qui inclut  $D$ .

Si  $D_{<}(p, r)$  est maximal alors nécessairement  $DT_X(p) = r$  car  $D_{<}(p, DT_X(p))$  est le disque de plus grand rayon centré sur  $p$  qui inclut, par définition des disques, les disques centrés sur  $p$  de rayons inférieurs.

**Définition 4.2** (axe médian). L'*axe médian* de l'ensemble  $X$ , noté  $MAT_X$ , est la fonction qui associe à un centre de disque maximal, son rayon, et 0 à tout autre point :

$$MAT_X(p) = \begin{cases} 0 & \text{si } \exists q, r : D_{<}(p, DT_X(p)) \subsetneq D_{<}(q, r) \subset X, \\ DT_X(p) & \text{sinon.} \end{cases} \quad (4.1)$$

Autrement dit,  $D_{<}(p, r)$  est maximal si et seulement si  $MAT_X(p) = r$ .

### 4.2.2 Axe Médian des Distances de Chanfrein

Pour les distances simples  $d_4$  et  $d_8$ , l'extraction de l'axe médian s'effectue par une détection des maxima locaux dans la carte de distance [148, section 4.3]. La situation est plus complexe pour les distances de chanfrein, même pour de petits masques, dès lors que les poids ne sont plus unitaires. Dans ce cas, certaines valeurs entières de la distance ne sont pas atteignables, et deux rayons différents  $r$  et  $r'$  peuvent alors correspondre au même ensemble de points. Les rayons  $r$  et  $r'$  sont dits *équivalents*. Puisque la transformation en distance donne toujours le plus grand rayon équivalent, les critères basés sur les différences de rayons échouent à détecter les inclusions de disques. Dans le cas des masques  $3 \times 3$  en 2D ou  $3 \times 3 \times 3$  en 3D, un simple remplacement des valeurs de la carte de distance par le plus petit rayon équivalent est suffisant [4, 171]. Cependant, cette méthode échoue pour des masques plus grands et la méthode la plus générale d'extraction de l'axe médian met en œuvre des tables de correspondance qui représentent les relations d'inclusion de disques : pour un vecteur  $\vec{v}$  donné du voisinage de test, la table fait correspondre à chaque valeur de rayon  $r_1$ , le rayon  $r_2$  du plus petit disque centré sur  $O + \vec{v}$  qui couvre  $D_{<}(O, r_1)$  [20] :

$$Lut_{\vec{v}}(r_1) = \min \{ r_2 : D_{<}(O, r_1) \subseteq D_{<}(O + \vec{v}, r_2) \} . \quad (4.2)$$

De manière équivalente, en utilisant des boules fermées :

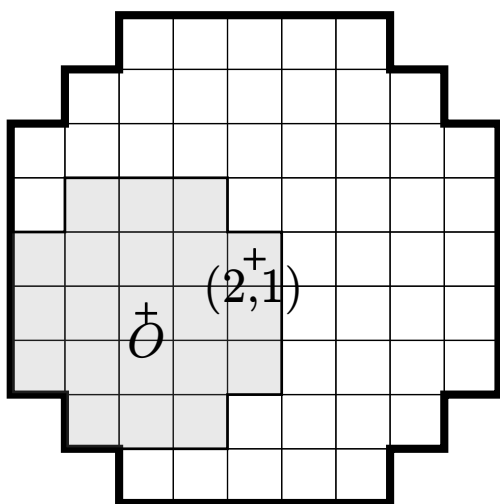
$$Lut_{\vec{v}}(r_1) = 1 + \min \{ r_2 : D_{\leq}(O, r_1 - 1) \subseteq D_{\leq}(O + \vec{v}, r_2) \} . \quad (4.3)$$

$$\mathcal{M}_{\text{Lut}} = \{\vec{a} = (1\ 0); \vec{b} = (1\ 1); \vec{c} = (2\ 1)\}$$

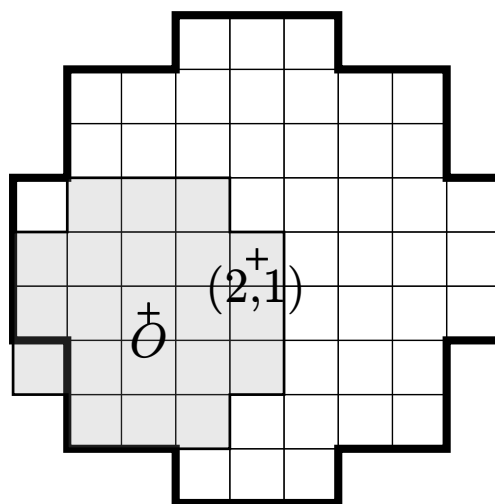
(a) Voisinage de test pour la distance  $d_{5,7,11}$

R	$\vec{a}$	$\vec{b}$	$\vec{c}$	R	$\vec{a}$	$\vec{b}$	$\vec{c}$	R	$\vec{a}$	$\vec{b}$	$\vec{c}$
5	6	8	12	21		27		38		44	
7	11	12	17	25	28	30	34	39		45	
10	12	15	19	27		33		40	44		
11		17		28		34		42		48	
<b>14</b>	17	19	<b>23</b>	29	33			46		52	
15	19			30	34			49		55	
16		22		31		37		53		59	
18	22	23	28	32		38		60		66	
20	23	26	30	35	39	41	45				

(b) Lut pour la distance  $d_{5,7,11}$



(c) Disques  $D_{<}(O, 14)$  et  $D_{<}(O + \vec{c}, 23)$



(d) Disques  $D_{<}(O, 14)$  et  $D_{<}(O + \vec{c}, 22)$

Comme É. THIEL l'a remarqué, le voisinage de test  $\mathcal{M}_{\text{Lut}}$  n'est pas nécessairement égal au masque de chanfrein  $\mathcal{M}$  [172].

Considérons par exemple la distance  $d_{5,7,11}$  [146, Fig. 14].  $\text{Lut}_{(1,0)}(10) = 12$  signifie que  $D_{<}(O, 10) \subseteq D_{<}((2, 1), 12)$  mais  $D_{<}(O, 10) \not\subseteq D_{<}((2, 1), 11)$ . Ou, en termes de boules fermées,  $D_{\leq}(O, 9) \subseteq D_{\leq}((2, 1), 11)$  mais  $D_{\leq}(O, 9) \not\subseteq D_{\leq}((2, 1), 10)$ .

### Coefficients des Tables d'Inclusion

Une méthode générale pour le calcul des coefficients des tables d'inclusion a été donnée par É. RÉMY et É. THIEL [145, 172, 173]. Le principe est que la relation d'inclusion des boules peut être extraite directement depuis les valeurs de la fonction distance. Si  $d(O, p) = r_1$  et  $d(O, p + \vec{v}_i) = r_2$ , on déduit :

$$\begin{aligned} p \in D_{\leq}(O, r_1) &= D_{<}(O, r_1 + 1) , \\ p + \vec{v}_i \notin D_{\leq}(O, r_2 - 1) &= D_{<}(O, r_2) , \end{aligned}$$

donc  $D_{<}(O + \vec{v}_i, r_1 + 1) \not\subset D_{<}(O, r_2)$  et  $\text{Lut}_{\vec{v}_i}(r_1 + 1) > r_2$ . Si  $\forall p, d(O, p) \leq r_1 \Rightarrow d(O, p + \vec{v}_i) \leq r_2$  alors  $\text{Lut}_{\vec{v}_i}(r_1 + 1) = r_2 + 1$ .

Finalement,  $\text{Lut}_{\vec{v}_i}(r) = 1 + \max \{d(O, p + \vec{v}_i) : d(O, p) < r\}$ .

Cette méthode nécessite uniquement un balayage des valeurs de la fonction distance pour chaque vecteur de déplacement  $\vec{v}_i$ , et ce, quel que soit le rayon considéré (toutefois la taille de la zone visitée devra prendre en compte une valeur maximale de distance). De plus, la zone visitée peut être restreinte en fonction des symétries dans le masque de chanfrein. L'ordre de complexité est de l'ordre de  $O(mL^n)$  pour  $m$  vecteurs si on limite le calcul de la fonction distance à une image de taille  $L^n$ .

É. RÉMY et É. THIEL utilisent une transformation en distance pour obtenir la fonction distance par propagation depuis l'origine. La carte de distance est calculée sur une image dont seul l'origine est un pixel de fond, les autres étant des pixels objet.

### Voisinage de test

É. THIEL a observé que l'ensemble des vecteurs du masque de chanfrein n'est pas adéquat pour le calcul de l'axe médian [173, p. 81]. Par exemple, avec la distance  $d_{14,20,31,44}$ ,  $\text{Lut}_{(2,1)}(291) = 321$  et  $\text{Lut}_{(2,1)}(321) = 352$ , mais la plus petite boule ouverte de centre  $O$  qui inclut  $D_{<}((4, 2), 291)$  est  $D_{<}(O, 351)$ . Cette relation d'inclusion n'est ni détectée par le vecteur  $\vec{c} = (2 \ 1)$  ni par les autres vecteurs du masque de chanfrein. É. RÉMY et É. THIEL ont alors introduit un voisinage de test, noté  $\mathcal{M}_{\text{Lut}}(r)$ , pour la description complète des relations d'inclusion [144].  $\mathcal{M}_{\text{Lut}}(r)$  est le plus petit ensemble de vecteurs suffisant pour la détection de l'axe médian dans les formes dont les rayons internes (taille de la plus grande boule) n'excèdent pas  $r$ . Dans le cas de la distance  $d_{14,20,31,44}$ , le vecteur  $(4 \ 2)$  n'est pas dans le masque de chanfrein mais il appartient au voisinage de test  $\mathcal{M}_{\text{Lut}}(r)$  pour  $r$  supérieur à 350.

Lorsque le voisinage de test est incomplet, des points imprévus sont inclus dans l'axe médian (non détection d'inclusion). Une méthode générale pour détecter et valider  $\mathcal{M}_{\text{Lut}}$  est basée sur le calcul de l'axe médian de toutes les boules [146]. Lorsque  $\mathcal{M}_{\text{Lut}}(r)$  est complet, l'axe médian est restreint au centre de la boule :

$$\text{MAT}_{D_{<}(O,r)}(p) = \begin{cases} r & \text{si } p = O, \\ 0 & \text{sinon.} \end{cases} \quad (4.4)$$

En revanche, si  $\mathcal{M}_{\text{Lut}}(r)$  est incomplet, il suffit de lui adjoindre les points additionnels de  $\text{MAT}_{D_{<}(O,r)}$ . Cette méthode de détermination de l'axe médian fonctionne dans toutes les dimensions mais elle a l'inconvénient d'être très coûteuse en temps de calcul.

## 4.3 Normes de chanfrein et $\mathcal{H}$ -polytopes

Le formalisme des  $\mathcal{H}$ -polytopes est fondamental pour cette méthode : il permet tout à la fois de représenter les boules, de tester leurs inclusions avec un coût constant et sans recourir au domaine image, et de vérifier les conditions de norme.

### 4.3.1 Géométrie des Boules de Chanfrein

G. BORGEFORS proposait en 1984 une description des boules de chanfrein sous forme de polygones en 2D et de polyèdres dans les dimensions supérieures [21, 23]. Mais dès 1968, U. MONTANARI décrivait des boules de distances pondérées à coefficients dans  $\mathbb{R}_+$  de cette manière [95]. L'optimisation des poids du masque de chanfrein était la motivation initiale de cette



représentation [21, 174, 180]. C. FOUARD et G. MALANDAIN, É. RÉMY, É. THIEL ont étudié la géométrie des boules de chanfrein pour elle-même et en lien avec d'autres propriétés de ces distances (*e.g.* les conditions de norme) [54, 144, 172].

L'objectif dans ce chapitre est de d'introduire des propriétés spécifiques de la  $\mathcal{H}$ -représentation des boules de chanfrein.

On peut déduire de (2.2) et (2.8) page 32 et page 36, une construction itérative des boules de chanfrein :

$$D_{\leq}(O, r) = D_{\leq}(O, r-1) \cup \bigcup_{0 \leq i \leq m} D_{\leq}(O + \vec{v}_i, r - w_i) . \quad (4.5)$$

Cette construction est suffisante pour obtenir une description exacte des boules de chanfrein lorsque celles-ci sont convexes, ce qui est notamment le cas pour les normes de chanfrein (section 4.5.1).

**Définition 4.3** (boule rationnelle). Considérons un masque de chanfrein  $\mathcal{M}$ , la *boule rationnelle unitaire* ou, simplement, *boule rationnelle*  $D_{\mathbb{Q}}$  est l'enveloppe convexe de l'ensemble des déplacements normalisés  $\vec{v}_i/w_i, (\vec{v}_i, w_i) \in \mathcal{M}$  :

$$\begin{aligned} D_{\mathbb{Q}} &= \text{conv} \left( \frac{\vec{v}_i}{w_i} : (\vec{v}_i; w_i) \in \mathcal{M} \right) \\ &= \left\{ \sum_{i=1}^m \alpha_i \frac{\vec{v}_i}{w_i} : \alpha_i \in \mathbb{R}_+, \sum_{i=1}^m \alpha_i = 1 \right\} . \end{aligned} \quad (4.6)$$

Notons que la boule rationnelle telle qu'elle est définie ici est convexe, *par définition*. En cela, elle diffère de la boule rationnelle équivalente décrite dans [54, 144, 172]. Une conséquence est que l'ajout dans le masque de pondérations inutiles (dont le poids est arbitrairement élevé) ne perturbe pas la détermination de la condition de norme, ce qui est conforme au fait que ces pondérations ne modifient pas la distance.

**Proposition 4.4.** L'homothétique de  $D_{\mathbb{Q}}$ ,  $rD_{\mathbb{Q}}$  contient  $D_{\leq}(O, r) : D_{\leq}(O, r) \subseteq rD_{\mathbb{Q}}$ .

*Démonstration.* Soit  $p$  un point de  $D_{\leq}(O, r)$  et soit  $\sum_{i=1}^m \lambda_i \vec{v}_i$  un chemin minimal depuis  $O$  vers  $p$ . Alors  $d(O, p) \leq r$  et  $d(O, p) = \sum_{i=1}^m \lambda_i w_i$ . On peut décrire  $p$  comme une combinaison de  $O, r\vec{v}_1/w_1, \dots, r\vec{v}_n/w_n$  :

$$p = \frac{r-d(O,p)}{r} \cdot O + \sum_{i=1}^m \frac{\lambda_i w_i}{r} \cdot \frac{r}{w_i} \vec{v}_i, \text{ avec } \frac{r-d(O,p)}{r} \geq 0, \frac{\lambda_i w_i}{r} \geq 0 \forall i \text{ et } \frac{r-d(O,p)}{r} + \sum_{i=1}^m \frac{\lambda_i w_i}{r} = 1.$$

Alors par définition de conv :

$$p \in \text{conv}(O, \frac{r}{w_1} \vec{v}_1, \dots, \frac{r}{w_n} \vec{v}_n) \subset rD_{\mathbb{Q}} \text{ et chaque point de } D_{\leq}(O, r) \text{ est dans } rD_{\mathbb{Q}}. \quad \square$$

**Définition 4.5** (cône). On appelle *cône* de l'ensemble  $U = \{\vec{u}_i, 1 \leq i \leq m\}$  de  $\mathbb{R}^d$ , l'ensemble des combinaisons à coefficients positifs des vecteurs  $\vec{u}_i$  [*e.g.*, 192] :

$$C(U) = \left\{ \sum_{i=1}^m \lambda_i \vec{u}_i, \lambda_i \geq 0, \lambda_i \in \mathbb{R}_+ \right\} .$$

Le *cône translaté*  $C_{o,U}$  d'*apex*, ou *sommet*,  $o$  est l'ensemble des points  $o + \vec{v} : \vec{v} \in \text{cone}(U)$ .

**Définition 4.6** (cône simplicial). Un cône est dit *simplicial* lorsqu'il est défini par des vecteurs linéairement indépendants. Sa dimension est donc égale au nombre des vecteurs qui le définissent.

Dans un cône simplicial, chaque vecteur est représenté par une combinaison à coefficients non négatifs unique des vecteurs qui définissent le cône.

$$p \in C_{o,U} \Rightarrow \vec{op} = \sum_{i=1}^m \lambda_i \vec{u}_i, \text{ où } \{\lambda_i\} \in \mathbb{R}_+^m \text{ est unique.}$$

**Définition 4.7** (parallélépipède fondamental). Soit  $C(U)$  un cône simplicial engendré par  $U = \{\vec{u}_i, 1 \leq i \leq m\}$ , on appelle parallélépipède fondamental, l'ensemble :

$$\Pi(U) = \left\{ \sum_{i=1}^m \lambda_i \vec{u}_i, 0 \leq \lambda_i < 1 \right\} .$$

**Définition 4.8** (cône unimodulaire). Soit un cône simplicial  $C(U)$  défini par  $m$  vecteurs de  $\mathbb{Z}^n$ .  $C(U)$  est dit *unimodulaire* si son parallélépipède fondamental contient un et un seul point de  $\mathbb{Z}^n$ .

Chaque point discret d'un cône unimodulaire est atteint par une combinaison entière unique de ses vecteurs.

**Définition 4.9.** Soit  $\vec{v}$  tel que  $\mathcal{F} = H(D_{\mathbb{Q}}, \vec{v}) \cap D_{\mathbb{Q}}$  est une facette de  $D_{\mathbb{Q}}$ . On note  $\vec{n}_{\mathcal{F}}$  l'unique vecteur orthogonal à la facette tel que  $\forall p \in \mathcal{F}, \vec{n}_{\mathcal{F}} \cdot p = 1$  :

$$\vec{n}_{\mathcal{F}} = \frac{\vec{v}}{h(D_{\mathbb{Q}}, \vec{v})} .$$

Dans le cône simplicial engendré par les vecteurs  $\frac{\vec{v}_1}{w_1} \dots \frac{\vec{v}_n}{w_n}$  :

$$\vec{n}_{\mathcal{F}} \cdot \left( \frac{\vec{v}_1}{w_1} \dots \frac{\vec{v}_n}{w_n} \right) = (1 \dots 1) .$$

On a donc :

$$\vec{n}_{\mathcal{F}} = (w_1, \dots, w_n) \cdot (\vec{v}_1 | \dots | \vec{v}_n)^{-1} . \quad (4.7)$$

Le vecteur  $\alpha = (\vec{v}_1 | \dots | \vec{v}_n)^{-1} p$  forme l'unique combinaison linéaire des vecteurs  $\vec{v}_1, \dots, \vec{v}_n$  qui produit  $p$ . Si  $\alpha$  a des coefficients entiers alors il existe un chemin  $\alpha(\vec{v}_1 | \dots | \vec{v}_n)$  vers  $p$  et sa longueur est  $(w_1, \dots, w_n) \cdot \alpha = \vec{n}_{\mathcal{F}} \cdot p$ . Le vecteur normal est équivalent au *déplacement élémentaire* et au *gradient discret* du cône qui intersecte  $\mathcal{F}$  tels qu'ils sont définis pour les normes de chanfrein dans [172] et [144]. Notons, qu'ici, le vecteur normal est défini pour un masque de chanfrein quelconque, qu'il induise une norme ou non.

Par exemple, pour la norme de chanfrein  $d_{5,7,11}$ , le point  $(3, 1)$  est dans le cône engendré par les vecteurs  $\vec{a} = (1 \ 0)$  et  $\vec{c} = (2 \ 1)$  dont les poids associés sont 5 et 11. La distance de l'origine vers ce point est donc [(4.32) 172] :

$$d(O, (3, 1)) = \begin{pmatrix} 5 & 11 \end{pmatrix} \cdot \begin{pmatrix} 1 & 2 \\ 0 & 1 \end{pmatrix}^{-1} \cdot \begin{pmatrix} 3 \\ 1 \end{pmatrix} = \begin{pmatrix} 5 & 1 \end{pmatrix} \cdot \begin{pmatrix} 3 \\ 1 \end{pmatrix} = 16 .$$

### 4.3.2 Condition de norme

**Proposition 4.10** (Borne minimale de distance). Soient  $\mathcal{F}$  une facette de la boule rationnelle et  $\vec{n}_{\mathcal{F}}$  son vecteur normal, alors :

$$d(O, p) \geq \vec{n}_{\mathcal{F}} \cdot p . \quad (4.8)$$

*Démonstration.* Soit  $p = \sum_i \lambda_i \vec{v}_i$  un chemin minimal vers  $p$ , autrement dit  $d(O, p) = \sum_i \lambda_i w_i$ . La facette  $\mathcal{F}$  a comme hyperplan d'appui  $\{x \in \mathbb{R}^n : d_C(x) = 1\}$ . Par convexité, la boule unitaire est incluse dans le demi-espace  $\{x \in \mathbb{R}^n : d_C(x) \leq 1\}$ . En particulier pour les sommets de la boule unitaire  $\vec{v}_i/w_i : d_C(\vec{v}_i/w_i) \leq 1$ . Par linéarité,  $d_C(\vec{v}_i) \leq w_i$  et  $d_C((\sum_i \lambda_i \vec{v}_i)) = \sum_i \lambda_i d_C(\vec{v}_i) \leq \sum_i \lambda_i w_i$ . Donc  $d_C(\sum_i \lambda_i \vec{v}_i)$  est toujours inférieur ou égal à la longueur du chemin  $\sum_i \lambda_i \vec{v}_i$ .  $\square$

Cette borne inférieure vaut pour tout point  $p$ , qu'il soit ou non dans le cône qui intersecte  $\mathcal{F}$  et sans que nécessairement le masque induise une norme.

**Proposition 4.11** (Homogénéité dans un cône unimodulaire). Soit  $C_{O,U}$  un cône unimodulaire qui intersecte une facette unique  $\mathcal{F}$  de  $D_{\mathbb{Q}}$ . La distance  $d$  est homogène dans le cône et égale à  $d_C(p), \forall p \in C_{O,U}$  si et seulement si :

$$d(O, O + \vec{u}) = d_C(\vec{u}), \forall \vec{u} \in U . \quad (4.9)$$

*Démonstration.* Selon la proposition 4.10,

$d(O, p) \geq d_C(p), \forall p \in C_{O,U}$ . Comme  $C_{O,U}$  est unimodulaire,  $p$  peut être atteint par une combinaison à coefficients entiers non négatifs des vecteurs de  $U$ ,  $\vec{Op} = \sum \lambda_i \vec{u}_i$ . D'après l'inégalité triangulaire et les hypothèses,  $d(O, p) \leq \sum \lambda_i d(O, O + \vec{u}_i) = \sum \lambda_i d_C(\vec{u}_i) = d_C(p)$ . Donc  $d_C(p)$  est une borne simultanément inférieure et supérieure de  $d(O, p)$ , donc l'égalité est vérifiée et  $d$  est homogène dans le cône  $C_{O,U}$ .

Inversement, considérons un vecteur  $\vec{u}$  de  $U$  tel que  $d(O, p) \neq p \cdot \vec{n}_{\mathcal{F}}$ , donc  $d(O, p) > p \cdot \vec{n}_{\mathcal{F}}$ , par la proposition 4.10. Dans la facette  $\mathcal{F}$ , on peut toujours trouver  $n = \dim(\mathcal{F}) + 1$  sommets  $\vec{v}_1/w_1 \dots \vec{v}_n/w_n$  tels que  $\vec{u}$  est dans le cône simplicial engendré par les vecteurs  $\vec{v}_1 \dots \vec{v}_n$ . En conséquence,  $\vec{u}$  peut être représenté par une combinaison unique à coefficients non négatifs de  $n$  vecteurs linéairement indépendants :  $\vec{u} = \alpha \cdot (\vec{v}_1 | \dots | \vec{v}_n)$ . Les composantes de  $\alpha = (\vec{v}_1 | \dots | \vec{v}_n)^{-1} \cdot \vec{u}$  ne sont pas nécessairement entières (elles le sont si  $\Delta = |\det(\vec{v}_1 | \dots | \vec{v}_n)| = 1$ ). Cependant,  $\Delta(\vec{v}_1 | \dots | \vec{v}_n)^{-1}$  a des coefficients entiers donc  $\Delta \vec{u}$  peut être uniquement représenté par une combinaison entière des vecteurs  $\vec{v}_i$  :  $\Delta \vec{u} = \Delta(\vec{v}_1 | \dots | \vec{v}_n) \cdot \alpha$ , il existe donc un chemin vers  $O + \Delta \vec{u}$  dont la longueur est :  $\Delta(w_1 \dots w_n) \cdot \alpha = \Delta(w_1 \dots w_n) \cdot (\vec{v}_1 | \dots | \vec{v}_n)^{-1} \cdot \vec{u} = \Delta \vec{n}_{\mathcal{F}} \cdot p \geq d(O, \Delta \vec{u})$ . Donc  $d(O, O + \Delta \vec{u}) < \Delta d(O, O + \vec{u})$  et la distance n'est pas homogène dans le cône.  $\square$

Selon la proposition 4.11, l'homogénéité dans un cône unimodulaire de dimension  $n$  peut être déterminée uniquement en vérifiant les vecteurs du cône (qui ne sont pas nécessairement dans le masque de chanfrein). Le corollaire 4.12 fournit une condition nécessaire et suffisante de norme (une condition suffisante basée sur une triangulation unimodulaire est donnée dans [172]).

**Corollaire 4.12** (Condition de norme). Soient  $\{U_i\}$  les ensembles de vecteurs  $U_i$  avec les propriétés suivantes :

- chaque cône unimodulaire  $C_i = C_{O,U_i}$  intersecte une facette unique de  $D_{\mathbb{Q}}$  ;
- la réunion des cônes couvre l'espace entier  $\mathbb{Z}^n$ .

Le masque de chanfrein  $\mathcal{M}$  induit une norme si et seulement si :

$$d(O, O + \vec{u}) = d_C(\vec{u}), \forall \vec{u} \in \bigcup_i U_i . \quad (4.10)$$

**Proposition 4.13** (Formulation directe de la distance). Si  $d$  induit une norme, la distance de  $O$  à un point  $p$  quelconque s'écrit :

$$d(O, p) = \max_{1 \leq i \leq l} \{ \vec{n}_{\mathcal{F}_i} \cdot p \} \quad (4.11)$$

où  $l$  est le nombre de facettes de  $D_{\mathbb{Q}}$  et  $\vec{n}_{\mathcal{F}_i}$  est défini selon la définition 4.9 pour la  $i^e$  facette de  $D_{\mathbb{Q}}$ ,  $\mathcal{F}_i$ . Cette formule close ne nécessite pas de déterminer préalablement dans quel cône est le point  $p$ .

*Démonstration.* L'inégalité  $d(O, p) \geq \overline{n_{\mathcal{F}_i}} \cdot p$  est vraie pour toutes les facettes d'après la proposition 4.10 et l'égalité  $d(O, p) = \overline{n_{\mathcal{F}_i}} \cdot p$  est vraie, selon la proposition 4.11, pour toutes les facettes de  $D_{\mathbb{Q}}$  (il en existe au moins une) qui intersectent  $[O, p]$ .  $\square$

### 4.3.3 $\mathcal{H}$ -représentation des boules de normes de chanfrein

La  $\mathcal{H}$ -représentation des boules de chanfrein pour les normes découle directement de (4.11) :

$$p \in D_{\leq}(O, r) \Leftrightarrow \max_{1 \leq i \leq l} \{d_{C_i}(p)\} \leq r \Leftrightarrow A_{\mathcal{M}} \cdot p \leq y \quad (4.12)$$

où la matrice  $A_{\mathcal{M}}$  dépend seulement du masque  $\mathcal{M}$ . Elle contient autant de lignes que  $D_{\mathbb{Q}}$  de facettes. Chacune de ses lignes est égale au vecteur  $\overline{n_{\mathcal{F}_i}}$  correspondant à la  $i^{\text{e}}$  facette, cf. (4.7).  $y$  est un vecteur colonne dont toutes les composantes sont égale au rayon  $r$ .

**Exemple 4.14.** La matrice de la  $\mathcal{H}$ -représentation des boules de la distance  $d_{5,7,11}$  restreinte au cône générateur  $\mathcal{G}$  est

$$A_{\mathcal{M}} = \begin{pmatrix} 5 & 1 \\ 4 & 3 \end{pmatrix}$$

où (5 1) and (4 3) sont les vecteurs normaux des deux arêtes de  $D_{\mathbb{Q}}$  et

$$D_{\leq}(O, r) = \left\{ p \in \mathbb{Z}^n : \begin{pmatrix} 5 & 1 \\ 4 & 3 \end{pmatrix} \cdot p \leq \begin{pmatrix} r \\ r \end{pmatrix} \right\} .$$

**Proposition 4.15** (Point le plus éloigné). Soit  $A_{\mathcal{M}}$  la matrice de la  $\mathcal{H}$ -représentation de la boule rationnelle d'une norme de chanfrein engendrée par le masque  $\mathcal{M}$ . Le point le plus éloigné de l'origine dans le  $\widehat{\mathcal{H}}$ -polytope  $P = \{p : A_{\mathcal{M}} \cdot p \leq \widehat{\mathcal{H}}(P)\}$  est à une distance égale à la plus grande composante de  $\widehat{\mathcal{H}}(P)$ .

*Démonstration.* Par construction de  $A_{\mathcal{M}}$ , (4.11) est équivalente à  $d(O, p) = \max_i \{A_{\mathcal{M}_i} \cdot p\}$ , donc :

$$\max_{p \in P} \{d(O, p)\} = \max_{p \in P} \left\{ \max_{1 \leq i \leq l} \{A_{\mathcal{M}_i} \cdot p\} \right\} = \max_{1 \leq i \leq l} \{\widehat{\mathcal{H}}_i(P)\} .$$

$\square$

**Proposition 4.16** (Boule couvrante minimale). Le rayon de la plus petite boule centrée sur  $O$  qui contient tous les points d'un  $\widehat{\mathcal{H}}$ -polytope  $P$  représenté par la matrice  $A_{\mathcal{M}}$  et le vecteur  $\widehat{\mathcal{H}}(P)$  est égal à la plus grande composante de  $\widehat{\mathcal{H}}(P)$ .

*Démonstration.* La plus petite boule qui inclut le polytope  $P$  doit contenir son point le plus éloigné depuis l'origine :

$$\min \{r \in \mathbb{N} : P \subseteq D_{\leq}(O, r)\} = \max_{p \in P} \{d(O, p)\} = \max_{1 \leq i \leq l} \{\widehat{\mathcal{H}}_i(P)\} .$$

$\square$

**Définition 4.17** (transformation en couverture). On appelle *transformation en couverture* d'un ensemble fini de points  $X$  de  $\mathbb{Z}^n$ , la fonction  $\mathcal{C}_X$  qui associe à chaque point  $p$  de  $\mathbb{Z}^n$ , le rayon de la plus petite boule centrée en  $p$  qui inclut  $X$  :

$$\begin{aligned} \mathcal{C}_X : \mathbb{Z}^n &\rightarrow \mathbb{N} \\ \mathcal{C}_X(p) &= \min \{r : X \subseteq D_{\leq}(p, r)\} \end{aligned} .$$

La transformation en couverture de la boule de chanfrein  $D_{\leq}(O, r)$  au point  $p$  est le rayon de la plus petite boule de centre  $p$  qui inclut  $D_{\leq}(O, r)$ , ou, par symétrie centrale de la distance de chanfrein, le rayon de la plus petite boule de centre  $O$  qui couvre  $D_{\leq}(p, r)$  et donc la valeur de la composante maximale de la  $\widehat{\mathcal{H}}$ -representation de  $D_{\leq}(p, r)$  :

$$\mathcal{C}_{D_{\leq}(O, r)}(p) = \max \{ \widehat{\mathcal{H}}(D_{\leq}(p, r)) \} = \max \{ \widehat{\mathcal{H}}(D_{\leq}(O, r)) + A_{\mathcal{M}} \cdot p \} . \quad (4.13)$$

Il est intéressant de noter que la transformation en couverture du disque de rayon nul est égale à la fonction distance, de même que la transformation en distance du complémentaire de ce disque :

$$\mathcal{C}_{D_{\leq}(O, 0)}(p) = \text{DT}_{\mathbb{Z}^n \setminus \{O\}}(p) = d(p, O) = d(O, p) .$$

**Définition 4.18** (cône de couverture). Un *cône de couverture*  $C_{o, U}$  dans la transformation en couverture de  $X$ ,  $\mathcal{C}_X$  est un cône unimodulaire engendré par les vecteurs  $U$  et translaté par  $o$  tel que chaque boule de centre  $p$  est incluse dans la boule  $p + \vec{u}$ ,  $\vec{u} \in U$  :

$$\forall p \in C_{o, U}, \forall \vec{u} \in U, D_{\leq}(p, \mathcal{C}_X(p)) \subsetneq D_{\leq}(p + \vec{u}, \mathcal{C}_X(p + \vec{u})) .$$

**Proposition 4.19.** Si  $C_{o, U}$  est un cône de couverture de  $\mathcal{C}_{D_{\leq}(O, r)}$  alors pour tout point  $q$  dans  $C_{o, U} \setminus U \setminus \{o\}$  il existe  $p$  tel que la boule centrée dans  $q$  inclut la boule de centre  $p$  :

$$D_{\leq}(O, r) \subsetneq D_{\leq}(p, \mathcal{C}_{D_{\leq}(O, r)}(p)) \subsetneq D_{\leq}(q, \mathcal{C}_{D_{\leq}(O, r)}(q)) .$$

*Démonstration.* Si  $U$  n'est pas vide, il existe  $\vec{u} \in U$  tel que  $p = q - \vec{u}$  soit dans  $C_{o, U} \setminus \{o\}$ . Alors, par définition de  $C_{o, U}$ ,  $D_{\leq}(p, \mathcal{C}_{D_{\leq}(O, r)}(p)) \subsetneq D_{\leq}(q, \mathcal{C}_{D_{\leq}(O, r)}(q))$ . De plus,  $D_{\leq}(O, r) \subseteq D_{\leq}(p, \mathcal{C}_{D_{\leq}(O, r)}(p))$  est toujours vérifié par définition de  $\mathcal{C}_{D_{\leq}(O, r)}$ .  $\square$

La proposition 4.19 est utile pour limiter la recherche de boules couvrant directement  $D_{\leq}(O, r)$  dans la fonction de couverture  $\mathcal{C}_{D_{\leq}(O, r)}$ . Quand un cône de couverture est détecté, seul son sommet  $o$  ou les vecteurs du cône (if  $o = O$ ) doivent être vérifiés.

**Proposition 4.20.** S'il existe un entier  $j \in [1 \dots l]$  et un point  $o$  tels que  $A_{\mathcal{M}_j} \cdot \vec{u}_i, \forall \vec{u}_i \in U$  et  $\widehat{\mathcal{H}}_j(D_{\leq}(o, r))$  sont maximaux alors  $C_{o, U}$  est un cône de couverture dans  $\mathcal{C}_{D_{\leq}(O, r)}$ .

*Démonstration.* Soit  $j$  l'indice de la composante maximale de  $\widehat{\mathcal{H}}(D_{\leq}(o, r))$  et  $A_{\mathcal{M}} \cdot \vec{u}_i, \forall \vec{u}_i \in U$ , alors  $j$  est la composante maximale de toute combinaison linéaire positive de ces vecteurs. Soit  $p$  un point quelconque de  $C_{o, U}$ ,  $p = o + \sum_i \lambda_i \vec{u}_i$ ,  $\lambda_i \in \mathbb{N}$ . D'après (4.13), on déduit que, dans  $C_{o, U}$ ,  $\mathcal{C}_{D_{\leq}(O, r)}$  prend la forme d'une fonction affine des composantes  $\lambda_i$  :

$$\begin{aligned} \mathcal{C}_{D_{\leq}(O, r)}(p) &= \max \left\{ \widehat{\mathcal{H}}(D_{\leq}(o, r)) + \sum \lambda_i A_{\mathcal{M}} \vec{u}_i \right\} \\ &= \widehat{\mathcal{H}}_j(D_{\leq}(o, r)) + \sum \lambda_i A_{\mathcal{M}_j} \vec{u}_i . \end{aligned} \quad (4.14)$$

De la même façon, le rayon de la boule minimale de centre  $p + \vec{u}$ ,  $\vec{u} \in U$  qui couvre  $D_{\leq}(O, r)$  est :

$$\begin{aligned} \mathcal{C}_{D_{\leq}(O, r)}(p + \vec{u}) &= \widehat{\mathcal{H}}_j(D_{\leq}(o, r)) + \sum \lambda_i A_{\mathcal{M}_j} \vec{u}_i + A_{\mathcal{M}_j} \vec{u} \\ &= \mathcal{C}_{D_{\leq}(p, \mathcal{C}_{D_{\leq}(O, r)}(p))}(O + \vec{u}) . \end{aligned} \quad (4.15)$$

La boule  $D_{\leq}(p, \mathcal{C}_{D_{\leq}(O, r)}(p))$  est contenue dans  $D_{\leq}(p + \vec{u}, \mathcal{C}_{D_{\leq}(O, r)}(p + \vec{u}))$  et  $C_{o, U}$  est un cône de couverture selon la définition 4.18.  $\square$

## 4.4 Triangulation des boules de chanfrein 3D

La première étape de la méthode consiste à décomposer l'espace (restreint au cône générateur si la symétrie s'applique) en cônes unimodulaires qui intersectent chacune une facette unique de la boule rationnelle. Cette triangulation a trois objectifs :

1. calculer les vecteurs normaux aux facettes selon (4.7). Ces vecteurs constituent les lignes de la matrice  $A_{\mathcal{M}}$  commune à la  $\mathcal{H}$ -représentation de toutes les boules quel que soit leur rayon,
2. fournir un ensemble de cônes unimodulaires qui guidera la recherche des vecteurs du voisinage de test,
3. vérifier que le masque de chanfrein induit une norme en vérifiant la propriété d'homogénéité dans chacun des cônes.

L'extraction de l'enveloppe convexe des déplacements normalisés  $\vec{v}_i/w_i$  joue un rôle central dans cette étape. En premier lieu parce que la boule rationnelle est une enveloppe convexe, selon la définition 4.3. Mais également parce que l'extraction de l'enveloppe convexe répond à ces trois objectifs.

### 4.4.1 Sommets de l'enveloppe convexe dans le cône générateur

Le résultat escompté est une triangulation d'une partie de la boule rationnelle  $D_{\mathbb{Q}}$  située dans le cône générateur  $\mathcal{G}$ . Les données sont constituées de l'ensemble des vecteurs de déplacement du masque  $\mathcal{M}$  divisée par leurs poids correspondants (pour former la boule rationnelle unitaire). Dans l'algorithme ces points sont représentés par des pondérations, et tous les calculs sont effectués en arithmétique entière. Les pondérations du masque ne sont pas les seuls sommets de l'intersection de  $D_{\mathbb{Q}}$  avec  $\mathcal{G}$ .

L'intersection de  $D_{\mathbb{Q}}$  avec  $\mathcal{G}$  introduit de nouveaux sommets qui sont les intersection des arêtes de  $D_{\mathbb{Q}}$  avec les faces de  $\mathcal{G}$ . La figure figure 4.22a en donne une illustration. Les vecteurs  $\vec{e} = (2 \ 1 \ 1)$  et  $\vec{d} = (4 \ 1 \ 1)$  ne sont pas dans le masque de chanfrein et ne sont donc pas des sommets de  $D_{\mathbb{Q}}$ , cependant ils apparaissent dans l'intersection de  $D_{\mathbb{Q}}$  avec  $(O, \{\vec{a}, \vec{c}\})$ . Puisque les arêtes de  $D_{\mathbb{Q}}$  ne sont pas connues à cette étape de la méthode, toutes les intersections de faces de  $\mathcal{G}$  avec les segments formés par des couples de sommets rationnels de  $D_{\mathbb{Q}}$  sont ajoutées dans les données d'entrée de l'algorithme de triangulation. Ces points sont des combinaisons rationnelles de sommets et sont également représentés comme des pondérations, avec des coefficients positifs. Parmi ces points, la plupart sont intérieurs à  $D_{\mathbb{Q}}$  et seront éliminés pendant l'extraction de l'enveloppe convexe.

### 4.4.2 Enveloppe convexe et triangulation de la boule rationnelle

Parmi les nombreuses méthodes d'extraction d'enveloppe convexe disponibles, nous avons choisi de mettre en œuvre celle qui est connue sous le nom de *gift wrapping*. Son principe consiste à envelopper l'ensemble des points par un « papier d'emballage » en faisant pivoter un hyperplan autour des axes constitués par les crêtes déjà identifiées de l'enveloppe convexe. D. R. CHAND et S. S. KAPUR, pour le cas général, et R. A. JARVIS, pour le cas 2D, ont proposé indépendamment des algorithmes basés sur ce mécanisme [25, 81].

Lors de chaque itération, un hyperplan d'appui est identifié en pivotant autour d'une crête préalablement détectée. Les nouveaux points détectés dans cet hyperplan d'appui forment une facette de l'enveloppe convexe. Chaque crête de cette facette est ajoutée à une liste pour servir ultérieurement de pivot ou supprimée de cette liste si elle y était déjà présente. Cette manière de procéder garantit que les deux facettes adjacentes à chaque crête sont identifiées.

Une modification simple de l'algorithme [25] permet de produire une triangulation de l'enveloppe, c'est-à-dire une partition en cônes simpliciaux. De surcroît, une décomposition unimodulaire de ces cônes simpliciaux ainsi que le treillis des faces peuvent aisément être extraits comme sous-produits de cet algorithme.

---

**Algorithme 4.21:** Algorithme *gift wrapping* simplifié [25]

---

```

Entrée : Un ensemble de points  $S$ 
// Choix d'une arête initiale  $(p, q)$ 
// empiler  $(p, q)$  dans  $\ell$ , the list of active edges
 $\ell \leftarrow \{(p, q)\}$ 
tant que  $L$  est non vide faire
| dépiler  $(p, q)$  de  $\ell$ 
| prendre  $r$  dans  $S \setminus \{p, q\}$ 
|  $S' \leftarrow S \setminus \{p, q, r\}$ 
| pour chaque  $s \in S'$  faire
| | si  $s$  est au-dessus de  $(p, q, r)$  ou si  $(p, q, r, s)$  sont coplanaires et  $(p, q, r)$  est plus
| | petit que  $(p, q, r)$  alors
| | |  $r \leftarrow s$ 
| | fin
| fin
| si  $\det(p, q, r) = 1$  alors
| | //  $C_{O, \{p, q, r\}}$  est un cône unimodulaire de la triangulation
| | si  $(r, p) \in \ell$  alors  $\ell \leftarrow \ell \setminus \{(r, p)\}$  sinon  $\ell \leftarrow \ell \cup \{(r, p)\}$ 
| | si  $(q, r) \in \ell$  alors  $\ell \leftarrow \ell \setminus \{(q, r)\}$  sinon  $\ell \leftarrow \ell \cup \{(q, r)\}$ 
| sinon
| | Insérer un point discret  $s$  de  $C_{O, \{p, q, r\}}$  dans  $S$ 
| fin
fin

```

---

Après avoir choisi une première arête, l'algorithme 4.21 étend de manière itérative l'enveloppe en ajoutant un nouveau triangle et ses côtés. Puisque l'enveloppe convexe est calculée dans le cône  $\mathcal{G}$ , une extrémité de l'arête initiale peut être choisie comme le point sur le premier axe de  $\mathcal{G}$  qui est le plus éloigné de l'origine :  $p = (x_1, 0, 0)$  (avec  $x_1$  maximal si plusieurs points sont présents sur cet axe) et le second sommet est le point  $q$  qui maximise l'angle  $(O, p, q)$  sur une facette de  $\mathcal{G}$  qui contient  $p$ .

### 4.4.3 Triangulation unimodulaire

Chaque cône simplicial  $C_{O, \{p, q, r\}}$  produit par l'algorithme 4.21 est unimodulaire si et seulement si le déterminant des vecteurs qui l'engendre est unitaire,  $\det(p, q, r) = \pm 1$ . Lorsque  $|\det(p, q, r)| = 1$ , le cône détecté est inclus dans la triangulation sinon un point discret  $s$  de son parallélépipède fondamental est ajouté dans le masque en entrée de l'algorithme. On a nécessairement  $|\det(p, q, s)| < |\det(p, q, r)|$  ce qui garantit qu'un cône unimodulaire est inséré après un nombre fini d'itérations.

L'intersection de la boule rationnelle pour le masque 7, 8, 11, 14 avec le cône générateur  $\mathcal{G}$  est représenté sur la figure 4.22b. La face  $(\vec{b} = (1 \ 1 \ 0), \vec{e} = (2 \ 1 \ 1), \vec{o} = (4 \ 1 \ 1), \vec{d} = (2 \ 1 \ 0))$  n'a pas de décomposition unimodulaire sans insertion de point nouveau car  $\det(\vec{d}, \vec{e}, \vec{o}) = \det(\vec{b}, \vec{e}, \vec{o}) = 2$ .

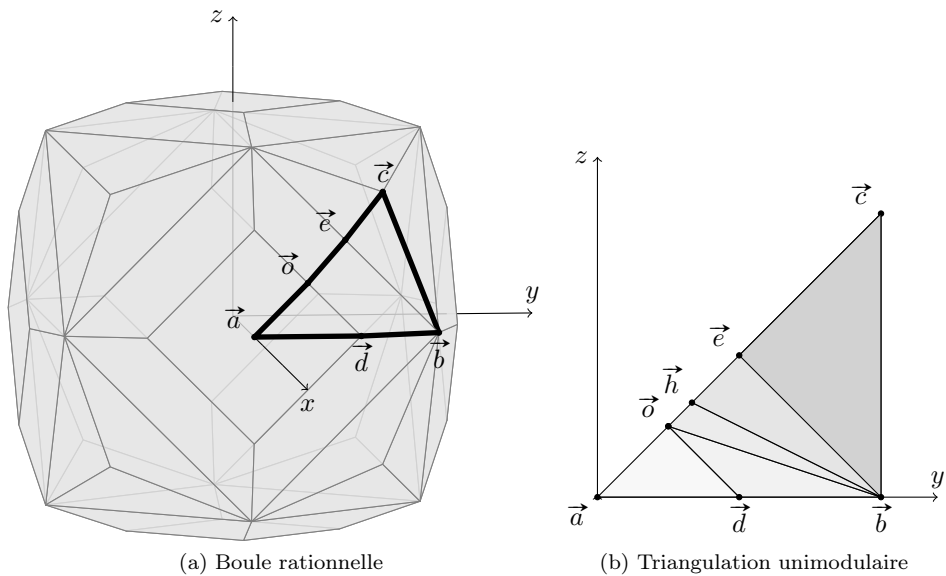


FIGURE 4.22 – Boule rationnelle  $D_{\mathbb{Q}}$  du masque 3D 7,8,11,14 et une triangulation unimodulaire  $D_{\mathbb{Q}} \cap \mathcal{G}$ . (b) Triangulation unimodulaire de  $D_{\mathbb{Q}} \cap \mathcal{G}$  (lignes épaisses dans (a)) en cônes 3D, représentés par des triangles. Chaque point de  $\mathbb{Z}^3 (x, y, z)$  est projeté sur  $(y/x, z/x)$  (espace de FAREY).

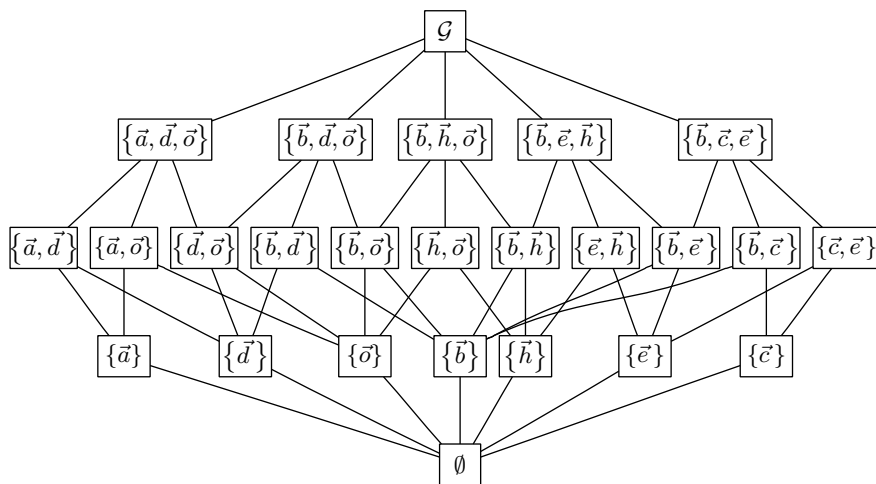


FIGURE 4.23 – Treillis des faces de la triangulation unimodulaire de  $D_{\mathbb{Q}} \cap \mathcal{G}$  (masque de chanfrein 7,8,11,14).



En ajoutant le vecteur  $\vec{h} = (3 \ 1 \ 1)$ , cette face est décomposée en deux cônes unimodulaires. Le treillis des faces de cette triangulation est présenté sur la figure 4.23.

#### 4.4.4 Test d'homogénéité

La triangulation unimodulaire de la boule rationnelle  $D_{\mathbb{Q}}$  produit un ensemble de cônes et leurs vecteurs normaux. Dans chacun des cônes unimodulaires, la comparaison de la distance  $d(O, O + \vec{u})$ , obtenue par propagation des poids du masque de chanfrein, avec  $d_{C_i}(\vec{u})$ , où  $\vec{n}_{\mathcal{F}}$  est le vecteur normal d'une facette qui intersecte  $[O, \vec{u})$ , permet de déterminer, en application du corollaire 4.12, si la distance est homogène dans ce cône. Si l'égalité est observée dans tous les cônes, on peut conclure que le masque engendre une norme. Dans le cas contraire, il suffit d'ajouter les vecteurs défectueux, combinés avec les distances propagées, pour obtenir un masque qui engendre une norme. Notons que ces résultats sont valides quelle que soit la triangulation unimodulaire obtenue, celle-ci n'étant généralement pas unique.

### 4.5 Voisinage de test et tables d'inclusion pour les normes de chanfrein

#### 4.5.1 $\widehat{\mathcal{H}}$ -représentation des boules de chanfrein

Le calcul des LUT utilise les représentations en  $\mathcal{H}$ -polytopes minimales des boules de chanfrein. Celles-ci partagent la même matrice  $A_{\mathcal{M}}$  (4.12), qui dépend du masque de chanfrein (4.12) et est obtenue par l'algorithme de triangulation. Les  $\widehat{\mathcal{H}}$ -coefficients sont obtenus itérativement par l'application de (3.19) et (4.5) à partir de la boule de rayon nul  $D_{\leq}(O, 0) = \{x : A_{\mathcal{M}} \cdot x = 0\}$  :

$$\widehat{\mathcal{H}}(D_{\leq}(O, r)) = \max \left\{ \widehat{\mathcal{H}}(D_{\leq}(O, r-1)); \widehat{\mathcal{H}}(D_{\leq}(O, r-w_i)) + A_{\mathcal{M}} \vec{v}_i, i \in [1..m] \right\} .$$

#### 4.5.2 Valeurs des tables d'inclusion

Les valeurs des tables d'inclusion sont directement obtenue à partir de la transformation en couverture (4.13) :

$$\text{Lut}_{\vec{v}_i}[r] = 1 + \mathcal{C}_{D_{\leq}(O, r-1)}(O + \vec{v}_i) = 1 + \max \left\{ \widehat{\mathcal{H}}(D_{\leq}(O, r-1)) + A_{\mathcal{M}} \cdot \vec{v}_i \right\} .$$

#### 4.5.3 Voisinage de test

L'algorithme 4.25 construit le voisinage de test en commençant avec un ensemble de vecteurs  $\mathcal{M}_{\text{Lut}}$  vide puis recherche les relations d'inclusion directes de boules de rayon croissant de manière similaire à la méthode de É. RÉMY et É. THIEL [146]. Cependant, dans [146], les relations d'inclusions sont considérées du point de vue des boules englobantes (dans la transformation en distance, ce qui nécessite la propagation des distances), tandis que notre méthode se situe du point de vue des boules incluses, *i.e.*, dans la transformation en ouverture que l'on peut calculer en tout point, indépendamment du voisinage et donc sans propagation avec (4.13). L'algorithme utilise le treillis des cônes  $\mathcal{L}$  (figure 4.23), produit de la phase de triangulation, comme une partition de  $\mathcal{G}$ . Chaque cône qui n'est pas couvrant est partitionné par la procédure `visitCone` comme

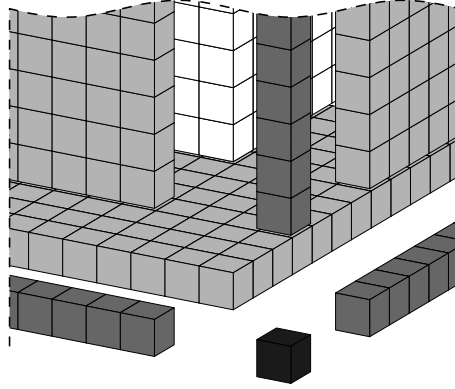


FIGURE 4.24 – Partition d'un cône unimodulaire 3D en  $2^3$  sous-cônes (du plus sombre au plus clair) :  $C_0^3 = 3 = 1$  point,  $C_1^3 = 3$  cônes 1D,  $C_2^3 = 3$  cônes 2D et  $C_3^3 = 1$  cônes 3D, avec  $\sum_{i=0}^3 C_i^3 = 2^3$ .

illustré dans la figure 4.24 : un cône  $n$ D  $C_{o,U}$  est divisé en  $2^n$  sous-cônes correspondant à tous les sous-ensembles de  $U$ . Les relations d'inclusion directes aux sommets des cônes visités sont vérifiées par la fonction `visitPoint` avec un calcul direct de la fonction de couverture pour tous les voisinages connus de  $\mathcal{M}_{\text{Lut}}$ .

---

**Algorithme 4.25:** Détermination du voisinage de test  $\mathcal{M}_{\text{Lut}}$

---

**Entrée :**  $\mathcal{L}$  treillis des cônes unimodulaires

**Entrée :**  $R$  rayon maximal

**Sortie :**  $\mathcal{M}_{\text{Lut}}$  voisinage de test

$\mathcal{M}_{\text{Lut}} \leftarrow \emptyset$

**pour**  $r \leftarrow 1$  à  $R$  **faire** // rayon de la boule incluse

**pour chaque**  $C \in \mathcal{L}$  *par dimension croissante* **faire**

$o \leftarrow \sum_{\vec{u} \in U} \vec{u}$   
        `visitCone`( $C_{o,U}$ ,  $\mathcal{M}_{\text{Lut}}$ ,  $r$ )

**fin**

**fin**

---

## 4.6 Résultats

Ces algorithmes ont été mis en œuvre en langage C. Ils produisent un résultat dans le même format que l'algorithme de référence [146] de façon à pouvoir comparer les sorties au caractère près. Des tests systématiques ont été conduits sur des masques de chanfrein variés et pour différents rayons maximaux. Les résultats sont identiques pour les deux algorithmes sauf pour des rayons proches de la borne maximale spécifiée pour lesquels la boule couvrante dépasse le rayon maximal. Bien que les deux algorithmes aient un traitement différent dans ce cas, la différence n'a pas d'impact sur l'extraction de l'axe médian.

Les temps de calcul des deux méthodes sont donnés dans le 4.26 pour plusieurs masques et

---

**Procédure** visitCone( $C_{o,U}, \mathcal{M}_{\text{Lut}}, r$ )

---

**Entrée** :  $C_{o,U}, \mathcal{M}_{\text{Lut}}$ , inner radius  $r$   
**visitPoint** ( $o, \mathcal{M}_{\text{Lut}}, r$ )  
**si non** isCoveringCone( $C_{o,U}$ ) **alors**  
    **pour chaque** face  $C_{o,U'}$  de  $C_{o,U}$  **faire**  
         $o' \leftarrow o + \sum_{\vec{u} \in U'} \vec{u}$ ;  
        **visitCone**( $C_{o',U'}, \mathcal{M}_{\text{Lut}}, r$ );  
    **fin**  
**fin**

---

---

**Procédure** visitPoint( $p, \mathcal{M}_{\text{Lut}}, r$ )

---

**Entrée/Sortie** :  $\mathcal{M}_{\text{Lut}}$  voisinage de test  
**Entrée** :  $r$  rayon de la boule inscrite  
**Entrée** :  $p$  centre de la boule à tester  
**si**  $\forall \vec{v} \in \mathcal{M}_{\text{Lut}} : D_{\leq}(O + \vec{v}, \mathcal{C}_{D_{\leq}(O,r)}(O + \vec{v})) \not\subset D_{\leq}(p, \mathcal{C}_{D_{\leq}(O,r)}(p))$  **alors**  
    //  $p$  n'est pas couvert par un vecteur de  $\mathcal{M}_{\text{Lut}}$   
     $\mathcal{M}_{\text{Lut}} \leftarrow \mathcal{M}_{\text{Lut}} \cup \{(\vec{Op}, d(0,p))\}$   
**fin**

---

différents rayons. Ces résultats indiquent une complexité linéaire avec le rayon maximal pour la méthode proposée ici, alors qu'elle est en  $O(r^4)$  en dimension 3 pour l'algorithme de référence.

L	3, 4, 5		3, 4, 5, 7		7, 8, 11, 14		4, 6, 7, 9, 10	
	ref.	$\mathcal{H}$	ref.	$\mathcal{H}$	ref.	$\mathcal{H}$	ref.	$\mathcal{H}$
10	0.0006	0.0003	0.0002	0.0004	0.0013	0.0010	0.0004	0.0008
20	0.0049	0.0002	0.0032	0.0004	0.0135	0.0011	0.0047	0.0010
50	0.1079	0.0003	0.1096	0.0005	0.2886	0.0027	0.1592	0.0015
100	1.6316	0.0005	1.6831	0.0008	4.4704	0.0054	2.4626	0.0023
200	30.391	0.0010	31.126	0.0012	81.627	0.0109	53.126	0.0038
500	3523	0.0022	3537	0.0024	9333	0.0272	6465	0.0088
1000		0.0042		0.0073		0.0549		0.0220

Tableau 4.26 – Temps de calcul (en secondes) des méthodes de référence (ref.) et proposées ( $\mathcal{H}$ ) pour quatre masques 3D et différentes dimensions  $L \times L \times L$  (paramètre utilisé dans l'algorithme de référence).

## 4.7 Conclusion

Ce chapitre a présenté une méthode de description des boules de normes de chanfrein sous la forme de  $\mathcal{H}$ -polytopes. La famille des boules d'une distance est définie par une matrice unique qui représente les directions des facettes, c'est-à-dire de leurs hyperplans d'appui, et pour chacune par un vecteur de paramètres qui représente la position de ces hyperplans. Le coût spatial de la représentation est donc fixe et indépendant du rayon. Dans ce formalisme, les opérations de translation et le test d'inclusion se font à coût constant, ce qui le rend particulièrement adapté

au calcul des tables d'inclusion utilisées, par exemple, pour l'extraction de l'axe médian. La caractérisation du voisinage de test, qui identifie l'ensemble minimal de vecteurs permettant de décrire les relations d'inclusion directes bénéficie de la  $\mathcal{H}$ -représentation des boules. Par une partition adaptée de l'espace discret obtenue par triangulation, le domaine de recherche est considérablement réduit et des gains très significatifs en temps de traitement sont obtenus en évitant le recours au calcul systématique de la transformation en distance pour chaque rayon testé.

L'intérêt de cette représentation dépasse le cas de l'axe médian en fournissant une détermination de la condition de norme et, lorsque cette condition est vérifiée, une formulation directe de la distance (4.11) entre deux points quelconques sans qu'il soit nécessaire de procéder à l'identification d'un cône d'influence.

# Chapitre 5

## Granulométrie

### 5.1 Introduction

Ce chapitre est dédié à l'analyse d'images granulaires ou poreuses. L'objectif est de mesurer les dimensions des objets dans une image binaire. Nous nous intéresserons en particulier à l'obtention d'une distribution de leurs tailles, c'est-à-dire au calcul du spectre granulométrique de l'image. Les méthodes disponibles sont basées sur des traitements morphologiques, avec des opérations qu'il faut répéter pour chaque taille d'élément, sur un calcul de la transformation en ouverture décrite dans ce chapitre ou sur un calcul direct pour des formes d'éléments simples (segments, carrés).

Nous proposons une nouvelle méthode qui calcule une estimation de bonne qualité de la granulométrie à partir de la transformation en distance de l'image par l'examen des configurations locales de pixels. Les algorithmes sont présentés pour les distances à séquences de voisinage du plan pour lesquelles nous avons décrit un algorithme de transformation en distance. La méthode s'accommode de la version décentrée de la distance, ce qui la rend particulièrement efficace.

La section 5.2 rappelle des définitions et résultats de morphologie mathématique, de géométrie convexe et de géométrie intégrale qui seront utiles pour la suite du chapitre. La méthode est décrite dans la section 5.3 et ses résultats dans la section 5.4.

Une publication est en cours de préparation pour une communication lors de la conférence *Discrete Geometry for Computer Imagery 2013*.

### 5.2 Préliminaires

#### 5.2.1 Granulométrie

**Définition 5.1** (Granulométrie). Soit une famille de transformations  $\Phi = (\phi_r)_{r \leq 0}$  de paramètre  $r$ .  $\Phi$  est une granulométrie si et seulement si elle vérifie les propriétés suivantes [90, 92] :

$$\forall r, \phi_r \text{ est croissante} \quad (5.1)$$

$$\forall r, \phi_r \text{ est anti-extensive} \quad (5.2)$$

$$\forall r, \forall s, \phi_r \phi_s = \phi_s \phi_r = \phi_{\max\{r,s\}} \quad (5.3)$$

La propriété (5.3) implique que  $\phi_r$  est une transformation idempotente. Par conséquent,  $\Phi = (\phi_r)_{r \leq 0}$  est une famille décroissante d'ouvertures algébriques [151, 184]. Inversement, si  $B$

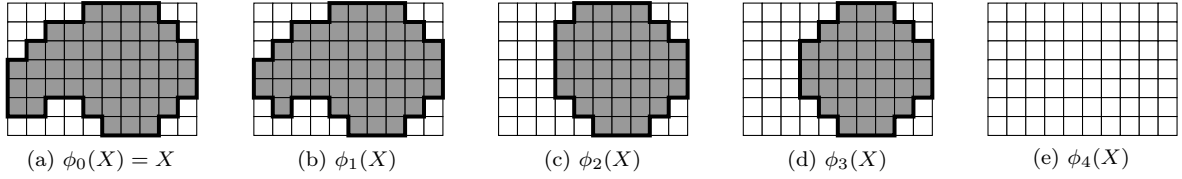


FIGURE 5.2 – Ouvertures morphologiques par une séquence de disques croissants.

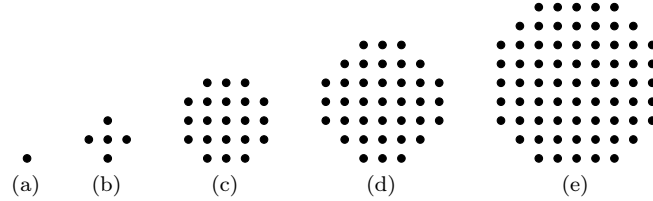


FIGURE 5.3 – Séquence des disques croissants de la distance octogonale utilisés comme éléments structurants pour l’ouverture morphologique. Les centres des disques ne sont pas représentés car ils n’ont pas d’influence sur le résultat de l’ouverture.

est un ensemble convexe, la famille des ouvertures morphologiques de paramètres  $rB$  est une granulométrie selon la définition précédente [92].

Une granulométrie agit sur l’image comme un tamis, le paramètre  $r$  spécifiant la taille des détails de l’image qui sont supprimés. Dans la suite, nous considérerons uniquement l’application d’ouvertures morphologiques à des images binaires.

**Définition 5.4** (Spectre granulométrique). Soit  $\mu$  une mesure bornée sur  $E$ , le spectre granulométrique de  $\mu$  sur l’ensemble  $X \in E$ , est défini par :

$$f_X(r) = \mu(\phi_{r-1}(X)) - \mu(\phi_r(X)) . \quad (5.4)$$

On définit également  $F_X$ , la somme cumulée de  $f_X$  :

$$F_X(r) = F_X(r-1) + f_X(r) = \mu(\phi_0(X) - \mu(\phi_r(X))) . \quad (5.5)$$

$\mu(\phi_r(X))$  peut être interprétée comme la quantité de matière qui ne passe pas dans un tamis de paramètre  $r$ . Si  $\mu$  mesure le nombre de points objet dans une image et  $\phi_r$  est une ouverture morphologique avec le disque de rayon  $r$ , alors  $\mu(\phi_0(X)) - \mu(\phi_r(X))$  représente le nombre de pixels de l’image initiale disparus par l’application de cette ouverture, et  $f_X(r)$  représente le nombre de pixels encore présents dans  $\phi_{r-1}(X)$  et absents de  $\phi_r(X)$ , autrement dit, le nombre de pixels qui traversent le tamis quand sa taille passe de  $r-1$  à  $r$ . Le tableau 5.5 et la figure 5.6 présentent le spectre granulométrique correspondant à la série d’ouvertures morphologiques de la figure 5.2.

**Définition 5.7** (Fonction d’ouverture). Soit  $d$  une fonction distance, ou pseudo-distance. La fonction d’ouverture, ou transformation en ouverture,  $OT_X$  de l’ensemble  $X$  est une fonction qui associe à chaque point  $p$  le rayon du plus grand disque ouvert de  $d$  inclus dans  $X$  contenant  $p$  :

$$\begin{aligned} OT_X : \mathbb{Z}^n &\rightarrow \mathbb{N} \\ OT_X(p) &= \max \{ r : \exists q, p \in \check{D}_<(q, r) \subset X \} . \end{aligned} \quad (5.6)$$

Tableau 5.5

$r$	0	1	2	3	4
$\mu(\phi_r(X))$	50	49	39	37	0
$f_X(r)$	0	1	10	2	37
$F_X(r)$	0	1	11	13	50

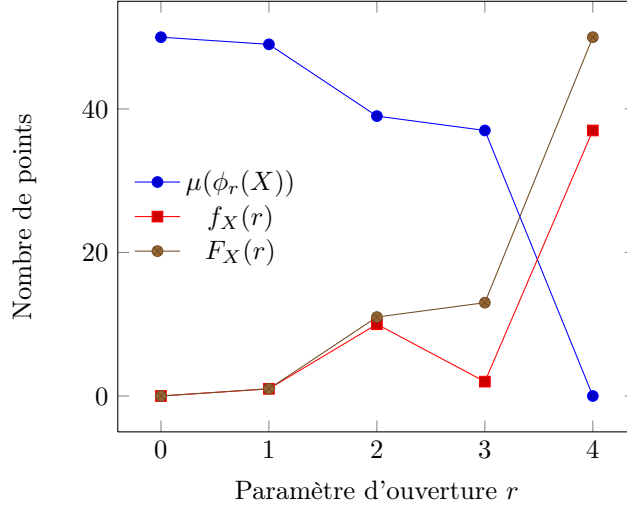


FIGURE 5.6

De manière équivalente,  $OT_X(p)$  est le plus petit paramètre  $r$  pour lequel le point  $p$  disparaît de l'ouverture morphologique de l'image :

$$OT_X(p) = \min \{r : p \notin \phi_r(X)\}.$$

Cette définition est très similaire à la définition de la transformation en distance (3.2). Cependant, la seconde ne considère que les disques centrés sur le point courant  $p$ . L'information liée à un disque particulier est donc localisée en son centre pour la transformation en distance alors qu'elle est répartie sur tous ses points pour la transformation en ouverture.

### 5.2.2 Calcul du spectre granulométrique

**Calcul direct** Le cas du spectre granulométrique en dimension 1 est particulièrement simple à traiter avec la série d'éléments structurants présentés sur la figure 5.10. Dans l'image  $X$ , la longueur d'une plage de points consécutifs est précisément le plus grand paramètre  $r$  de  $\phi_r(X)$  qui conserve cette plage. Une plage de longueur  $r$  contribue donc pour une quantité  $r$  à  $f_X(r)$ .

Les calculs sont détaillés dans l'algorithme 5.9. Cet algorithme peut être appliqué à des éléments structurants mono-dimensionnels dans des images de dimension supérieure [184].

**Séries d'ouvertures morphologiques** L'algorithme le plus général consiste à calculer les ouvertures morphologiques sur l'image  $X$ , indépendamment pour chaque valeur de  $r$ . Cependant, le coût algorithmique est important. Si la famille d'éléments structurants est une famille de disques, la transformation en distance  $DT_X$  correspondante synthétise les érosions de  $X$  avec

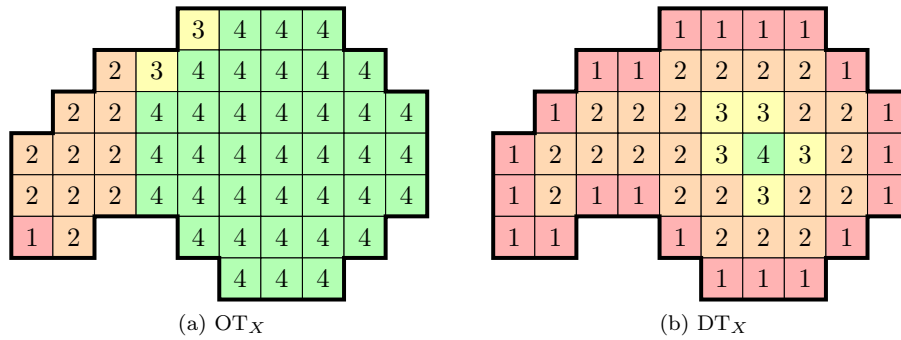


FIGURE 5.8 – Exemple de fonction d’ouverture  $OT_X$  et de transformation en distance  $DT_X$  d’une image binaire  $X$ . La distance utilisée pour  $DT_X$  est la distance octogonale, la famille d’éléments structurants utilisée pour  $OT_X$  est la famille des disques de la distance octogonale.

---

**Algorithme 5.9:** Calcul direct du spectre granulométrique en dimension 1 [184]

---

**Entrée :** image binaire  $X$  de taille  $L$

**Sortie :** spectre granulométrique  $f_X$

$\forall r, f_X(r) \leftarrow 0$

$r \leftarrow 0$

**début**

**pour**  $l \leftarrow 0$  à  $L - 1$  **faire**

$p \leftarrow (l)$

**si**  $p \in X$  **alors**

$r \leftarrow r + 1$

**sinon**

$f_X(r) \leftarrow f_X(r) + r$

$r \leftarrow 0$

**fin**

**fin**

**fin**

---



FIGURE 5.10 – Séquence des disques croissants horizontaux.



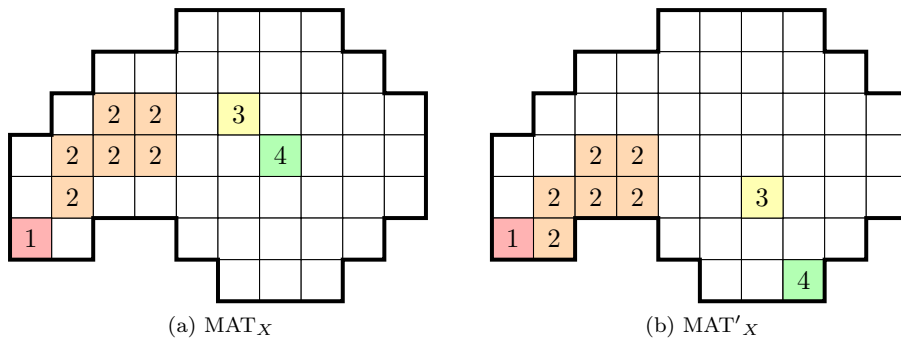


FIGURE 5.11 – Exemple d’axe médian pondéré  $MAT_X$  et d’axe médian translaté  $MAT'_X$  d’une image binaire  $X$ . La distance utilisée est la distance octogonale translatée ou non.

toutes les valeurs de rayon. On peut donc factoriser cette étape mais il reste autant de dilatations à effectuer que de valeurs possibles de  $r$ . Le coût ne peut donc être indépendant du contenu de l’image, comme c’est le cas pour la transformation en distance.

**Fonction d’ouverture** Le spectre granulométrique se déduit immédiatement de la fonction d’ouverture dont il est l’histogramme (pour des valeurs positives de  $r$ ) :

$$\forall r > 0, f_X(r) = \text{Card}(\{p : OT_X(p) = r\}) \quad (5.7)$$

Bien que les définitions des transformations en distance  $DT_X$  (3.2) et en ouverture  $OT_X$  (5.6) soient similaires, il n’existe pas d’algorithme en temps linéaire pour calculer la seconde. En effet, l’information du rayon de plus grand disque est localisée en son centre dans  $DT_X$  alors qu’elle est diffusée sur les pixels de sa surface dans  $OT_X$ .

Conceptuellement, il suffit pour chaque disque  $\check{D}_<(p, r)$  dont le rayon est donné par la transformation en distance  $r = DT_X(p)$ , d’associer la valeur de son rayon,  $r$ , à chacun de ses pixels. Bien entendu, de nombreux disques se recouvrent et le maximum de leurs rayons doit être retenu. Les recouvrements entraînent un coût de calcul important et dépendant du contenu de l’image.

Pour limiter le nombre de recouvrements, on peut observer que lorsqu’un disque n’est pas maximal, il n’est pas nécessaire de le remplir, car il est, par définition, entièrement inclus dans un disque de plus grand rayon. On peut donc se contenter de colorier les disques maximaux détectés dans l’axe médian de l’image [190]. Les applications pratiques de cet algorithme sont limitées aux cas où l’axe médian est simple à extraire. C’est le cas pour une distance simple, une distance à séquence de voisinages, et si le masque est de taille limitée, pour une distance de chanfrein. En revanche, dans le cas de la distance euclidienne, l’obtention de l’axe médian est plus complexe ou approchée [28, 79].

Notons que la détection des disques peut se faire dans un axe médian translaté. La figure 5.11 illustre les axes médians d’une forme simple pour la distance octogonale et la distance octogonale translatée. Le second peut être extrait avec un coût moindre (un seul balayage d’image) ce qui réduit la complexité globale du calcul de la fonction d’ouverture. Cependant, le gain est marginal car il intervient dans une phase peu coûteuse du calcul.

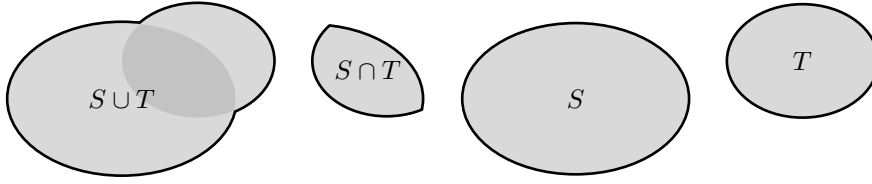


FIGURE 5.13 – Additivité des valuations

### 5.2.3 Théorie de BRUNN-MINKOWSKI

**Définition 5.12** (Valuation). Une fonction  $\mu : \mathbb{R}^n \rightarrow \mathbb{R}$  est une *valuation* si elle a la propriété d'*additivité*, c'est-à-dire, si pour tous les ensembles  $S, T \subset \mathbb{R}^n$  :

$$\mu(S \cup T) + \mu(S \cap T) = \mu(S) + \mu(T). \quad (5.8)$$

Une valuation est dite *invariante par translation* si pour tout vecteur  $t \in \mathbb{R}^n$ ,  $\mu(K + t) = \mu(K)$  et *invariante par translation entière* si c'est le cas pour tout vecteur de  $\mathbb{Z}^n$ . Les mesures de l'aire, du périmètre, de la caractéristique d'EULER-POINCARÉ sont des valuations invariantes par translation. La mesure du nombre de points discrets (cardinal de l'intersection de  $S$  avec  $\mathbb{Z}^n$ ) est une valuation invariante par translation entière.

On déduit de l'additivité, le principe d'inclusion-exclusion :

$$\mu(S_1 \cup \dots \cup S_n) = \sum_{i=1}^n (-1)^{i-1} \sum_{1 \leq j_1 < \dots < j_i \leq n} \mu(S_{j_1} \cap \dots \cap S_{j_i}) \quad (5.9)$$

#### Formule de STEINER

**Définition 5.14** (somme de MINKOWSKI). Soient  $A$  et  $B$  deux sous-ensembles de  $\mathbb{R}^n$ , la somme de MINKOWSKI de  $A$  et  $B$ , notée  $A \oplus B$  est l'ensemble :

$$A \oplus B = \{a + b : a \in A, b \in B\}. \quad (5.10)$$

Lorsque  $A$  et  $B$  sont convexes, leur somme de MINKOWSKI l'est également.

**Définition 5.15.** On note  $\lambda A$  l'homothétique de  $A$  de rapport  $\lambda \in \mathbb{R}$ , l'ensemble :

$$\lambda A = \{\lambda a : a \in A\}. \quad (5.11)$$

Soit  $\mathcal{K}$  un sous-ensemble convexe de  $\mathbb{E}^n$ . On appelle *tube* ou *r-offset* (G. BOULIGAND, G. DURAND parlent de *construction de CANTOR-MINKOWSKI* [24, 47]) de  $\mathcal{K}$  à une distance  $r$ , noté  $\mathcal{K}_r$ , le dilaté de  $\mathcal{K}$  par la boule euclidienne de rayon  $r$  :

$$\mathcal{K}_r = \mathcal{K} \oplus rB$$

où  $B$  est la boule euclidienne unitaire.

Le volume du tube de  $\mathcal{K}$  est donné par la formule de STEINER [164] :

$$\text{Vol}_n(\mathcal{K}_r) = \sum_{k=0}^n \Phi_{k,n}(\mathcal{K}) r^k, \forall r \geq 0 \quad (5.12)$$

où  $\Phi_{0,n}(\mathcal{K}) = \text{Vol}_n(\mathcal{K})$ .

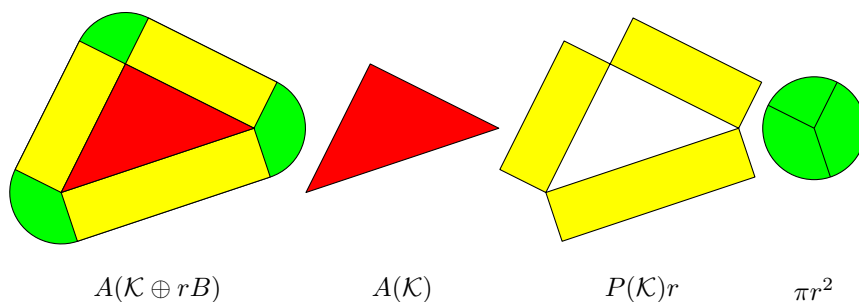


FIGURE 5.16 – Illustration des termes de la formule de STEINER dans le plan. L’aire d’un triangle dilaté par une boule euclidienne est la somme de l’aire du polygone initial,  $A(\mathcal{K})$ , l’aire de rectangles sur tout le périmètre de  $\mathcal{K}$ ,  $P(\mathcal{K})r$ , et l’aire de la boule,  $\pi r^2$ .

Les coefficients  $\Phi_{k,n}$  sont dénommés *quermassintégrales* ou *mixed volumes*. Avec une normalisation adéquate, ils deviennent *volumes intrinsèques*, indépendants de la dimension de l’espace ambiant [57, 77, 96].

En dimension 2,  $\Phi_{k,n}(\mathcal{K})$  peut être interprété comme le périmètre  $P(\mathcal{K})$  de  $\mathcal{K}$ . Le polynôme (5.12) devient :

$$A(\mathcal{K}_r) = A(\mathcal{K}) + P(\mathcal{K})r + \pi r^2 .$$

Par exemple, pour un disque euclidien de rayon  $R$  :

$$\pi(R + r)^2 = \pi R^2 + 2\pi Rr + \pi r^2 .$$

La figure 5.16 illustre les termes de la somme pour un polygone convexe dans le plan.

**Théorème 5.17** (Théorème de MINKOWSKI-STEINER). Soient  $K_1$  et  $K_2$  deux compacts convexes de  $\mathbb{R}^n$ . Pour tout  $\lambda_1 \geq 0$  et  $\lambda_2 \geq 0$ , le volume de  $\lambda_1 K_1 \oplus \lambda_2 K_2$  est un polynôme homogène de degré  $n$  en  $\lambda_1, \lambda_2$  :

$$\text{Vol}(\lambda_1 K_1 \oplus \lambda_2 K_2) = \sum_{i=0}^n \binom{n}{i} \text{Vol}(K_1^{[i]}, K_2^{[n-i]}) \lambda_1^i \lambda_2^{n-i} . \quad (5.13)$$

### Extension à des ensembles non convexes

**Définition 5.18** (Portée, H. FEDERER). La *portée* d’un sous-ensemble  $A$  d’un espace euclidien  $\mathbb{E}^n$  est le plus grand  $\epsilon$  (éventuellement  $\infty$ ) tel que si  $x \in \mathbb{E}^n$  et  $x$  est à une distance de  $A$  inférieure à  $\epsilon$  alors le point de  $A$  le plus proche de  $x$ , noté  $\xi_A(x)$ , est unique [53].

Pour un ensemble de portée  $\epsilon$ , la formule de STEINER (5.12) s’applique avec la restriction  $r \leq \epsilon$  [53]. Un exemple d’ensemble de portée positive est donné sur la figure 5.19.

G. DURAND avait dès 1931 proposé une définition équivalente à celle de H. FEDERER : par tout point de l’ensemble on peut faire passer une sphère de rayon  $\epsilon$  n’enfermant intérieurement aucun point de cet ensemble [47]. G. MATHERON a, de son côté, étendu la formule de STEINER aux érosions (autrement dit, avec un rayon  $r$  négatif) pour les ensembles qui sont des ouverts, ce qui revient à dire que l’ouvert est un ensemble de portée positive [91].

R. SCHNEIDER étend l’application de la formule de STEINER aux ensembles de portée nulle en donnant une définition différente du volume, qui prend en compte les multiplicités introduites [150].

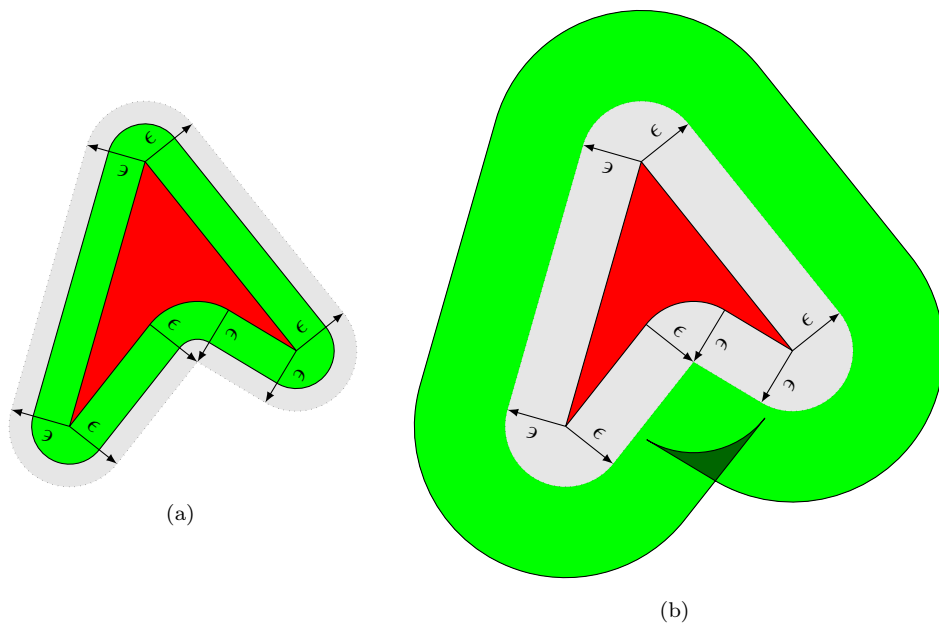


FIGURE 5.19 – Ensemble non convexe mais de portée  $\epsilon$  positive. Dilatation avec une boule euclidienne dont le rayon est inférieur à  $\epsilon$  à gauche, supérieur à droite.

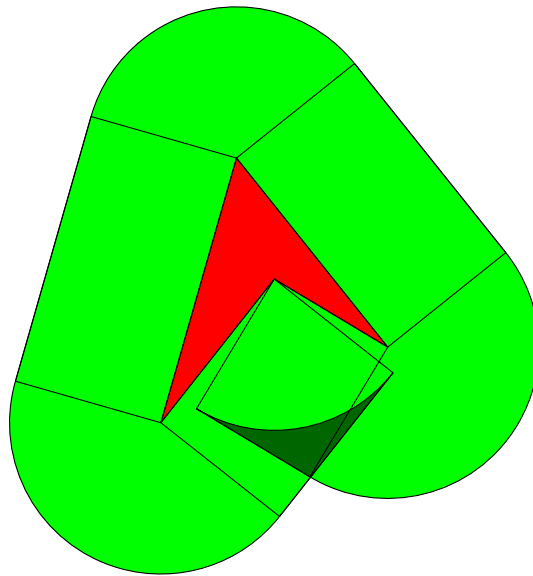


FIGURE 5.20 – Dilatation d'un ensemble au-delà de sa portée  $\epsilon$  (nulle ici). L'application de la formule de STEINER (5.12) induit des multiplicités dans le calcul de la surface du dilaté.

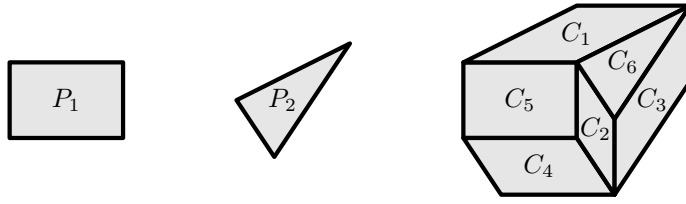


FIGURE 5.24 – Subdivision mixte (exemple tiré de [163]).

### Calcul des volumes mixtes

**Définition 5.21** (Cellule). On appelle *cellule* d'une famille  $\mathcal{P} = (P^{(1)}, \dots, P^{(m)})$  de polytopes de  $\mathbb{R}^n$ , une famille  $\mathcal{C} = (C^{(1)}, \dots, C^{(m)})$  de sous-ensembles  $C^{(i)} \subset P^{(i)}$ .

**Définition 5.22** (Subdivision). Une *subdivision* de  $\mathcal{P} = (P^{(1)}, \dots, P^{(m)})$  avec  $P^{(1)} \oplus \dots \oplus P^{(m)} = P$  est une collection  $(C_1, \dots, C_k)$  de cellules de  $\mathcal{P}$  telle que :

- (i)  $\dim(P_i) = n$ ,
- (ii)  $P_i \cap P_j$  est une face de  $P_i$  et de  $P_j$ ,
- (iii)  $\bigcup_{i=1}^k P_i = P$ ,

Une subdivision est dite *mixte* lorsque :

$$\dim C_1^{(i)} + \dots + \dim C_k^{(i)} = n$$

**Exemple 5.23.** L'exemple suivant est dû à R. J. STEFFENS [163]. Soit la famille  $\mathcal{P} = (\text{conv}\{(0, 0), (3, 0), (0, 2), (3, 2)\}, \text{conv}\{(0, 2), (3, 2)\})$ ,  $\Gamma = (C_1, \dots, C_6)$  est une subdivision mixte de  $\mathcal{P}$  avec :

$$\begin{aligned} C_1 &= (\text{conv}\{(0, 2), (3, 2)\}, \text{conv}\{(0, \frac{3}{2}), (3, 3)\}), \\ C_2 &= (\text{conv}\{(3, 0), (3, 2)\}, \text{conv}\{(0, \frac{3}{2}), (1, 0)\}), \\ C_3 &= (\text{conv}\{(3, 0), (3, 2)\}, \text{conv}\{(1, 0), (3, 3)\}), \\ C_4 &= (\text{conv}\{(0, 0), (3, 0)\}, \text{conv}\{(1, 0), (0, \frac{3}{2})\}), \\ C_5 &= (\text{conv}\{(0, 0), (3, 0), (0, 2), (3, 2)\}, \{(0, \frac{3}{2})\}), \\ C_6 &= (\{(3, 2)\}, \text{conv}\{(1, 0), (0, 3), (3, 3)\}). \end{aligned}$$

Les cellules sont respectivement de type (1, 1) pour  $C_1$  à  $C_4$ , de type (2, 0) pour  $C_5$  et (0, 2) pour  $C_6$ .

Une subdivision mixte d'une famille de polytopes fournit un moyen de calculer les volumes mixtes nécessaires à l'application du théorème de MINKOWSKI (5.13).

**Somme de MINKOWSKI d'un polytope et d'un segment** Soit  $P_1$  un polytope de  $\mathbb{R}^n$ , et  $P_2 = \text{conv}(-p/2, p/2)$ , un segment. Soit  $N_{d-1}(P) = \{\vec{u} \in S^{d-1} : \dim(H(P, \vec{u}) \cap P) = d - 1\}$  l'ensemble des vecteurs normaux externes unitaires des facettes de  $P$ . On note  $\mathcal{F}_+$ , respectivement  $\mathcal{F}_-$ , l'ensemble des facettes de  $P_1$  pour lesquelles le point  $p$  est situé strictement à l'extérieur, respectivement à l'intérieur :

$$\begin{aligned} \mathcal{F}_+ &= \{H(P_1, \vec{u}) \cap P_1 : \vec{u} \cdot p > 0, \vec{u} \in N_{d-1}(P_1)\} \\ \mathcal{F}_- &= \{H(P_1, \vec{u}) \cap P_1 : \vec{u} \cdot p < 0, \vec{u} \in N_{d-1}(P_1)\} \end{aligned}$$

alors

$$\Gamma = ((P_1, O), \{(F, \text{conv}(O, p/2)), F \in \mathcal{F}_+\}, \{(F, \text{conv}(O, -p/2)), F \in \mathcal{F}_-\})$$

est une subdivision mixte de  $(P_1, P_2)$ .

On en déduit que le volume de  $P = P_1 \oplus P_2$  est :

$$\text{Vol}_n(P_1 \oplus P_2) = \text{Vol}_n(P_1) + \sum_{\vec{u} \in N_{d-1}(P_1)} |\vec{u} \cdot p| \text{Vol}_{n-1}(H(P_1, \vec{u}) \cap P_1) . \quad (5.14)$$

### 5.3 Estimation du spectre granulométrique à partir de la transformation en distance

L'additivité des valuations et le principe d'inclusion-exclusion qui en découle ont une conséquence pratique capitale : une valuation sur une image peut être calculée comme la somme des valuations de configurations locales [59, 151]. R. GUDERLEI, S. KLENK, J. MAYER, V. SCHMIDT et E. SPODAREV, S. KLENK, V. SCHMIDT et E. SPODAREV proposent une méthode d'estimation des volumes intrinsèques sur des réalisations de modèles aléatoires [59, 85]. Bien que très similaire à la méthode présentée ici, elle diffère notamment par son objectif qui est de donner une estimation précise de volumes intrinsèques et non d'obtenir une granulométrie.

La méthode proposée dans cette section consiste à estimer le spectre granulométrique à partir de la transformation en distance sans recourir explicitement à la transformation en ouverture, plus coûteuse à calculer. Pour un paramètre  $r$  donné, l'érodé de l'image par le disque de rayon  $r$ ,  $\epsilon_r(X)$ , est obtenu par seuillage de la carte de distance  $\text{DT}_X$ . L'ouvert  $\phi_r(X)$ , c'est-à-dire le dilaté de  $\epsilon_r(X)$ , n'est pas calculé mais ses caractéristiques sont estimées en extrapolant celles de  $\epsilon_r(X)$ . Nous utiliserons une version de la formule de STEINER [164] adaptée à notre problème (dilatation par un segment et non une boule euclidienne) et les propriétés de la somme de MINKOWSKI sur les polytopes.

#### 5.3.1 Modèle continu

Le modèle continu adopté est celui d'un polyconvexe dont les contours 8-connexes joignent les points discrets de la forme (voir figure 5.25c). Chaque contour peut être codé par une chaîne de H. FREEMAN, c'est-à-dire par une séquence d'arêtes élémentaires dont les sommets sont deux points 4- ou 8-voisins [55]. Un contour est soit intérieur, soit extérieur et la caractéristique d'EULER-POINCARÉ est la différence du nombre de contours extérieurs et du nombre de contours intérieurs. Par convention, FREEMAN numérote les arêtes selon leur direction, dans le sens trigonométrique. Dans un réseau de points carré, un vecteur de code  $i$  est donc orienté dans la direction  $i\pi/4$ . Le contour extérieur de la forme présentée sur la figure 5.25c peut être codé selon la séquence 171731215323565375075, en partant du point en bas à gauche et en tournant dans le sens trigonométrique. Le contour d'un polygone convexe admet une forme réduite unique  $0^{\nu_0}1^{\nu_1} \dots 7^{\nu_7}$  où  $\nu_d$  est le nombre (éventuellement nul) d'arêtes dans la direction  $d$ . Le polygone de la figure 5.25a, par exemple, a pour chaîne réduite  $0^11^22^23^24^05^36^17^2$ .

La somme de MINKOWSKI d'un polygone convexe  $P_1$  représenté par la chaîne  $0^{\nu_0}1^{\nu_1} \dots 7^{\nu_7}$  avec un segment  $P_2$  de direction  $d$  est  $P_1 \oplus P_2$  et dont chaîne de FREEMAN réduite est caractérisée par [e.g., 125] :

$$\nu_\delta = \begin{cases} \nu_1 + 1 & \text{si } \delta \equiv d \pmod{4} \\ \nu_1 & \text{sinon} \end{cases} . \quad (5.15)$$

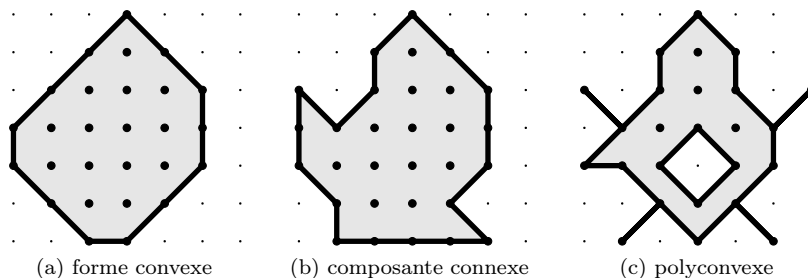


FIGURE 5.25

La condition de convexité peut être remplacée par une condition de *portée directionnelle* : si pour toute droite  $D$  de direction  $\vec{v}$ , l'intersection de  $P_1$  avec  $D$  est un ensemble dont la portée  $\epsilon$  est au moins égale à la moitié de la longueur du segment. Autrement dit, si  $x$  est à une distance de  $A \cap \{x + \lambda \vec{v}, \lambda \in \mathbb{R}\}$  inférieure à  $\|\vec{v}\|/2$  alors le point de  $A \cap \{x + \lambda \vec{v}, \lambda \in \mathbb{R}\}$  le plus proche de  $x$  est unique. Ceci peut encore être exprimé par le fait que le contour est un érodé (par le segment), argument similaire à celui de G. MATHERON [91]. (5.15) reste alors valide si  $\nu_d$  est le décompte des segments de direction  $d$ , indépendamment de la séquence des arêtes dans la chaîne de FREEMAN. De manière similaire, si  $0^{\nu_0} 1^{\nu_1} \dots 7^{\nu_7}$  représente un contour intérieur, celui-ci devient :

$$\nu_\delta = \begin{cases} \nu_1 - 1 & \text{si } \delta \equiv d \pmod{4} \\ \nu_1 & \text{sinon} \end{cases} .$$

Pour un polyconvexe  $P_1$ , dont la caractéristique d'EULER-POINCARÉ est  $\chi$  et de portée directionnelle  $\epsilon$  supérieure ou égale à  $1/2$  (directions 0, 2, 4 ou 6) ou  $\sqrt{2}/2$  (directions 1, 3, 5 ou 7) :

$$\nu_\delta = \begin{cases} \nu_1 - \chi & \text{si } \delta \equiv d \pmod{4} \\ \nu_1 & \text{sinon} \end{cases} . \quad (5.16)$$

Il apparaît donc (d'après 5.14 et 5.16) que la connaissance des chaînes de FREEMAN, autrement dit, de la séquence ordonnée des arêtes, n'est pas nécessaire à l'extrapolation des paramètres d'un dilaté par un segment. Seul est requis le décompte de chaque type d'arête. De plus, les mesures des arêtes de directions congruentes modulo 4 peuvent être confondues.

### Configurations locales

Les mesures qui nous intéressent sont :  $\sharp$  le nombre de points discrets,  $\nu_0, \nu_1, \nu_2, \nu_3$ , le nombre d'arêtes élémentaires orientées respectivement à  $0^\circ, 45^\circ, 90^\circ$  et  $135^\circ$  ainsi que la caractéristique d'EULER-POINCARÉ,  $\chi$ .

Le modèle continu associé à un ensemble de points de la grille discrète peut être déterminé à partir des configurations locales de  $2 \times 2$  points discrets. Le calcul d'une valuation, quelle qu'elle soit, ne nécessite donc que l'examen de ces configurations locales et de leurs intersections, selon (5.8) et (5.9).

La caractéristique d'EULER-POINCARÉ du polyconvexe présenté figure 5.25c est nulle (une composante et un trou), ce qui correspond au contributions de ses configurations locales (en l'occurrence, 73 contributions positives, autant de contributions négatives, voir la vue éclatée figure 5.26). Puisque chaque cellule  $2 \times 2$  contient l'information des intersections avec les cellules voisines, les contributions des intersections peuvent être intégrées aux cellules  $2 \times 2$ , ce qui évite

un comptage séparé des cellules  $2 \times 1$ ,  $1 \times 2$  et  $1 \times 1$  qui forment les intersections. Le choix fait ici est de répartir les poids des intersections équitablement entre toutes les cellules qui les contiennent. Aucune des cellules  $2 \times 2$ ,  $2 \times 1$ ,  $1 \times 2$  ou  $1 \times 1$  ne pouvant comporter de trou, leur caractéristique d'EULER-POINCARÉ est 0 (cellule vide) ou 1 (cellule non vide). Les cellules  $2 \times 1$  et  $1 \times 2$  distribuent leur contribution aux deux cellules  $2 \times 2$  voisines, les cellules  $1 \times 1$ , aux quatre cellules voisines. Le procédé est identique pour le comptage des arêtes. Il faut bien sûr tenir compte du fait qu'un segment compte pour deux arêtes, une par face.

Par convention, la cellule  $\{(k, l), (k + 1, l), (k, l + 1), (k + 1, l + 1)\}$  est numérotée de manière unique  $2^0 f(k, l) + 2^1 f(k + 1, l) + 2^2 f(k, l + 1) + 2^3 f(k + 1, l + 1)$  pour une image  $f$  binaire. Le tableau 5.28 illustre les 16 configurations locales possibles et leurs contributions aux différentes valuations. Le nombre de configurations distinctes aurait pu être réduit à 6 en prenant en compte leurs symétries, mais sans bénéfice calculatoire.

Ces paramètres sont redondants, l'aire  $A$  vérifie, ligne par ligne, le théorème de PICK [140] :

$$A = \# - \frac{1}{2}(\nu_1 + \nu_2 + \nu_3 + \nu_4) - \chi . \quad (5.17)$$

Le comptage des points discrets est particulier car il ne nécessite pas de recours au modèle continu de la forme. Deux méthodes de calcul sont proposées, celle qui vient d'être décrite (colonne  $\#$ ) et la méthode la plus simple et naturelle qui consiste à comptabiliser isolément chaque point de la forme (colonne  $\#'$ ). Le principe d'inclusion-exclusion est toujours respecté, les points étant distincts deux à deux.

### 5.3.2 Extraction simultanée des mesures pour tous les rayons

En tirant profit de la propriété de l'*additivité*, il est possible de mesurer, sur une image, des valuations invariantes par translation en dénombrant les configurations locales des points de l'image et en appliquant, pendant ou à l'issue du comptage, les pondérations adéquates à chaque configuration. Pour une image  $f$  à valeurs dans un intervalle de  $\mathbb{N}$ , les différents choix de seuil  $f(\cdot, \cdot) \geq r$  sur une cellule  $2 \times 2$   $v = ((k, l), (k + 1, l), (k, l + 1), (k + 1, l + 1))$  entraînent au plus 5 configurations binaires locales selon la façon dont  $r$  se compare aux valeurs des 4 pixels. Si  $\sigma$  est la permutation réciproque du rang des éléments de  $v$  par ordre croissant, *i.e.*,  $v(\sigma(i)) \leq v(\sigma(i + 1))$ ,  $\forall i \in [0, 2]$ , un seuil  $r$  inférieur à  $v(\sigma(0))$  correspond à la configuration 0,  $r > \sigma(3)$  à la configuration 15, et en général,  $r \in [v(\sigma(i)), v(\sigma(i + 1))]$  à  $\sum_{j=0}^i 2^{\sigma(j)}$ .  $\sigma$  étant déterminé par un algorithme de tri classique, le décompte est obtenu par codage différentiel en associant à la configuration  $\sum_{j=0}^i 2^{\sigma(j)}$ , +1 au seuil  $v(\sigma(i))$ , -1 au seuil  $v(\sigma(i + 1))$  pour l'intervalle  $[v(\sigma(i)), v(\sigma(i + 1))]$ . L'algorithme 5.29 fonctionne sur ce principe : partant du seuil  $\theta = 0$  et de la configuration correspondante  $c = 15$ , il parcourt les valeurs de seuil qui provoquent des changements de configuration et met à jour la valeur de  $c$  en fonction de l'indice du point dont la valeur devient inférieure au seuil. Un algorithme basé sur le principe de codage différentiel d'intervalle a déjà été présenté par L. SNIDARO et G. L. FORESTI [161]. Cependant, il est limité à la mesure de la caractéristique d'EULER-POINCARÉ et sa structure moins générale le rend peu adaptable à d'autres valuations.

L'algorithme 5.29 termine par une phase d'intégration (ligne 13 et suivantes) après laquelle  $\boldsymbol{\mu}(r)$  est le vecteur des caractéristiques de l'image binarisée avec un seuil  $r$ . Notons que cet algorithme est directement généralisable à la troisième dimension et est compatible avec toute valuation sur le modèle continu 8-connexe.

Les résultats sont identiques à ceux présentés dans le tableau 5.5 et la figure 5.6 pour la transformation en ouverture de la figure 5.32. En particulier, les pixels dont la valeur est au



















Configuration locale	$2A$	$4\sharp$	$\sharp'$	$\nu_0$	$\nu_1$	$\nu_2$	$\nu_3$	$4\chi$
0000 	0	0	0	0	0	0	0	0
0001 	0	1	1	0	0	0	0	1
0010 	0	1	0	0	0	0	0	1
0011 	0	2	1	1	0	0	0	0
0100 	0	1	0	0	0	0	0	1
0101 	0	2	1	0	0	1	0	0
0110 	0	2	0	0	2	0	0	-2
0111 	1	3	1	0	1	0	0	-1
1000 	0	1	0	0	0	0	0	1
1001 	0	2	1	0	0	0	2	-2
1010 	0	2	0	0	0	1	0	0
1011 	1	3	1	0	0	0	1	-1
1100 	0	2	0	1	0	0	0	0
1101 	1	3	1	0	0	0	1	-1
1110 	1	3	0	0	1	0	0	-1
1111 	2	4	1	0	0	0	0	0

Tableau 5.28 – Contribution des 16 configurations locales à différentes mesures.  $\sharp$  et  $\sharp'$  sont deux manières équivalentes (au sens de leur somme sur le domaine d'une image) de compter les points.

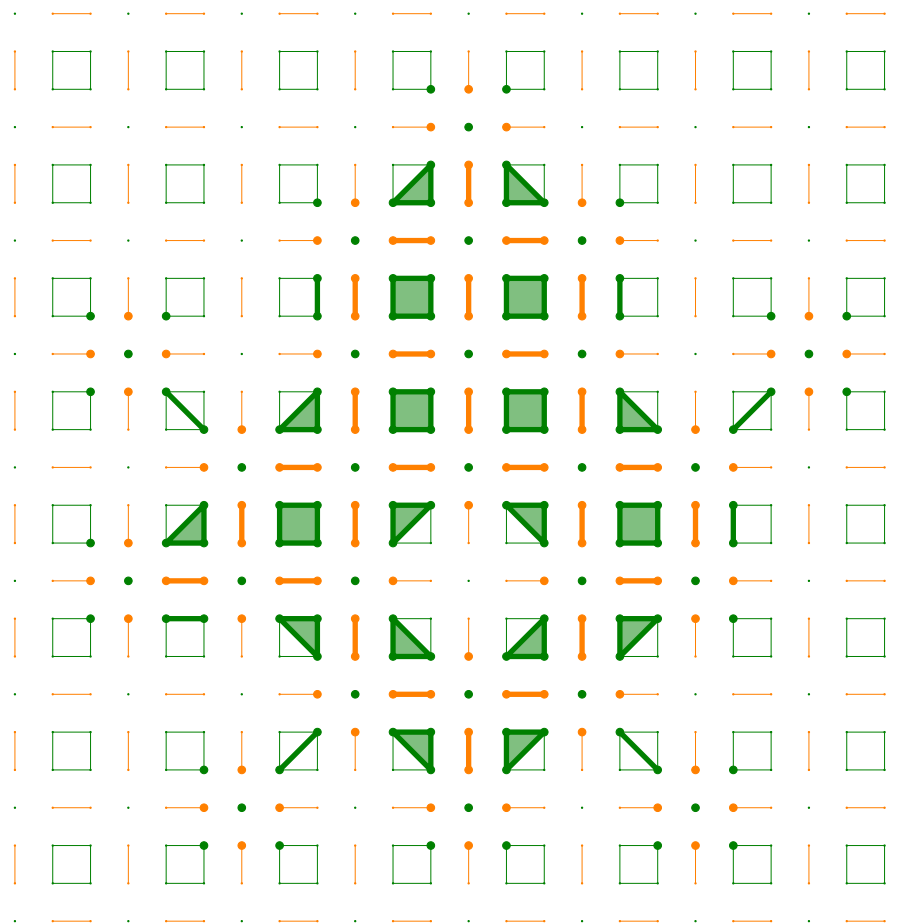


FIGURE 5.26 – Vue éclatée du polyconvexe de la figure 5.25c. Chacune des 48 cellules  $2 \times 2$  non vides contribue pour une unité à la caractéristique d'EULER-POINCARÉ, 25 cellules  $1 \times 1$  non-vides qui correspondent aux points de la forme initiale contribuent positivement. Ceci est compensé par 73 cellules  $1 \times 2$  ou  $2 \times 1$  non vides qui contribuent de manière négative. La caractéristique d'Euler-Poincaré de la forme est donc nulle.

moins 3 forment une composante connexe unique sans trou qui contient 39 points, dont le contour possède 10 arêtes horizontales ou verticales et 7 arêtes diagonales.

### 5.3.3 Extrapolation des mesures à un dilaté

La somme de MINKOWSKI d'un ensemble avec un segment dans une direction de FREEMAN (sous condition de portée directionnelle suffisante) est une application linéaire dans l'espace des mesures, d'après 5.14 et 5.16, qu'on peut donc exprimer de manière matricielle. Les 4 matrices de



FIGURE 5.27 – Contribution d'une cellule locale  $2 \times 2$  à différentes valuations en intégrant les contributions des intersections avec les cellules voisines.

À gauche. Pour la caractéristique d'EULER-POINCARÉ la cellule contribue en elle-même pour 1. Les contributions des quatre cellules  $1 \times 2$  et  $2 \times 1$  non vides voisines sont  $-1$  pour chacune, partagées sur deux cellules  $2 \times 2$ . Trois des quatre cellules  $1 \times 1$  voisines ne sont pas vides. La contribution de chacune vaut 1, répartie sur quatre cellules  $2 \times 2$ . La contribution globale de la cellule centrale est donc égale à  $\chi = 1 - 4 \times \frac{1}{2} + 3 \times \frac{1}{4} = -\frac{1}{4}$ .

Pour la mesure  $\nu_2$  du nombre d'arêtes verticales, la cellule  $2 \times 2$  en possède 1, mais une cellule voisine  $1 \times 2$  en possède 2 (contour fermé). Le compte total est donc nul.

À droite.

$$\chi = 1 - 3 \times \frac{1}{2} + 2 \times \frac{1}{4} = 0$$

$$\nu_2 = 2 - 2 \times \frac{1}{2} = 1$$

---

**Algorithme 5.29:** Extraction simultanée des mesures pour tous les seuils de binarisation d'une image.

---

**Entrée :** image  $f$  de taille  $K \times L$  ( $f(k, l) = 0, \forall (k, l) \notin [0, K - 1] \times [0, L - 1]$ )

```

1 début
2   pour  $l \leftarrow 0$  à  $L$  faire
3     pour  $k \leftarrow 0$  à  $K$  faire
4        $v \leftarrow (f(k - 1, l - 1), f(k, l - 1), f(k - 1, l), f(k, l))$ 
5       déterminer  $\sigma$ , permutation réciproque du rang des éléments de  $v$  par ordre
        croissant, i.e.,  $v(\sigma(i)) \leq v(\sigma(i + 1)), \forall i \in [0, 2]$ 
6        $c \leftarrow 15$ 
7        $\tau \leftarrow 0$ 
8       pour  $t \leftarrow 0$  à 3 faire
9          $\mu(\tau) \leftarrow \mu(\tau) + \nu(c)$ 
10         $\tau \leftarrow v(\sigma(t)) + 1$ 
11         $\mu(\tau) \leftarrow \mu(\tau) - \nu(c)$ 
12         $c \leftarrow c - 2^{\sigma(t)}$ 
13      fin
14    fin
15  fin
  // Cumul
16  pour  $\tau = 0$  à  $\max(f)$  faire
17     $\mu(\tau + 1) \leftarrow \mu(\tau + 1) + \mu(\tau)$ 
18  fin
19 fin
```

---

$r$	#	$\nu_0 + \nu_2$	$\nu_1 + \nu_3$	$4\chi$	$r$	#	$\nu_0 + \nu_2$	$\nu_1 + \nu_3$	$4\chi$
0	88	0	0	0	0	88	0	0	0
1	50	12	10	1	1	-38	12	10	1
2	49	10	11	1	2	-1	-2	1	0
3	<b>39</b>	<b>10</b>	<b>7</b>	<b>1</b>	3	-10	0	-4	0
4	37	8	8	1	4	-2	-2	1	0
5	0	0	0	0	5	-37	-8	-8	-1

Tableau 5.30 – Paramètres mesurés sur la transformation en ouverture de la figure 5.32 sous leur forme absolue à gauche et différentielle à droite, pour chaque valeur de seuil  $r$ . Il y a, par exemple, sur la figure 5.32, 39 (*i.e.*,  $88 - 38 - 1 - 10$ ) pixels de valeur supérieure ou égale à 3.

$r$	#	$\nu_0 + \nu_2$	$\nu_1 + \nu_3$	$4\chi$	$r$	#	$\nu_0 + \nu_2$	$\nu_1 + \nu_3$	$4\chi$
0	88	0	0	0	0	88	0	0	0
1	50	12	10	1	1	-38	12	10	1
2	28	10	7	1	2	-22	-2	-3	0
3	<b>6</b>	<b>2</b>	<b>3</b>	<b>1</b>	3	-22	-8	-4	0
4	1	0	0	1	4	-5	-2	-3	0
5	0	0	0	0	5	-1	0	0	-1

Tableau 5.31 – Paramètres mesurés sur la transformation en distance de la figure 5.33 sous leur forme absolue à gauche et différentielle à droite, pour chaque valeur de seuil  $r$ . Il y a, par exemple, sur la figure 5.33, 6 (*i.e.*,  $88 - 38 - 22 - 22$ ) pixels de valeur supérieure ou égale à 3. Les résultats sont identiques sur la carte symétrique  $DT_X$  et la carte asymétrique  $DT'_X$ .

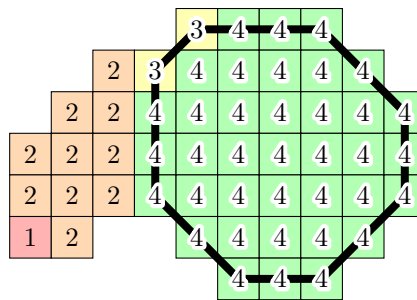


FIGURE 5.32 – Extraction de mesures sur une transformation en ouverture. Les 39 pixels de valeur supérieure ou égale à 3 sont entourés d'un contour épais qui possède 10 arêtes horizontales ou verticales et 7 arêtes diagonales. Il forment une composante connexe unique sans trou.

transformation pour chacune des directions de FREEMAN (modulo 4), s'écrivent :

$$M_0 = \begin{pmatrix} 1 & 0 & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & 1 \\ 0 & 1 & 0 & 0 & 0 & 2 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, M_1 = \begin{pmatrix} 1 & \frac{1}{2} & 0 & \frac{1}{2} & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 2 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix},$$

$$M_2 = \begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{2} & 0 & \frac{1}{2} & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 2 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, M_3 = \begin{pmatrix} 1 & \frac{1}{2} & 1 & \frac{1}{2} & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix},$$

pour des vecteurs de mesures  $\boldsymbol{\mu} = (\# \nu_0 \nu_1 \nu_2 \nu_3 \chi)$ . Ces matrices ont la propriété d'être commutantes, induite par la commutativité de la somme de MINKOWSKI. La dilatation par un zonotope  $Z$  produit d'une somme de segments de FREEMAN se caractérise donc, dans l'espace des mesures, par la matrice :

$$M_Z = M_0^{\nu_0(Z)} \cdot M_1^{\nu_1(Z)} \cdot M_2^{\nu_2(Z)} \cdot M_3^{\nu_3(Z)}.$$

Notons que  $M_0M_2$ ,  $M_1$  et  $M_3$  préservent la différence  $\nu_0 - \nu_2$  et que  $\nu_0 - \nu_2$  n'a pas d'influence sur les autres mesures. Pour s'en convaincre, il suffit de constater que  $T_{02}M_0M_2T_{02}^{-1}$ ,  $T_{02}M_1T_{02}^{-1}$  et  $T_{02}M_3T_{02}^{-1}$  possèdent un coefficient diagonal correspondant à  $\nu_0 - \nu_2$  égal à 1, seul coefficient non nul dans sa ligne et sa colonne ;

$$\text{où } T_{02} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Il en va de même pour  $\nu_1 - \nu_3$  et les matrices  $M_0$ ,  $M_2$  et  $M_1M_3$ .

En conséquence, si  $\nu_0(Z) = \nu_2(Z)$  et  $\nu_1(Z) = \nu_3(Z)$ , les mesures  $\nu_0$  et  $\nu_2$ , d'une part, ainsi que  $\nu_1$  et  $\nu_3$ , d'autre part, peuvent être additionnées. Si  $Z$  est le disque ouvert de rayon  $r$  de la distance définie par la séquence de voisinages  $B$ , alors  $\nu_1(Z) = \nu_3(Z) = \mathbf{1}_B(r-1)$  et  $\nu_0(Z) = \nu_2(Z) = 2 \times \mathbf{2}_B(r-1)$  :

$$M_{D_{<}(O,r)} = ((M_0M_2)^2)^{\mathbf{2}_B(r-1)} \cdot (M_1M_3)^{\mathbf{1}_B(r-1)} \\ = (M_{\mathcal{N}_4})^{\mathbf{1}_B(r-1)} \cdot (M_{\mathcal{N}_8})^{\mathbf{2}_B(r-1)}$$

avec  $M_{\mathcal{N}_4} = M_1M_3$  et  $M_{\mathcal{N}_8} = (M_0M_2)^2$ .

Finalement, pour des vecteurs de mesures  $\boldsymbol{\mu} = (\# \nu_0 + \nu_2 \nu_1 + \nu_3 \chi)$ , les matrices  $M_{\mathcal{N}_4}$  et  $M_{\mathcal{N}_8}$  s'écrivent simplement :

$$M_{\mathcal{N}_4} = \begin{pmatrix} 1 & 1 & 1 & 4 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 4 \\ 0 & 0 & 0 & 1 \end{pmatrix}, M_{\mathcal{N}_8} = \begin{pmatrix} 1 & 1 & 2 & 8 \\ 0 & 1 & 0 & 8 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Tous les éléments sont maintenant réunis pour estimer les mesures des ouverts par les disques à partir des érodés disponibles dans la transformation en distance :

$$\hat{\boldsymbol{\mu}}(\phi_r(X)) = (M_{\mathcal{N}_4})^{\mathbf{1}_B(r-1)} \cdot (M_{\mathcal{N}_8})^{\mathbf{2}_B(r-1)} \cdot \boldsymbol{\mu}(\{p : \text{DT}_X(p) \geq r\})$$

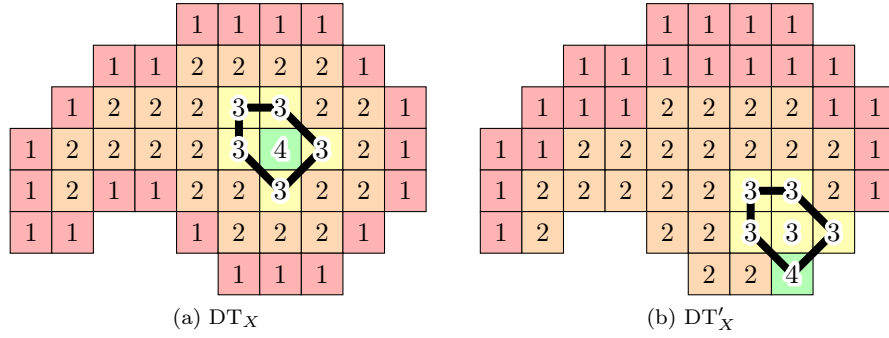


FIGURE 5.33 – Extraction de mesures sur une transformation en distance et une transformation en distance translaturée. Les 6 pixels de valeur supérieure ou égale à 3 sont entourés d'un contour épais qui possède 2 arêtes horizontales ou verticales et 3 arêtes diagonales. Il forment une composante connexe unique sans trou. La distance octogonale a été utilisée.

Le mécanisme est détaillé dans l'algorithme 5.34. Les paramètres mesurés sur la transformation en ouverture (figure 5.32 et tableau 5.30) sont égaux à l'extrapolation à partir de ceux mesurés sur la transformation en distance (figure 5.33 et tableau 5.31). Par exemple, pour le rayon 3 :

$$M_{\mathcal{N}_4} \cdot M_{\mathcal{N}_8} \cdot (6 \ 2 \ 3 \ 1)^T = (39 \ 10 \ 7 \ 1)^T$$

Pour chaque valeur de rayon  $r$ ,  $\{p : DT_X(p) \geq r\}$  et  $\{p : DT'_X(p) \geq r\}$  se distinguent uniquement par une translation, qui par principe n'a pas d'influence sur les mesures. L'algorithme 5.34 accepte donc aussi bien comme données les mesures sur la transformation en distance que sur la transformation en distance translaturée.

---

**Algorithme 5.34:** Extrapolation des mesures sur les ouverts à partir des mesures sur les érodés.

---

**Entrée :** Mesures sur des érodés  $\mu(\{p : DT'_X(p) \geq r\})$

**Sortie :** Estimation des mesures sur les ouverts  $\hat{\mu}(r)$

$M = I$

**début**

**pour**  $r = 1$  **à**  $\max(f)$  **faire**

$\hat{\mu}(r) \leftarrow M \cdot \mu(\{p : DT'_X(p) \geq r\})$

$M = M \cdot M_{\mathcal{N}_{B(r)}}$

**fin**

**fin**

---

## 5.4 Résultats et discussion

La méthode a été testée sur des coupes de reconstruction tomographique d'images de matériaux et comparée aux résultats d'une mesure sur une granulométrie morphologique. La figure 5.35 montre un exemple d'image. Les résultats numériques sont présentés dans le tableau 5.36 et le tableau 5.37 et illustrés sur la figure 5.38. Les écarts numériques sont très faibles sur la mesure

$r$	$\#(\phi_r(X))$	$(\mu_0 + \mu_2)(\phi_r(X))$	$(\mu_1 + \mu_3)(\phi_r(X))$	$\chi(\phi_r(X))$
0	430033	77442	60168	-333
1	424083	67596	63763	122
2	408971	80018	52185	1401
3	382211	67154	59853	2342
4	133233	29324	15325	1216
5	1774	270	209	14
6	0	0	0	0

Tableau 5.36

du nombre de pixels (au maximum 2,25 %). Ils atteignent au maximum 13,33 % sur le nombre de segments horizontaux et verticaux. Cependant, c'est bien le comptage des pixels qui motive cette étude, le but n'étant pas ici de faire un estimateur de courbure, par exemple. Les écarts peuvent dépasser un ordre de grandeur pour la caractéristique d'EULER-POINCARÉ. Mais il faut se souvenir qu'il s'agit d'une différence (entre le nombre de composantes et le nombre de trous), ce qui explique des écarts importants. En l'occurrence, l'erreur la plus forte apparaît pour le rayon  $r = 1$  et coïncide avec la plus petite erreur sur le nombre de pixels (hors  $r = 0$  qui est nécessairement exact).

L'origine des écarts provient de l'hypothèse de portée directionnelle qui n'est pas toujours respectée et des multiplicités apparaissent dans le comptage des points. Bien que l'érosion élimine la plupart de ces problèmes en filtrant les aspérités des contours, elle peut également scinder des composantes et des collisions apparaissent avec une vraie dilatation mais restent invisibles de l'approximation par comptage de configurations locales qui, par principe, a perdu les informations de proximité entre composantes.

Une solution qui a commencé à être étudiée repose sur l'examen de la transformation en distance. En effet, à chaque collision de composantes connexes correspond un point selle dans les valeurs des distances. Inversement, un point selle peut apparaître sans présence de collision et lorsque c'est le cas, il ne donne pas directement le rayon auquel cette collision survient. En suivant les lignes de plus grande pente, il semble possible de retrouver les collisions, de les caractériser et de compenser leurs effets.

## 5.5 Conclusion

Dans ce chapitre, nous avons décrit une nouvelle méthode qui fournit une très bonne approximation de la granulométrie d'une image binaire par l'extrapolation de mesures prises sur une transformation en distance en utilisant une distance à séquence de voisinages. Par rapport à la méthode basée sur la transformation en ouverture, les écarts constatés sont très faibles (au maximum 2,25 % sur le nombre de points). La structure algorithmique rend la méthode utilisable avec des volumes de données très importants, avec des données acquises de manière continue. La complexité temporelle des algorithmes est optimale : linéaire avec le nombre de pixels pour la transformation en distance, pour l'extraction des mesures, linéaire avec le rayon maximal dans l'image pour l'extrapolation des mesures. La complexité spatiale est très réduite : de l'ordre d'une ligne d'image pour la transformation en distance, et égale à la taille des paramètres de sortie pour

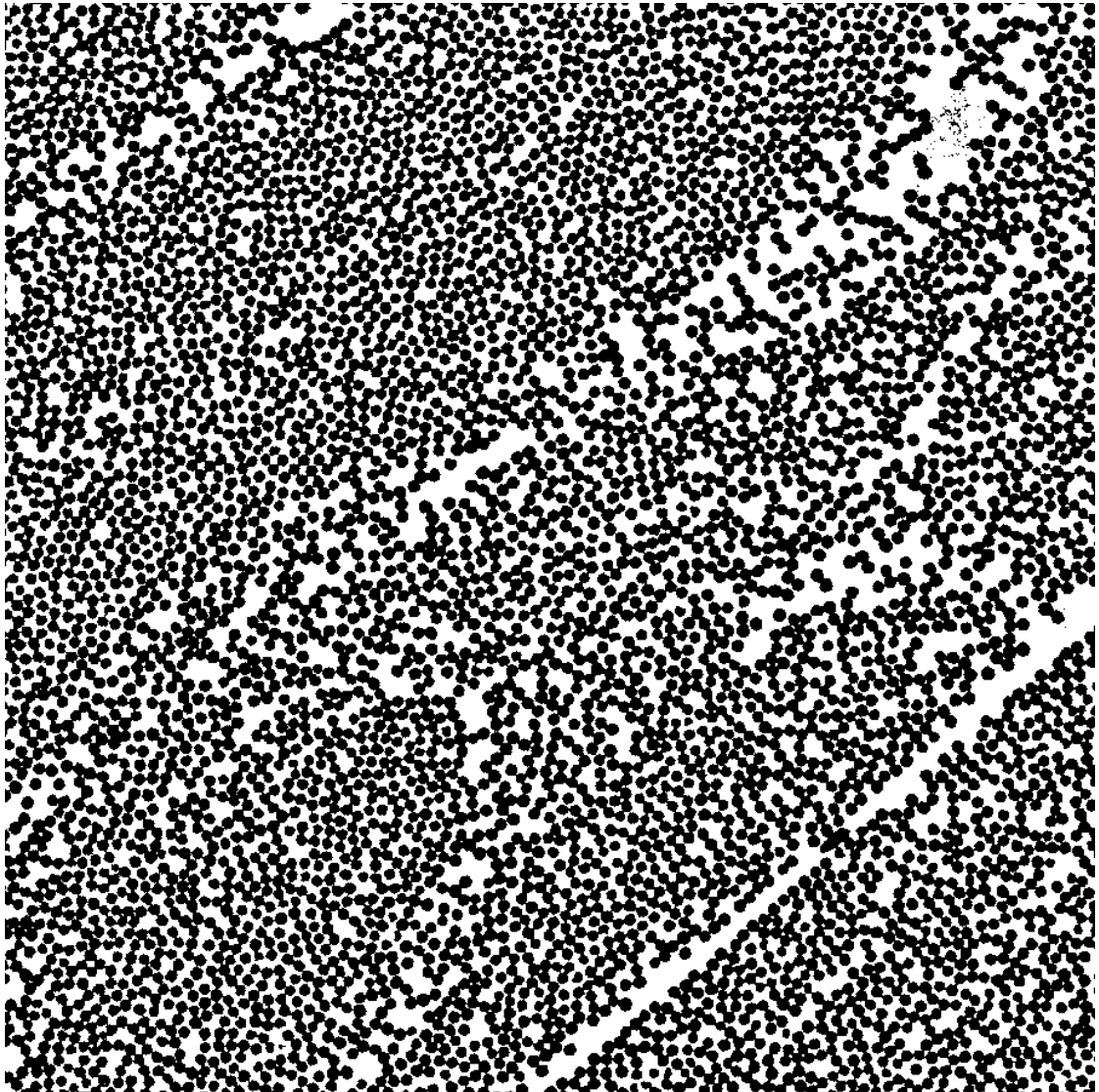


FIGURE 5.35 – Coupe tomographique d’une image de matériau composite (la coupe est perpendiculaire aux fibres). Données fournies par NICOLAS BOYARD, Laboratoire de Thermocinétique de Nantes UMR 6607.



$r$	$\hat{\#}(\phi_r(X))$	$(\mu_0 + \hat{\mu}_2)(\phi_r(X))$	$(\mu_1 + \hat{\mu}_3)(\phi_r(X))$	$\hat{\chi}(\phi_r(X))$
0	430033	77442	60168	-333
1	424443	67680	62646	1386
2	414309	90682	49706	5288
3	384196	72648	56763	5545
4	133896	30332	14700	1790
5	1814	296	219	18
6	0	0	0	0

Tableau 5.37

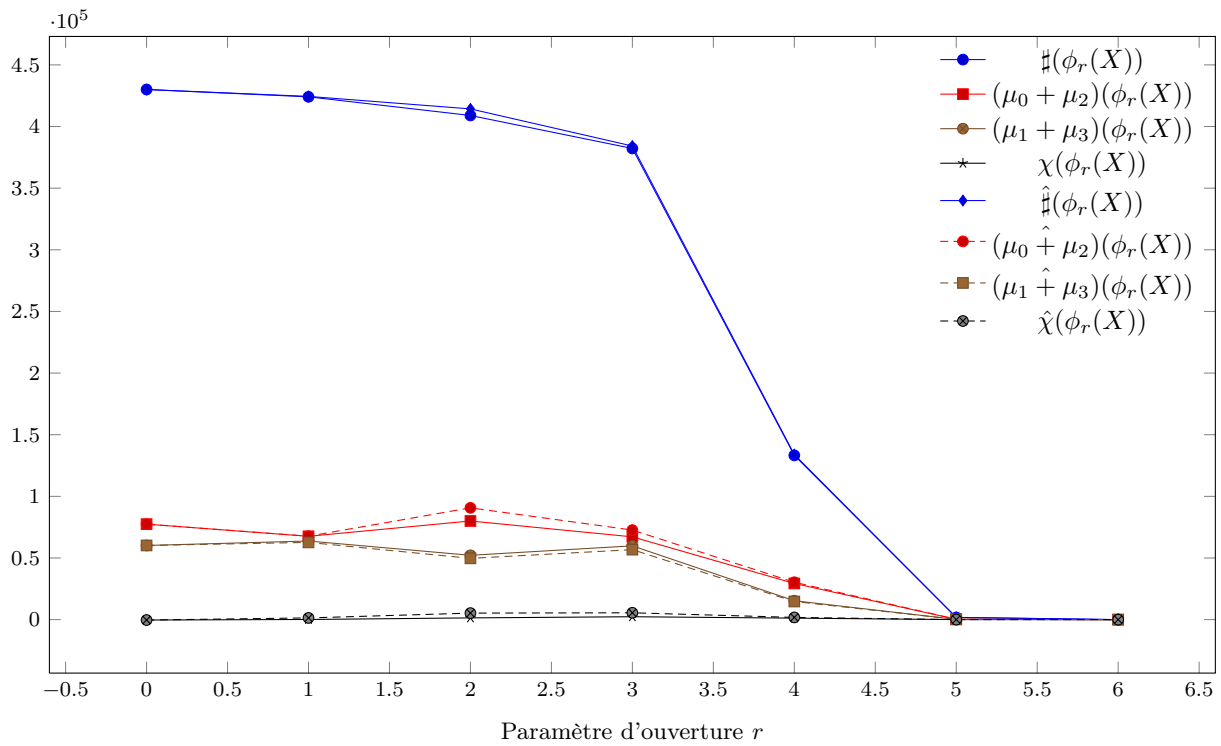


FIGURE 5.38

l'estimation des mesures.

# Perspectives de recherche

## 1 Transformation Mojette et FRT

**Meilleure connaissance des propriétés des fantômes** IMANTS SVALBE, avec ANDREW KINGSTON et SHEKHAR CHANDRA, a commencé la chasse aux fantômes minimaux FRT avec des outils de combinatoire (carrés latins). Nous savons maintenant comment générer des fantômes FRT minimaux par une séquence de produits avec une matrice racine de la matrice identité ce qui lui confère une proximité troublante avec la transformée de FOURIER. IMANTS SVALBE a observé que la transformée de FOURIER 2D d'un fantôme minimal était une combinaison de fantômes minimaux. Pendant mon séjour dans la School of Physics de Monash, des discussions avec Andrew Tirkel et Thomas Hall de la School of Mathematical Sciences, ont montré des liens étroits avec les séquences dites parfaites (dont l'autocorrélation est nulle partout sauf en zéro où elle est maximale) qui ont des applications en transmission de l'information et en stéganographie. Il y a autour des fantômes FRT une convergence étonnante, qui laisse à penser que beaucoup de ses propriétés restent à découvrir et pourraient avoir des applications dans des domaines inattendus.

**Exploitation du formalisme algébrique** Nous avons proposé une représentation polynomiale de la FRT qui permet de représenter la transformée complète (c'est-à-dire dans toutes les directions de projection et non pas une seule) sous forme d'une matrice de VANDERMONDE. Elle a été introduite dans le contexte du codage à effacement avec un algorithme qui s'appuie sur une décomposition des inverses de matrices de VANDERMONDE [120]. Cependant, l'algorithme est indépendant du formalisme polynomial et je suis persuadé que tous les algorithmes FRT précédemment connus et fonctionnant par ligne peuvent être symbolisés et réinterprétés de cette manière. Potentiellement, nous pouvons en attendre une optimisation des méthodes existantes, le développement de nouvelles mais également un rapprochement avec d'autres outils. Est-ce que la FRT et les codes de REED-SOLOMON sont en réalité un code unique? Cette question n'est pas tranchée.

Malgré la proximité des transformations Mojette et FRT, les algorithmes sont fort différents ((1.3) établit malgré tout un pont). Finalement, le formalisme polynomial introduit pour la FRT s'applique tout autant à la transformation Mojette, pour peu que les ensemble de direction aient une certaine forme, qui est précisément celle utilisée dans le contexte du codage à effacement. Un objectif à court terme sera de réutiliser ce formalisme pour la transformation Mojette.

**Développement des applications** Une partie des objectifs précédents sera couverte par la thèse de DIMITRI PERTIN qui s'attachera aussi à déployer concrètement la FRT dans une application de stockage distribué. HERVÉ ANDRIEU et MYRIAM SERVIÈRES sont initiateurs d'un projet dont le but est de reconstruire des images de champ pluvieux par tomographie discrète à partir de mesures d'atténuation, non pas de rayon X, mais d'ondes millimétriques. BAKHTIYOR

ZOKHIDOV débute un doctorat sur ce thème.

## 2 Distances discrètes et morphologie mathématique

**Meilleure connaissance des propriétés des boules** Le comptage des points discrets dans les polytopes est un thème de recherche en tant que tel qui compte des figures telles que MATTHIAS BECK et ALEXANDER BARVINOK [11, 15]. Il fait parfois appel à des méthodes identiques à celles qui ont été utilisées ici, par exemple la triangulation unimodulaire. Alors qu’il est assez simple d’exprimer le nombre de points dans une boule en fonction de son rayon pour les distances à séquence de voisinages (cf. chapitre 5) et pour les distances de chanfrein, nous n’avons pas de résultat dans le cas général. Identifier les passerelles avec des résultats issus de ce domaine permettra probablement de profiter d’outils non exploités dans le domaine des distances discrètes et de la morphologie mathématique.

**Généralisation aux dimensions supérieures** Parmi les algorithmes que nous avons présentés, certains existent ou sont immédiatement généralisables aux dimensions supérieures (transformation en distance, axe médian). D’autres (squelettisation, granulométrie) demanderont des recherches complémentaires. Ce travail mérite d’être entrepris car les gains en temps de calcul et en mémoire sont d’autant plus significatifs que les volumes de données et la dimension sont importantes.

**Développement des applications** Nous avons montré des algorithmes capables de traiter des images ou des volumes à la volée, au fil de leur acquisition : transformation en distance et morphologie mathématique (chapitre 3), axe médian (chapitre 4), squelettisation (cf. thèse de doctorat d’AUREOLE ARLICOT), analyse granulométrique (chapitre 5) et d’autres algorithmes qui n’ont pas encore été publiés (*e.g.* distance de HAUSDORFF). Des prototypes matériels ont été développés avec OLIVIER DÉFORGES et PATRICK LE CALLET et valident notre approche. Nous avons également développé des applications de ces algorithmes, par exemple en analyse d’images de matériaux composites, synthétiques ou naturels tels que les os trabéculaires. Il reste à faire pour déployer nos méthodes vers d’autres domaines d’application, pour d’autres usages et sur d’autres matériels. Les usages et terminaux mobiles, en particulier, sont demandeurs de méthodes économes en ressources, processeur comme mémoire.

Le développement d’applications issues de nos méthodes passe aussi par une dissémination plus large des algorithmes. Bien que la plupart soient déjà accessibles publiquement, ils gagneraient à être intégrés à des plates-formes logicielles (*e.g.* DGtal) ou à être publiés dans des journaux en ligne dédiés à la diffusion du code (*e.g.* IPOL). Cet effort qui a été initié [116], doit être maintenu et amplifié.

## 3 Thèmes connexes

En parallèle d’activités sur la tomographie et les distances discrètes, quelques thèmes feront l’objet de mon attention.

**Synchronisation d’horloges** Ce thème est lié à une collaboration naissante avec JULIEN RIDOUX qui conçoit depuis 5 ans des algorithmes de synchronisation d’horloge par le réseau ou par des capteurs externes (*e.g.* GPS) et les implante sur divers systèmes d’exploitation (Linux, FreeBDS...). Notre implication, qui s’est matérialisée par l’encadrement d’étudiants (JEAN-DANIEL

FISCHER, BENJAMIN VILLAIN) et une soumission dans une conférence internationale [183], est motivée par le besoin de références temporelles précises dans des solutions de stockage distribué et par une possible mise en application de distances discrètes dans l'estimation des biais de mesure.

**Séquences discrètes** Je souhaite terminer en exprimant le souhait de promouvoir l'article fondamental de J. LAMBEK et L. MOSER. Son intérêt pratique pour l'étude des distances discrètes a déjà été relevé mais sa portée va bien au-delà. La notion de séquence inverse est l'équivalent discret de la notion de fonction réciproque qui est bien sûr mentionnée dans tout ouvrage d'analyse. Pourtant, la version discrète est très peu citée. Son lien avec les séquences complémentaires, la fabrication d'itérateurs des valeurs d'une séquence que nous avons établie ( proposition 2.17) lui confèrent un intérêt pratique majeur. Parmi les moyens de faire connaître cet article, j'aimerais entreprendre une traque systématique des séquences inverses et complémentaires dans l'*On-Line Encyclopedia of Integer Sequences* qui pourrait faire l'objet d'une collaboration avec SIMON PLOUFFE, l'un des grands contributeurs de cette encyclopédie hors-norme.



# Sélection d'articles

J'ai choisi de présenter cinq articles représentatifs de mes travaux et de mes collaborations nationales et internationales en recherche.

[26] et [133], traitent de tomographie discrète, le premier utilise les fantômes pour la reconstruction avec des données manquantes, le second décrit un système de transmission à descriptions multiples dans un contexte de réseau local sans fil. [38, 105, 167] sont relatifs au thème des distances discrètes. [38] présente une mise en œuvre matérielle d'algorithmes de morphologie mathématique en niveaux de gris avec une décomposition de formes par des éléments structurants à deux pixels. Dans [167], nous donnons des algorithmes de transformation en distance pour les distances à séquences de voisinages pondérées. Les méthodes proposées dans les quatre articles précédemment cités n'ont pas été exposées en détail dans ce manuscrit. [105] décrit notre algorithme de calcul des voisinages de test et des tables d'inclusion pour les boules des normes de chanfrein (chapitre 4).

- [26] Shekhar Chandra, Imants Dzintars Svalbe, Jeanpierre Guédon, Andrew Maurice Kingston, **Nicolas Normand**, « Recovering missing slices of the discrete fourier transform using ghosts », *IEEE Transactions on Image Processing*, t. 21, n° 10, p. 4431–4441, oct. 2012. DOI : 10.1109/TIP.2012.2206033, 5-year IF : 4.205 (ISI 2010), ERA 17873 A\*.
- [38] Olivier Déforges, **Nicolas Normand**, Marie Babel, « Fast recursive grayscale morphology operators : from the algorithm to the pipeline architecture », *Journal of Real-Time Image Processing*, p. 1–10, juil. 2010. DOI : 10.1007/s11554-010-0171-8.
- [105] **Nicolas Normand**, Pierre Evenou, « Medial axis lookup table and test neighborhood computation for 3D chamfer norms », *Pattern Recognition*, t. 42, n° 10, p. 2288–2296, oct. 2009. DOI : 10.1016/j.patcog.2008.11.014, 5-year IF : 3.453 (ISI 2009), ERA 4503 A\*.
- [133] Benoît Parrein, **Nicolas Normand**, Jeanpierre Guédon, « Multimedia forward error correcting codes for wireless LAN », *Annals of telecommunications*, t. 58, n° 3-4, p. 448–463, mar. 2003.
- [167] Robin Strand, **Nicolas Normand**, « Distance transform computation for digital distance functions », *Theoretical Computer Science*, t. 448, p. 80–93, 24 août 2012. DOI : 10.1016/j.tcs.2012.05.010, 5-year IF : 1.087 (ISI 2010), ERA 18143 A.





# Recovering Missing Slices of the Discrete Fourier Transform using Ghosts

Shekhar S. Chandra, *Member, IEEE*, Imants Svalbe, Jeanpierre Guédon, Andrew Kingston and Nicolas Normand

**Abstract**—The Discrete Fourier Transform (DFT) underpins the solution to many inverse problems commonly possessing missing or un-measured frequency information. This incomplete coverage of Fourier space always produces systematic artefacts called Ghosts. In this paper, a fast and exact method for deconvolving cyclic artefacts caused by missing slices of the DFT using redundant image regions is presented. The slices discussed here originate from the exact partitioning of DFT space, under the projective Discrete Radon Transform, called the Discrete Fourier Slice Theorem. The method has a computational complexity of  $O(n \log_2 n)$  (for an  $n = N \times N$  image) and is constructed from a new cyclic theory of Ghosts. This theory is also shown to unify several aspects of work done on Ghosts over the past three decades. The paper concludes with an application to fast, exact, non-iterative image reconstruction from a highly asymmetric set of rational angle projections that give rise to sets of sparse slices within the DFT.

**Index Terms**—Discrete Radon Transform, Mojette Transform, Discrete Tomography, Image Reconstruction, Discrete Fourier Slice Theorem, Ghosts, Number Theoretic Transform, Limited Angle, Cyclic Ghost Theory

## I. INTRODUCTION

The Discrete Fourier Transform (DFT) is an important tool for inverse problems, where the discrete Fourier representation of an object is used as a mechanism to recover that object. The Discrete Fourier Slice Theorem (FST) property of the DFT, independently developed by Grigoryan and others [1–4], is especially important for discrete tomographic inverse problems, where a discrete object can be recovered exactly from its discrete projected “views” or projections [5].

The Discrete FST provides an exact partitioning of two dimensional (2D) DFT space into a finite number of one dimensional (1D) discrete “slices” of equal length. These slices have discrete slopes  $m$  and  $s$  with  $m, s \in \mathbb{N}_0$ , wrapping around DFT space in both rows and columns, while tiling the space perfectly without interpolation [6]. The partitioning is valid for  $N \times N$  spaces, where  $N = p^n$ ,  $p$  is prime,  $n \geq 1$  and  $p, n \in \mathbb{N}_0$ . This includes prime sizes [7], power of two sizes [8–10] and has also been generalised to composite  $N$  [11, 7]. Rectangular spaces can be handled by zero-padding them to the nearest square size. A total of  $N + N/p$  slices is required to tile all of 2D DFT space exactly [7]. Fig. 1 gives an example of this slice partitioning when  $N$  is prime.

Copyright (c) 2012 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org. S. S. Chandra (Shekhar.Chandra@csiro.au) is with the Australian e-Health Research Centre, CSIRO, Australia. I. Svalbe (Imants.Svalbe@monash.edu) is with Monash University, Australia. A. Kingston is with Australian National University, Australia. J. Guédon and N. Normand are with LUNAM Université, Université de Nantes, IRCCyN UMR CNRS 6597, Nantes, France.

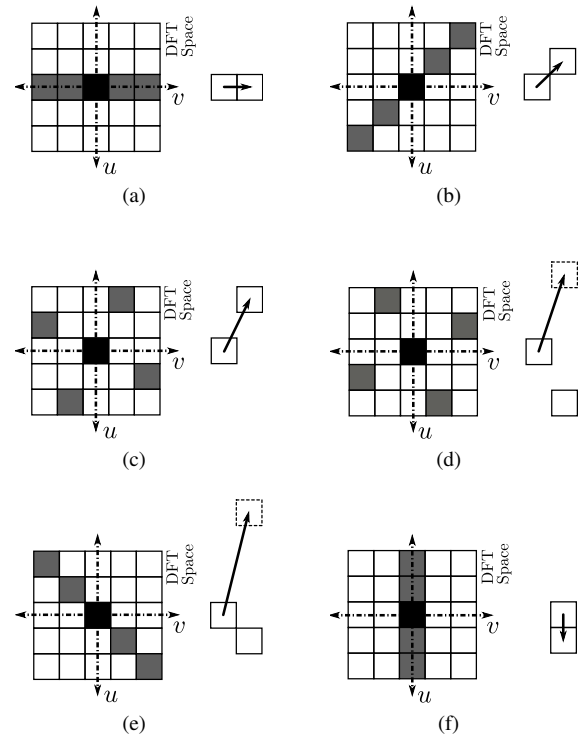


Figure 1. The slices of the DFT having a square size of length  $N = 5$ . Grey pixels in (a)-(e) and (f) each represent a slice of different discrete slopes  $m$  and  $s$  respectively (indicated by their accompanying vectors  $[-m, 1]$  and  $[1, -ps]$ ) with the DC coefficient centred (black). Note that each slice wraps around DFT space (i.e. computed modulo  $N$ ) and the six slices tile all of 2D DFT space at least once without interpolation.

The inverse DFT (iDFT) of the slices corresponds directly to the projections of the Discrete Radon Transform (DRT) [6]. These projections are computed as sums along the lines formed by the vectors  $[1, m]$  and  $[ps, 1]$ , i.e.  $m$  pixels across and one pixel down or one pixel across and  $ps$  pixels down. Fig. 2 gives the equivalent projections for the slices of the DFT shown in Fig. 1. The DRT projections are normally ordered by slope and translate  $t$ , and is often referred to as DRT space. Fig. 3(a) shows the DRT space of an image of Lena.

By computing  $N + N/p$  DRT projections with the set of slopes  $m$  and  $s$  as

$$m = \{m : m < N, m \in \mathbb{N}_0\}, \quad (1)$$

$$s = \{s : s < N/p, s \in \mathbb{N}_0\}, \quad (2)$$

and the set of translates or intercepts  $t$  as

$$t = \{t : t < N, t \in \mathbb{N}_0\}, \quad (3)$$

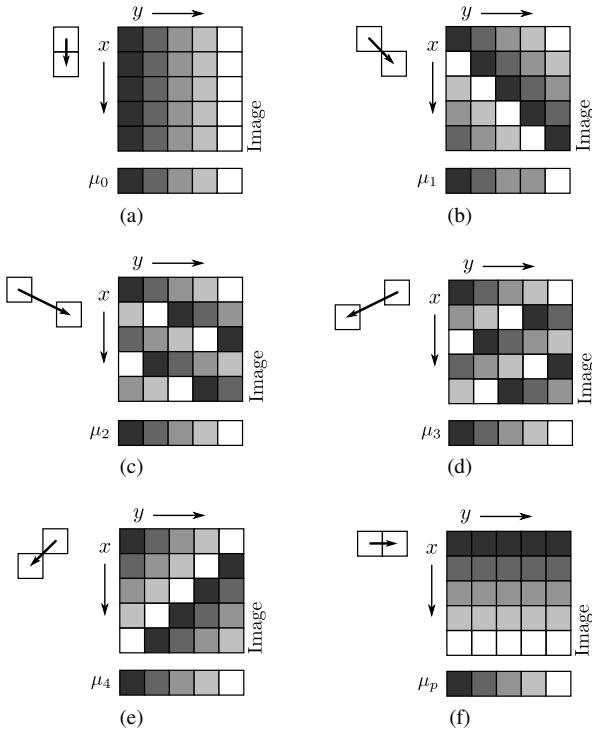


Figure 2. The projections of the DRT that are equivalent to the slices shown in Fig. 1. (a)-(e) and (f) represent projections at different discrete slopes  $m$  and  $s$  (indicated by their accompanying vectors  $[1, m]$  and  $[ps, 1]$ ). Note that each projection wraps around image space (i.e. computed modulo  $N$ ), the coordinate system of the projection vectors are perpendicular to the DFT space slice vectors and that each projection tiles all of image space at least once without interpolation.

where  $x, y \in \mathbb{N}_0$ , the 1D DFT of the projections can be placed into 2D DFT space directly (without interpolation) and the inverse 2D DFT utilised to recover the image. The slices are placed along the vectors  $[-m, 1]$  and  $[1, -ps]$  in 2D DFT space (see Fig. 1). For a prime-sized image, the result is normalised by subtracting the total image sum from each pixel (or equivalently correcting the DC coefficient in DFT space) and dividing by  $N$ . The reconstruction of the image is exact when there is no noise present in the projections.

It is common that not all projections are available to tile Fourier space in discrete tomographic inverse problems. This effectively means there are missing slices or projections, which introduce artefacts in the reconstruction known as discrete Ghosts [12]. Ghost artefacts have zero-sums in the projected directions of the missing data, so that their corresponding Fourier coefficients are also zero [12]. They may be superimposed on an image while still being invisible in the directions of the missing projections. Fig. 4 gives an example of what a Ghost can look like as an image.

Missing projections in the DRT produce cyclic artefacts on the reconstructed image [13, 14] and occur naturally when the DRT is applied to discrete tomography [13], image/erasure coding [15] and image processing [16]. Fig. 3(b) gives an example of missing projections and Ghosts in the DRT.

This paper presents a fast method for exactly removing Ghost artefacts formed from missing slices of the prime-sized DFT. The method utilises redundant or known image regions

within the reconstruction to deconvolve these Ghosts artefacts. The method is constructed from a theory of cyclic Ghosts within the prime-sized Discrete FST that is also developed in this work. This theory for the Discrete FST is constructed using cyclic convolutions and results in algorithms of low computational complexity ( $O(n \log_2 n)$  for an  $n = N \times N$  image). A schematic summary of the process is given in Fig. 3(c). Finally, the Ghost deconvolution method is applied to exactly reconstruct images from a highly asymmetric set of rational angle projections that give rise to sets of sparse slices within the DFT.

The paper is structured as follows. Previous work on Ghosts and reconstructing from sparse signals is reviewed in the next section. This is followed by a theory of cyclic Ghosts for the Discrete FST in Sec. III. The convolution and deconvolution techniques for missing slices of the DFT is presented in Sec. IV. Finally, the results for applying these techniques in reconstructing an image exactly from a highly asymmetric set of rational angle projections is presented in Sec. V.

## II. PREVIOUS WORK

Bracewell and Roberts [17] introduced the concept of the “invisible distribution”, later referred to as Ghosts by Katz [12] and Cornwell [18], in the context of the Fourier Transform (FT). These distributions are a consequence of non-unique solutions arising from an incomplete Fourier space or “ $u, v$  coverage” due to a finite number of measurements. Explicitly, a classical Ghost or invisible distribution exists where a continuous theory, i.e. one that utilises an integral transform, is the mechanism for an inverse problem. For example, the theory for Computed Tomography (CT) is commonly based on the FT, and is therefore ill-posed as a result and always has classical Ghosts [19]. Katz [12] rediscovered the concept of a Ghost and showed that Ghosts must have zero-sums in the directions of the missing projections.

Logan [20] presented the uncertainty principle for projections in the FT and determined that Ghosts are mostly present in the high frequencies. He proved why low-pass filtering of the projections, using filters such as the Ram-Lak [21] or Shepp-Logan [22] filters, is required when reconstructing from projections in the FT. These and other related filters are still being used today in modern CT [23]. Louis [24] reformulated this result in a simpler form using the Hankel Transform.

Katz [12] presented the uncertainty principle for discrete rational angle projections for square image sizes, where a projection is computed as sums along the lines formed by using the vector  $[b, a]$  with  $a, b \in \mathbb{Z}$ . These acyclic projections constitute what is now known as the Mojette Transform (MT) [25]. Fig. 5 shows a simple example of a MT for a  $4 \times 4$  image using three projections.

Katz [12] determined that an  $N \times N$  image can be reconstructed exactly from a set of  $\mu$  rational angle projections  $[b_j, a_j]$  if and only if

$$N \leq \max \left( \sum_{j=0}^{\mu-1} |a_j|, \sum_{j=0}^{\mu-1} |b_j| \right). \quad (4)$$

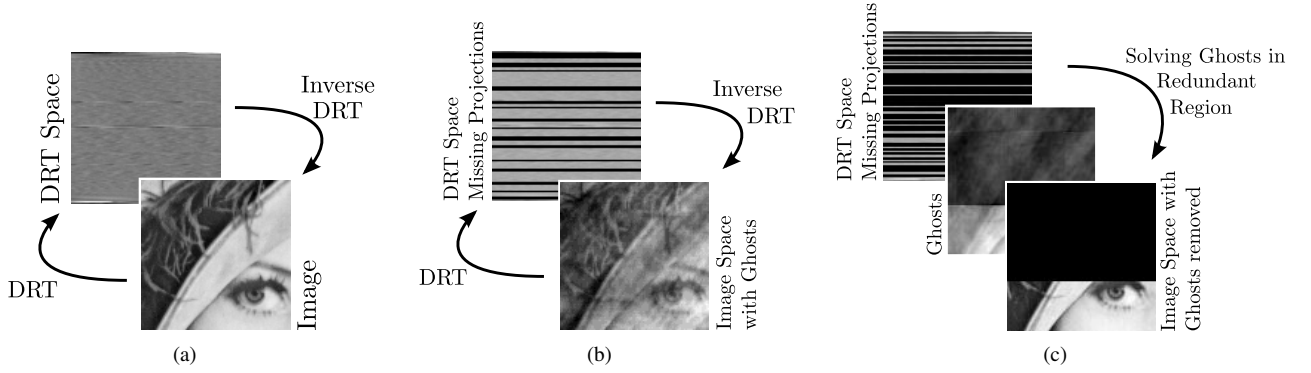


Figure 3. An illustration of the Discrete Radon Transform (DRT) and its cyclic Ghosts for an image of Lena. (a) shows the DRT space of the image of Lena and its one-to-one nature. (b) shows missing projections (black rows in DRT space) and their effect on the reconstructed image. (c) shows the result of deconvolving the Ghosts or “De-Ghosting” an image with Ghost artefacts in order to restore the image when a redundant image area is present.

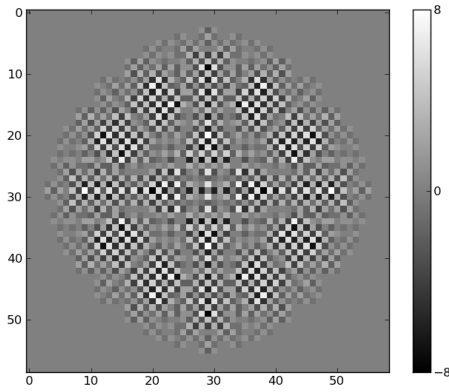


Figure 4. An example of a Ghost represented as an image. This Ghost is invisible at 24 rational angle projections and may be superimposed on an image without changing the projections of the image for these 24 angles.

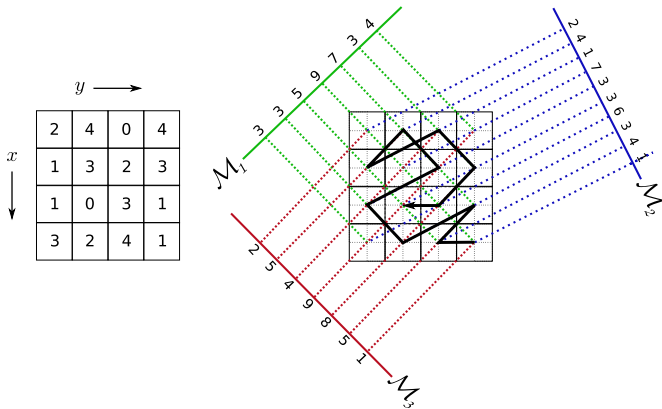


Figure 5. An example of a Mojette Transform for a discrete image of size  $4 \times 4$  using the three projections  $[1, 1]$ ,  $[1, -1]$  and  $[1, -2]$ . The bold lines within the right-hand grid shows a possible reconstruction path using a corner-based reconstruction method [26].

This is now known as the Katz criterion. It is a statement that the information contained in the projection set needs to be one-to-one with the image data. When the criterion is not met, discrete Ghosts superimpose on the image because of the ambiguity in the projections [12]. These Ghosts look similar to the image in Fig. 4.

Normand *et al.* [27] extended the Katz criteria to arbitrary convex regions using mathematical morphology. They proved that a set of rational angle projections can only be reconstructed unambiguously if and only if the largest possible Ghost is larger than the convex region being reconstructed, so that there is no ghost present in the region due to the projection set. This less stringent criteria for the MT allows one to reconstruct images from a highly asymmetric set of projections [28]. These projections have a limited coverage in terms of the half-plane, i.e. the interval  $[0, \pi)$ , but are sufficient for an exact reconstruction in terms of the Katz criterion (4). A number of schemes have been proposed for these types of MT projections, including a Conjugate Gradient method [29] and a Geometric Graph approach [26], but the former is not suitably convergent and the latter is very sensitive to noise. Sec. V will present a fast alternative method based on the slices of the DFT and their Ghosts.

Highly asymmetric projection sets also occur in conventional limited angle tomography. Boyd and Little [30] presented a solution to limited angle tomography by reducing Ghosts through the fusion of multi-modal data to improve Fourier coverage. Techniques that minimise the  $L_1$ -norm are a common approach to reduce Ghost artefacts in reconstructions [31–34]. Vetterli *et al.* [35] showed that  $\mathcal{X}$  spikes can be recovered exactly from  $2\mathcal{X} + 1$  consecutive Fourier samples by solving a system of equations. Work by Kuba [36] inspired many others (for a recent example see [37]) to develop the theory of “switching components”. They considered the problem of uniquely reconstructing all entries in a matrix from its directed sums, based initially along the matrix rows and columns. These switching components are structures that prohibit unique inversion. They are equivalent to the Ghosts discussed here.

In recent work, Candès *et al.* [38] showed that an  $N \times N$  image can be recovered exactly using convex optimisation of a very small number of projections. However, their method suffers from high computational complexity and is still an active area of research [39]. Herman and Davidi [40] discussed the Candès *et al.* [38] result using Ghosts and showed that Ghost artefacts may still remain invisible to small number of projections when using their method. Nevertheless, their work paves the way in using Ghosts for analysing the sensitivity

of reconstruction algorithms. Inspired by their work, the next section constructs a cyclic theory of Ghosts for the Discrete FST.

### III. CYCLIC GHOSTS

In this section, it is shown that Ghosts of the Discrete FST have exact cyclic forms known as  $m$ -Circulant matrices (or circulants).

**Definition 1** ( $m$ -Circulant<sup>1</sup> [41]). *An  $m$ -Circulant is an  $N \times N$  matrix containing a unique row  $f(j)$  with  $j = 0, \dots, N - 1$  replicated on each row, but where each row is cyclically shifted  $(\text{mod } N)$  by an additional  $m$  elements to the right.*

$m$ -Circulants represent the slices of the Discrete FST via its Fourier “diagonalisation” property. A 2-Circulant is diagonalised by the DFT as illustrated in (5).

$$\begin{bmatrix} \mathbf{a}_0 & a_1 & a_2 & a_3 & a_4 \\ a_3 & a_4 & \mathbf{a}_0 & a_1 & a_2 \\ a_1 & a_2 & a_3 & a_4 & \mathbf{a}_0 \\ a_4 & \mathbf{a}_0 & a_1 & a_2 & a_3 \\ a_2 & a_3 & a_4 & \mathbf{a}_0 & a_1 \end{bmatrix} \xrightarrow{\text{(i)DFT}} \begin{bmatrix} \lambda_0 & & & & \\ & \lambda_1 & & & \\ & & \lambda_2 & & \\ & & & \lambda_3 & \\ & & & & \lambda_4 \end{bmatrix} \quad (5)$$

The diagonalisation is represented by monomial matrices  $M = PA$ , where  $A$  is an  $N \times N$  diagonal matrix and  $P$  is an  $N \times N$  permutation matrix with  $N$  non-zero elements representing a discrete line at slope  $m \pmod{N}$  [41]. Hence, the lines of slope 1 given in part (b) of Fig. 2 is a 1-Circulant. The lines of slope 2 (as in Fig. 2(c)) will be a 2-Circulant and so on.

Every projection can then be represented by an  $m$ -Circulant and its 2D DFT represents a slice in discrete Fourier space. Equivalently, since the 1D DFT of the unique circulant row are the diagonal values [41], the 1D DFT of the (1D) projections are the slice values, so that only a computation complexity of  $O(N \log_2 N)$  is required to compute each slice. Superimposing a circulant in image space places a slice of corresponding slope in DFT space. The tiling of the diagonals is the same for the lines formed by the vectors  $[-m, 1]$  and  $[1, -ps]$ , so a total of  $N + N/p$  diagonals is required for full tiling of DFT space. The result is a sum of  $N + N/p$  circulants in image space to recover the image. This is known as Circulant Back-Projection (CBP) [42] and is illustrated in Fig. 6.

Fill [4] showed that the Discrete FST can be constructed using circulant matrices. Chandra and Svalbe [42] showed that the construction can be extended to other DFT-like transforms, such as the Number Theoretic Transform (NTT). The NTT will be important in removing the cyclic Ghosts exactly (see discussion at the end of this section).

The circulant matrices also allow one to naturally define and understand Ghosts. In what follows, we restrict ourselves to prime-sized image spaces, so that projections are acquired via the vectors  $[1, m]$  and  $[0, 1]$  with a total of  $N + 1$  projections required for an exact reconstruction.

**Proposition 1** (Cyclic Ghosts). *Each missing projection  $\mathbf{a}'$  at  $[1, m_a]$  in the DRT, which corresponds to the missing slice of slope  $m_a$  in the DFT, forms artefacts superimposed on the reconstructed image in the form of a  $m_a$ -Circulant. The unique*

<sup>1</sup>Also referred to as a generalised circulant or  $g$ -circulant [41].

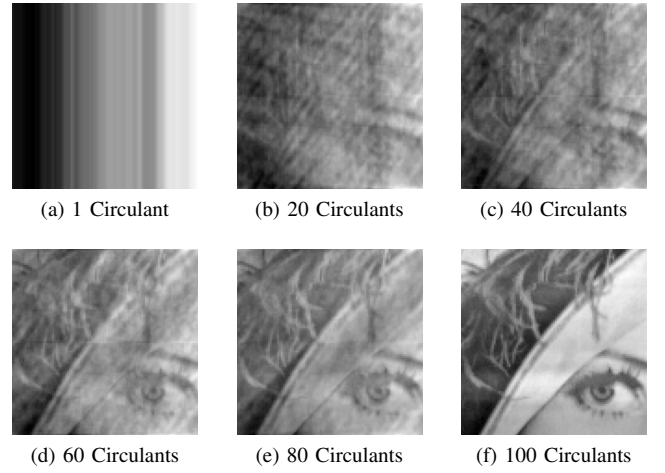


Figure 6. Illustration of the CBP of a  $101 \times 101$  image of Lena using different number of circulants. (a)-(f) show the effect of an increasing number of slices placed into DFT space on the image. The artefacts on the partial reconstructions are cyclic Ghosts.

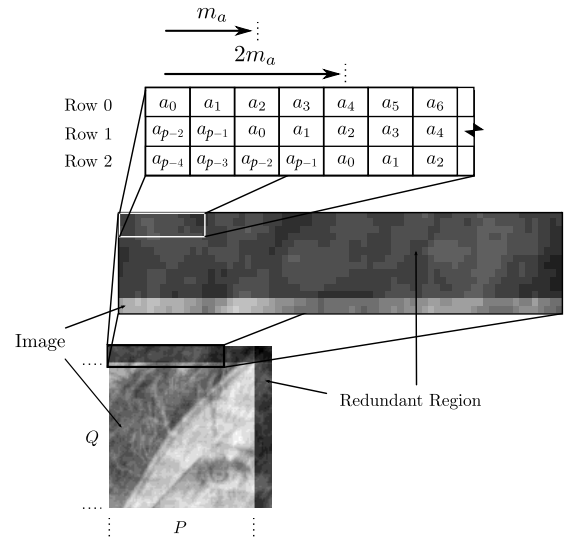


Figure 7. The  $Q \times P$  image (Lena) is of  $100 \times 100$  pixels embedded in a image space  $N = 113$  with the remaining pixels equal to zero. Assuming one missing projection  $\mathbf{a}'$  at  $m_a = 2$ , then the table shows the circulant artefacts that become embedded over the reconstructed image values (including known areas) where  $\mathbf{a} = -\mathbf{a}'$ .

row of the circulant is  $\mathbf{a} = -\mathbf{a}'$  and these artefacts are called cyclic Ghosts.

*Proof:* When projections are missing, the CBP is incomplete since there must be  $N + 1$  projections for an exact reconstruction. Each projection is equivalent to a circulant in the reconstruction process. Thus, the remaining missing contributions must be a superposition of  $\mathcal{N}$  number of circulants with shifts  $m_\ell$ , where  $\ell = 0, \dots, \mathcal{N} - 1$  and  $\mathcal{N}$  represents the number of missing projections. The Ghosts are negative-valued since they are missing contributions in the reconstruction. A schematic of the circulant nature of Ghosts is shown in Fig. 7. ■

Chandra *et al.* [13] showed that given  $\mathcal{N}$  redundant or known image rows,  $\mathcal{N}$  cyclic Ghosts can be removed exactly by shifting



and subtracting redundant rows from the rows that contain image data. Although the algorithm allowed exact recovery of images with missing projections, it required the use of arbitrary precision integers as values grew to the order of  $10^{40}$  when removing just 80-90 Ghosts for large images. The result was an algorithm that had very high computational complexity and one that gave no understanding of the cause of such large values. In the next section, a deconvolution approach is constructed which has a lower computational complexity and shows why such large values arise.

#### IV. DE-GHOSTING

This section presents a deconvolution algorithm for removing cyclic artefacts formed from  $\mathcal{N}$  missing slices in the Discrete FST. Given  $\mathcal{N}$  redundant image rows or columns and a list of unknown projection slopes  $m_\ell$ , a Ghost deconvolution or De-Ghost filter is constructed for an image size  $N$ . The filter is then used to deconvolve the Ghost artefacts from the image rows. Note that an open source implementation of all the algorithms constructed in this paper can be found in the Finite Transform Library written by the authors [43].

**Proposition 2** (Ghost Convolution). *Ghosts formed from the missing projections of the DRT are cyclic convolutions of these projections in the geometry of the DFT.*

*Proof:* According to the Discrete FST, each slice is a cyclic smearing or back-projection of its corresponding projection in image space [4, 42]. Within the geometry of the DFT, convolutions can be represented as a sum of  $m$ -Circulants [41]. Thus, the Ghosts that arise due to empty (i.e. zero) projections in the DRT result in an incomplete back-projected image, which are a sum of the missing  $m$ -Circulants, and are convolutions in image space. ■

**Proposition 3** (Ghost Kernels). *The Ghost convolution kernels required for De-Ghosting are the vectors  $[1, m_j]$ , which sum to zero along its vector for each of the missing projections  $j = 0, \dots, \mathcal{N} - 1$ .*

*Proof:* The projections of Ghosts are zero-valued at their corresponding  $m$ -values, so the projection vectors  $[1, m_j]$  must also be Ghosts at their  $m$ -values in order that Ghosts annihilate with shifted versions of themselves. The kernels apply to each bin in the missing projection or Ghost row (using Prop. 2). An example of a Ghost kernel is shown in Fig. 8(a)-(c). ■

Using Prop. 3, the De-Ghost filter is constructed in the following way:

- 1) Create a 2D convolution kernel for each unknown slope  $m_\ell$  by placing a +1 at the image origin and -1 at  $(1, m_\ell)$ .
- 2) Convolve these kernels in 2D to obtain the De-Ghost filter.

The convolution of the kernels may be done either in 2D (in Fourier or image space), or as a set of 1D convolutions on the projections via the Projection Convolution Theorem (PCT) [6] (see Fig. 8). The advantage of the latter is efficiency, especially when the number of known projections  $\mu \ll N$ .

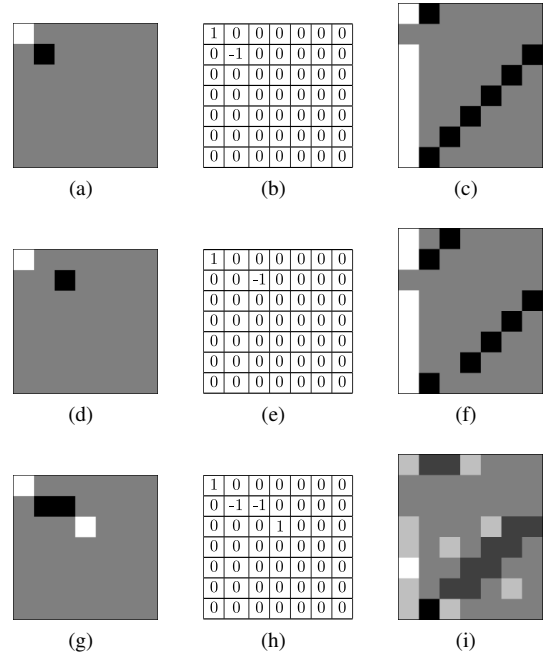


Figure 8. An example of the 1D convolution approach to constructing 2D Ghosts, where grey and black denote zero and minus greyscale values respectively. (a) and (d) show the kernels  $[1, 1]$  and  $[1, 2]$  respectively. (c), (f) and (i) show the DRT projections of the kernels (a) & (d) and their convolution respectively.

##### A. Projection Convolution Theorem

The PCT states that a 2D convolution is equivalent to the 1D convolution of each projection or slice in Fourier space. Thus, the 2D Ghost convolution can be computed as a series of 1D convolutions on the known projections. For Ghost convolution, when the number of Ghosts  $\mathcal{N}$  is close to  $N$ , DRT space is sparse and the 2D convolution is computed over a small number of 1D signals.

The PCT can be interpreted as a consequence of the Discrete FST. Since the slices tile DFT space exactly and are the DFT of the projections, cyclically convolving the slices of two objects is equivalent to cyclically convolving the objects themselves. Thus, in order to utilise the PCT, one needs to know the projections of the two objects. In this case, the projections of the Ghost kernels are particularly convenient, which makes a PCT approach very efficient.

**Proposition 4** (Kernel Projections). *The projections of the 2D convolution kernel  $[1, m_\ell]$ , with a positive term at the origin and a negative term at the coordinate  $(1, m_\ell)$ , will have the positive term at the zeroth translate and the negative term at translate  $t_j$  as*

$$t_j = (m_\ell - m_j) \pmod{N} \quad (6)$$

for each projection  $m_j$  with  $j = 0, \dots, N-1$  and  $\ell = 0, \dots, \mathcal{N}-1$ . For the  $j = N$  projection, negative term is always at  $t_j = 1$ . Note that negative values  $-x$  modulo  $N$  are equivalent to the value  $N - x \pmod{N}$ .

*Proof:* The projection  $m_\ell$  will be zero by definition with all other projections being non-zero. For projections with  $m_j < m_\ell$ ,

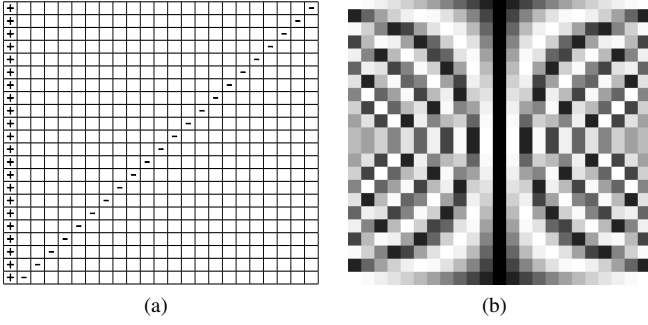


Figure 9. A schematic of the 1D Ghost convolution operators (a) for  $N = 23$ . Each row represents the unique projection of a Ghost convolution kernel. (b) shows their eigenvalues in (centred) discrete Fourier space.

the negative term will appear in the translates  $t > 0 \pmod{N}$  because the initial vector  $[1, m_j]$  for  $t = 0$  will not sample the point  $(1, m_\ell)$ , since slope of the line is smaller than  $m_\ell$ . The difference of slope  $(m_\ell - m_j)$  defines the translate where the point is eventually sampled. For projections with  $m_j > m_\ell$ , the negative term will appear in the translates  $t < 0 \pmod{N}$ , i.e.  $N - t \pmod{N}$ , because the initial vector  $[1, m_j]$  will not sample the point  $(1, m_\ell)$  until the vector wraps around the image, since the slope of the line is more than  $m_\ell$ . Hence, the difference of slope  $(m_\ell - m_j)$  will be negative and defines the translate where the point is eventually sampled. ■

**Corollary 1 (Kernel Operators).** *The projections of the kernels  $[1, m_\ell]$  only have  $N$  distinct combinations of positive and negative values.*

*Proof:* This follows from Prop. 4 and the fact that the system is constructed within a cyclic geometry, i.e. because of the wrapping of the values since there are only  $N$  residue classes. ■

An example of the unique kernel projections (following Cor. 1) and their eigenvalues are shown in parts (a) and (b) of Fig. 9 respectively. These eigenvalues can be precomputed, since they depend purely on the set of slopes of the missing projections as given by Prop. 4. The form of the DFT eigenvalues for  $N = 479$  is provided as supplementary material.

Therefore, the 1D Ghost convolution approach may be used to generate a 2D Ghost as follows (see also Fig. 8):

- 1) Pre-compute the Ghost operator eigenvalues via the DFT for a given  $N$ . The results can be used as a hash table to pick out the relevant operator based on the projection being convolved.
- 2) Convolve a discrete delta function with each of the kernels  $[1, m_\ell]$  by selecting the correct eigenvalues for each operator using the eigenvalues hash table (as given by previous step) via Eq. (6) and multiplying these operators with the eigenvalues of the delta function.
- 3) Repeat the 1D Ghost convolutions for all known projections  $m_j$ .
- 4) Compute the inverse DFT of the result from 3 to obtain the Ghost structure in image space.

The computational complexity is  $O(\mathcal{N}N \log_2 N)$  for pre-computing the eigenvalues,  $O(\mu N \log_2 N)$  for the 1D DFTs

of the known projections and  $O(\mu \mathcal{N}N)$  for computing the convolutions. When  $\mu \ll N$ , the algorithm has a computational complexity of  $O(n \log_2 n)$  (where  $n = N^2$ ). However, the convolution is susceptible to round-off errors and loss of precision for large  $\mathcal{N}$  when utilising the DFT for the convolutions. This problem, which manifested as numerical overflow, was also encountered by Chandra *et al.* [13] with their method. This numerical growth can be easily seen as a direct consequence of the convolutions of the eigenvalues in Fig. 9. The solution is to use the NTT and the Number-Theoretic Radon Transform (NRT) of Chandra and Svalbe [42].

## B. Number Theoretic Convolution

The NTT allows one to compute convolutions using only integers because the unit circle is replaced with the digital “circle”

$$\alpha^{M-1} \equiv 1 \pmod{M}, \quad (7)$$

so that  $\alpha^{M-1} - 1$  is a multiple of  $M$ ,  $N$  is a multiple of  $M - 1$  and  $\alpha, M \in \mathbb{N}_0$  [44]. The successive powers  $\{1, \dots, M - 1\}$  of  $\alpha$  generates a unique set of integers in some order modulo  $M$ . Such a number  $\alpha$  is called the primitive root. The primitive root(s)  $\alpha$  in these cases have to be found by trial and error and can be computed by dividing  $M - 1$  by the prime factors  $p_j$  of  $M - 1$ , such that  $\alpha^{(M-1)/p_j} \not\equiv 1 \pmod{M}$ , where the trial value of  $\alpha$  is prime.

For a prime-length  $N$ , the NTT can be computed by first selecting the modulus  $M$  as  $M = k \cdot N + 1$ . This allows the power of  $\alpha$  in Eq. (7) to be a multiple of  $N$ . For example, the modulus for the prime-length  $N = 101$  is 607 with  $k = 6$ . Then Rader’s [45] algorithm can be used to compute its fast form. Integer coefficients allow computations to be done without round-off error or numerical overflow, since the results are congruent modulo  $M$  [46].

Chandra and Svalbe [42] showed that the discrete FST still holds within the 2D NTT when placing Number Theoretic slices, i.e. the NTTs of the projections, into 2D NTT space. This allows one to replace the DFT in all computations, including those within the Ghost convolutions, with the NTT. The resulting NRT was constructed specifically to remedy the loss of precision when forming cyclic Ghosts. Chandra and Svalbe [42] also showed that the implementation of the NTT is faster than the DFT because of its integer-only operations. Consequently, the Ghost convolution method is impervious to numerical overflow and faster than DFT approaches.

Ghost in the NRT have another important property in that any physical or perceivable structures in the Ghosts are difficult to discern. The Ghosts in this space have the value  $0 \pmod{M}$  in the direction of the missing projections. This makes the NRT Ghosts well suited for encoding and encryption. An example of Ghosts within the NRT can be seen in Fig. 10(b). Once the Ghost is constructed/convolved, it can be used to remove the artefacts formed by the missing slices for which it was constructed. This process of De-Ghosting is described in the next section.

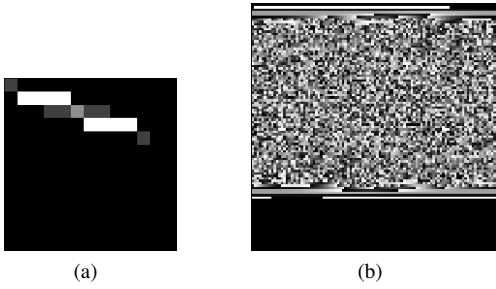


Figure 10. Examples of Ghosts in the NRT. (a) shows a Ghost with  $\mathcal{N} = 4$  at  $m = 1, 2, 3, 4 \pmod{53}$ , where  $N = 13$ . (b) shows a Ghost with  $\mathcal{N} = 80$  at  $m = 1, \dots, 80 \pmod{607}$ , where  $N = 101$ .

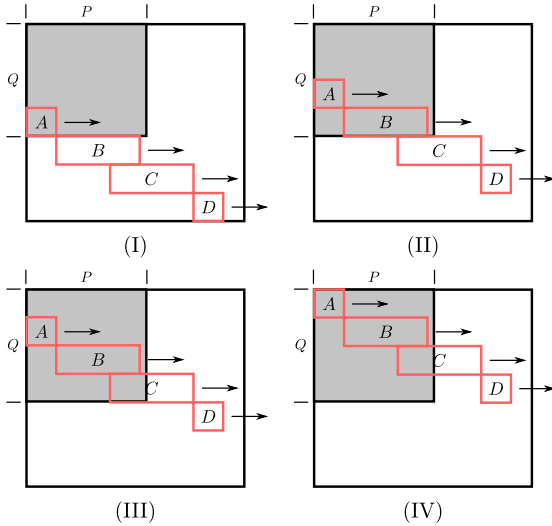


Figure 11. A De-Ghost example for a  $Q \times P$  image (in grey), where a row of the De-Ghost filter (in red) is aligned to an image row  $\tilde{I}(3,0)$  (with the remaining filter rows in redundant space) and the row-wise 1D convolution of the filter (with the Ghosted image rows) recovers the image row  $\tilde{I}(3,0)$ . The result is back-substituted to create a redundant row and the process repeated, as in (II), (III) and (IV), to recover the image.

### C. Ghost Deconvolution

The Ghost constructed in the previous section may be applied as an exact deconvolution mask/filter to recover the missing slices of the Discrete FST. The filter is aligned to an image row to be De-Ghosted and the deconvolution computed in one of two ways.

A simple but inefficient deconvolution method is to zero-pad the sub-image that contains the image row to be recovered, as well as the required  $\mathcal{N}$  redundant rows, within the  $N \times N$  space and compute the 2D convolution with the filter. A more efficient approach is to individually convolve each row of the filter with the corresponding row of the image as shown in Fig. 11. The result of each 1D convolution would then need to be summed to the image row being recovered. For example, the sum of convolutions in Fig. 11(I) is effectively a sub-circulant expression given as

$$\mathbf{A} \cdot \tilde{I}(3,0)^T + \mathbf{B} \cdot \tilde{I}(4,0)^T + \mathbf{C} \cdot \tilde{I}(5,0)^T + \mathbf{D} \cdot \tilde{I}(6,0)^T = I(3,0)^T, \quad (8)$$

where  $\tilde{I}$  denotes the Ghosted image,  $I$  denotes the desired De-Ghosted sub-image and the matrices  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$  and  $\mathbf{D}$  are

circulants defined by the rows  $A, B, C$  and  $D$  of the filter in Fig. 11. These circulants represent the 1D convolutions of the rows of the filter. The resultant image row is back-substituted and the filter translated up or down to recover the next row. In this case, row  $I(3,0)$  is back-substituted into  $\tilde{I}(3,0)$  in order to make it a redundant row and the filter is translated upwards to the next row and Eq. (8) is repeated for (II). The process is repeated for all image rows until all the Ghosts are removed.

In the 1D convolution method, the  $m$ -values of the Ghosts need to be  $-m_\ell$  (i.e.  $N - m_\ell$  rather than  $m_\ell$ ) when convolving the Ghosts to undo the right shift of the  $m$ -Circulants. A total of  $\mathcal{N} + 1$  rows need to be convolved, resulting in a computational complexity of  $O(Q\mathcal{N}N)$ . When  $\mu = \min(P, Q) + 1$  and  $\mu \ll N$ , the algorithm has a computational complexity of  $O(n \log_2 n)$  (where  $n = N^2$ ). A visual interpretation to constructing Ghosts, that unifies the Ghost recovery algorithm of Chandra *et al.* [13], Latin squares approach of Chandra and Svalbe [14] and the 2D convolution approach of this paper, can be made using  $n$ -gons (see the thesis of Chandra [47]).

More work has to be done with both methods when the projections or slices contain noise or inconsistencies as these also become convolved during the De-Ghost process. The convolution approach requires a very good estimation of noise prior to De-Ghosting, so that the estimates may be used to deconvolve their effects on the results. Future work includes generalising the approach to arbitrary missing discrete Fourier coefficients. Recent work by Svalbe *et al.* [48–50] discussed the minimal extent of cyclic Ghosts, which may prove useful in this endeavour. In the next section, the De-Ghost method is applied to the discrete inverse problem of determining a reconstruction from a set of highly asymmetric rational angle projections.

## V. APPLICATION: DISCRETE RECONSTRUCTION

Chandra *et al.* [13] utilised their Ghost removal technique to exactly reconstruct an image from rational angle (noise-free) discrete 1D projections of the MT. The projections sets in their work required covering the half-plane, i.e. the interval  $[0, \pi)$ . In this section, the De-Ghost method will be applied to reconstructing from a set of rational angle projections with arbitrary coverage of the half-plane, such as those within a quadrant or the interval  $[0, \pi/2)$ , provided there are  $\min(P, Q) + 1$  projections for a  $Q \times P$  image. Both sets require that the projections satisfy the Katz criterion (4) for exact reconstruction.

In the De-Ghost reconstruction algorithm, a  $Q \times P$  image is zero padded into an  $N \times N$  space, where  $N \geq \max(P, Q)$ . A total of  $\min(P, Q) + 1$  projections are required to ensure that the number of redundant image rows is equal to the number of missing slices  $\mathcal{N}$ , since there will be  $N + 1$  slices within DFT space. An efficient new method for generating  $\min(P, Q) + 1$  highly asymmetric rational angles is presented in the Appendix. Assuming that  $\min(P, Q) + 1$  of this type of projections have been acquired, the remaining parts of this section will demonstrate the performance of the De-Ghost algorithm in reconstructing these rational angle projections exactly, where a large number of Ghosts are present.



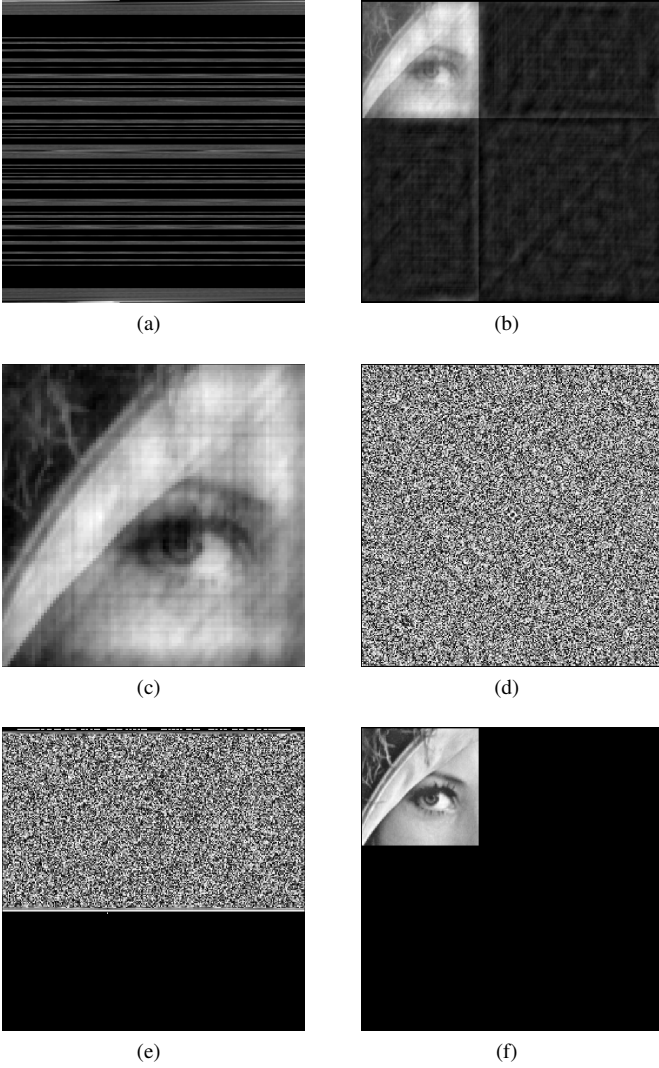


Figure 12. The results of the De-Ghost method applied to discrete reconstruction. A total of 101 asymmetric rational angle projections of a  $100 \times 100$  image of Lena was exactly reconstructed within a DFT space of size  $257 \times 257$ . (a) shows the resulting DRT space. (b) shows the resulting Ghosts when (a) is reconstructed. (c) shows a cropped version of (b) focussing on the embedded image. (d) and (e) show the Ghost eigenvalues and the Ghost that was used to recover the reconstructed image exactly (shown in (f)).

Consider a  $100 \times 100$  image of Lena and 101 rational angle projections similar to the geometry of Fig. 14(a). Also let  $N = 257$  for performance purposes, so that  $N - 1$  is a power of two. Thus, there will be 156 Ghosts to remove and also 156 redundant rows in the image, so that the De-Ghost method can be applied.

Fig. 12(a) shows the DRT space resulting from 101 rational projections acquired across the quadrant and mapped via Eq. (9). Fig. 12(b) and (c) show the resulting Ghosts superimposed on the reconstruction due to the missing projections in (a). Fig. 12(d) shows the NTT Ghost eigenvalues that are used to construct the deconvolution Ghost in Fig. 12(e). Fig. 12(f) shows the final result of the De-Ghosting, which is an exact reconstruction. The computation time was approximately 500 milliseconds on a single-core 1.6 GHz AMD Turion™ 64-bit Laptop. This algorithm performs several orders of magnitude

faster than the method of Chandra *et al.* [13], whose computations took in the order of hours to complete on the same machine.

Further work needs to be done to handle inconsistencies within projections while utilising De-Ghosting. Also, a study of the small variations within the rational angle multiplicity may be critical in understanding whether some projections are more “important” than others. The information in an image is known to be distributed non-uniformly amongst its discrete projections. The variance for a DRT projection  $m$ , with equivalent MT projection  $[b, a]$ , scales inversely as  $(a^2 + b^2)$ .

## CONCLUSION

A theory for missing slices of the Discrete FST was constructed that allowed description of Ghost artefacts in the 2D DFT. The theory was used to construct new Ghost convolution and deconvolution methods, having a computational complexity of  $O(n \log_2 n)$  with  $n = N^2$ , which can be utilised in recovering missing slices in the DFT (see Props 1 to 4). The methods required the use of the NTT in order to avoid numerical overflow and loss of precision problems. This De-Ghost method was then used to solve the discrete inverse problem of reconstructing exactly from highly asymmetric rational angle projections (see Figs 12 and 14).

## ACKNOWLEDGEMENTS

S. Chandra would like to thank Monash University for a Ph.D scholarship and a publications award. N. Normand would like to thank the Australian Research Council for his International Fellowship (ARC LX 0989907).

## APPENDIX GHOST ANGLE SETS

Chandra *et al.* [51] showed that any MT projection, acquired along a rational vector  $[b, a]$ , can be directly and efficiently mapped to a prime-sized DFT space exactly as

$$m \equiv ab^{-1} \pmod{N}, \quad (9)$$

where  $a, b \in \mathbb{Z}$  with  $\gcd(a, b) = 1$  and  $b^{-1}$  is the multiplicative inverse of  $b$ . The mapping (9) results from solving  $mb \equiv a \pmod{N}$ . Each Mojette bin or translate  $t_{\mathcal{M}}$  is then placed into a DRT translate  $t_{\mathcal{R}}$  as

$$t_{\mathcal{R}} = \begin{cases} b^{-1} t_{\mathcal{M}} \pmod{N}, & \text{if } \gcd(a, N) > 1 \\ a^{-1} t_{\mathcal{M}} \pmod{N}, & \text{if } \gcd(b, N) > 1 \end{cases}. \quad (10)$$

An efficient way to generate the values  $a$  and  $b$  is to minimise the  $L_1$ -norm of the rational vectors  $[b, a]$ . This can be computed as

$$b_3 = \left\lfloor \frac{b_1 + a_1 + N}{a_2} \right\rfloor b_2 - b_1, \quad a_3 = \left\lfloor \frac{b_1 + a_1 + N}{a_2} \right\rfloor a_2 - a_1, \quad (11)$$

where  $\lfloor \cdot \rfloor$  is the floor (round-down) operator, beginning the computation with  $[b_1, a_1] = [0, 1]$  and  $[b_2, a_2] = [1, N]$  until  $[b_3, a_3] = [1, 1]$  [51]. Positive  $[b, a]$  values represent the first octant of the half-plane with the other octants produced by  $[-b, a]$ ,  $[a, b]$  and  $[a, -b]$  vectors [16]. The resulting  $m$  values from these rational vectors can be compared with values



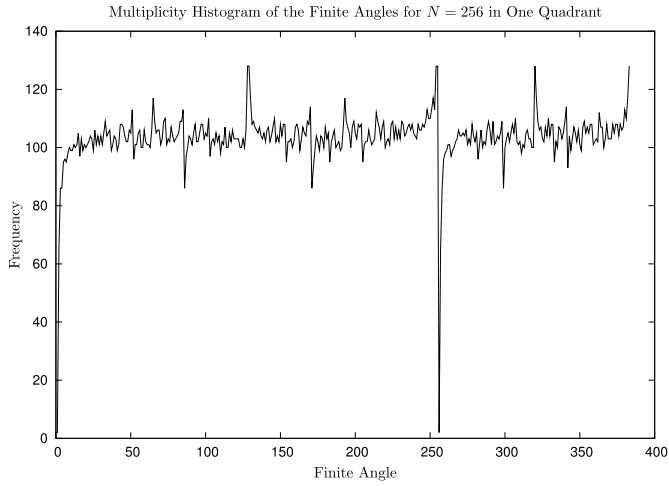


Figure 13. The multiplicity of the mapping between the finite angle and rational angle sets for  $N = 256$ , i.e. for the dyadic case. Because of the dyadic size, the range of values are  $0 \leq m < N$  and  $s' = N + s$ , where  $0 \leq s < N/2$ .

previously computed to obtain the desired DFT coverage. Eq. (11) produces rational angles that result in more compact projections in terms of the number of bins  $B$ , since

$$B = |a|(Q - 1) + |b|(P - 1) + 1. \quad (12)$$

Minimising  $L_1$ -norm of the rational angles allows one to acquire  $N + 1$  projections of certain coverage with reduced redundancy. The De-Ghost methods discussed in this work does not depend on how the rational angles are generated or selected.

This method can be extended to cover a fraction of the half-plane by limiting the number of octants used within the projection set. One then can obtain a set that sparsely covers the half-plane, but one that can still completely cover DFT space. This is because there exists a multiplicity of possible rational vectors for each finite angle  $[1, m]$  across the half-plane. The multiplicity is a consequence of the one-to-many nature of the  $m$  rational mapping in Eq. (9). A total of  $\min(P, Q) + 1$  projections can then be selected from this set that satisfies Katz criterion (4), while still being highly asymmetric. This multiplicity is shown as graphs in Fig. 13 for a single quadrant and the half plane.

The multiplicity of the  $m$  rational mapping appears to be relatively “flat” in a discrete sense, but with small variations. Graph 13 is reminiscent of curves obtained by Svalbe and Kingston [52] when observing the “unevenness” of the rational vectors from uniform coverage. The unevenness of the rational vectors is related to the distribution of prime numbers and the Riemann hypothesis [53]. Once  $\min(P, Q) + 1$  projections are chosen, the theory of cyclic Ghosts can be applied to reconstruct these projections exactly. It effectively reduces the number of projections required to  $\min(P, Q) + 1$ , rather than a total  $N + 1$  in the traditional case. Examples of each projection set are given in Fig. 14.

## REFERENCES

[1] A. Grigoryan, “New algorithms for calculating the discrete Fourier transforms,” *J. Vichislit. Matem. i Mat. Fiziki*, vol. 25, no. 9, pp. 1407–

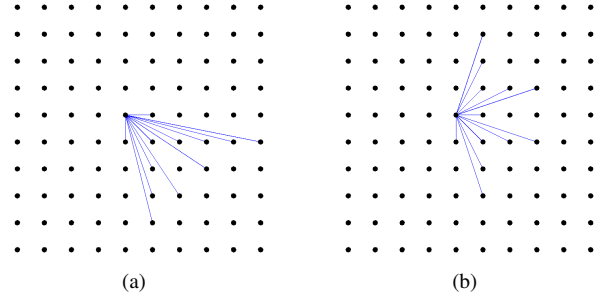


Figure 14. Examples of rational projection sets when utilising Ghosts in discrete reconstruction. (a) shows a set limited to a quadrant and (b) shows a set using the half-plane. The sets apply to a  $11 \times 11$  image embedded into a  $23 \times 23$  space.

- 1412, 1986.
- [2] E. D. Bolker, “The Finite Radon Transform,” *Contemporary Mathematics (American Mathematical Society)*, vol. 63, pp. 27–49, 1987.
- [3] I. Gertner, “A new efficient algorithm to compute the two-dimensional discrete Fourier transform,” *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 36, no. 7, pp. 1036–1050, Jul 1988.
- [4] J. A. Fill, “The Radon Transform on  $\mathbb{Z}_m$ ,” *SIAM Journal on Discrete Mathematics*, vol. 2, no. 2, pp. 262–283, 1989.
- [5] G. T. Herman and A. Kuba, *Discrete tomography: foundations, algorithms, and applications*. Birkhäuser, 1999.
- [6] F. Matúš and J. Flusser, “Image Representation via a Finite Radon Transform,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 15, no. 10, pp. 996–1006, 1993.
- [7] A. Kingston and I. Svalbe, “Generalised finite Radon transform for  $N \times N$  images,” *Image and Vision Computing*, vol. 25, no. 10, pp. 1620 – 1630, 2007, *Discrete Geometry for Computer Imagery 2005*.
- [8] T. Hsung, D. Lun, and W.-C. Siu, “The discrete periodic Radon transform,” *Signal Processing, IEEE Transactions on*, vol. 44, no. 10, pp. 2651–2657, 1996.
- [9] D. Lun, T. Hsung, and T. Shen, “Orthogonal discrete periodic Radon transform. Part I: theory and realization,” *Signal Processing*, vol. 83, no. 5, pp. 941–955, 2003.
- [10] —, “Orthogonal discrete periodic Radon transform. Part II: applications,” *Signal Processing*, vol. 83, no. 5, pp. 957–971, 2003.
- [11] A. Kingston, “Orthogonal discrete Radon transform over  $p^n \times p^n$  images,” *Signal Processing*, vol. 86, no. 8, pp. 2040 – 2050, 2006, special Section: *Advances in Signal Processing-assisted Cross-layer Designs*.
- [12] M. Katz, *Questions of Uniqueness and Resolution in Reconstruction from Projections*, ser. Lecture Notes in Biomathematics. Springer-Verlag, 1977.
- [13] S. S. Chandra, I. Svalbe, and J.-P. Guédon, “An exact, non-iterative Mojette inversion technique utilising ghosts,” in *Lecture Notes in Computer Science (LNCS)*. Springer Berlin / Heidelberg, 2008, vol. 4992, pp. 401–412.
- [14] S. S. Chandra and I. Svalbe, “A method for removing cyclic artefacts in discrete tomography using Latin squares,” *19th International Conference on Pattern Recognition*, pp. 1–4, Dec. 2008.
- [15] N. Normand, I. D. Svalbe, B. Parrein, and A. M. Kingston, “Erasure coding with the finite Radon transform,” in *IEEE Wireless Communications & Networking Conference*, Sydney, Apr. 2010.
- [16] I. Svalbe, “Exact, scaled image rotation using the Finite Radon Transform,” *Pattern Recognition Letters*, vol. 32, no. 9, pp. 1415–1420, 2011.
- [17] R. N. Bracewell and J. A. Roberts, “Aerial Smoothing in Radio Astronomy,” *Australian Journal of Physics*, vol. 7, pp. 615–640, Dec. 1954.
- [18] T. Cornwell, “Image Restoration and the Clean Technique,” in *Synthesis Mapping*, A. R. Thompson & L. R. D’Addario, Ed., 1982, p. 9.
- [19] A. K. Louis, “Ghosts in tomography - the null space of the Radon transform,” *Mathematical Methods in the Applied Sciences*, vol. 3, pp. 1–10, 1981.
- [20] B. F. Logan, “The uncertainty principle in reconstructing functions from projections,” *Duke Mathematical Journal*, vol. 42, no. 4, pp. 661–706, 1975.
- [21] G. N. Ramachandran and A. V. Lakshminarayanan, “Three-dimensional Reconstruction from Radiographs and Electron Micrographs: Application of Convolutions instead of Fourier Transforms,” *Proc. of the Nat. Acad.*

- of Sciences, vol. 68, no. 9, pp. 2236–2240, 1971.
- [22] L. Shepp and B. Logan, “The Fourier reconstruction of a head section,” *Nuclear Science, IEEE Transactions on*, vol. 21, pp. 21–43, 1974.
- [23] A. C. Kak and M. Slaney, *Principles of Computerized Tomographic Imaging*. Society of Industrial and Applied Mathematics, 2001.
- [24] A. K. Louis, “Nonuniqueness in inverse Radon problems: The frequency distribution of the ghosts,” *Mathematische Zeitschrift*, vol. 185, no. 3, pp. 429–440, 1984.
- [25] J.-P. Guédon, D. Barba, and N. Burger, “Psychovisual image coding via an exact discrete Radon transform,” *Proc. of the SPIE - The International Society for Optical Engineering*, vol. 2501, pp. 562–572, 1995.
- [26] N. Normand, A. Kingston, and P. Évenou, “A geometry driven reconstruction algorithm for the Mojette transform,” in *Lecture Notes in Computer Science (LNCS)*. Springer Berlin / Heidelberg, 2006, vol. 4245, pp. 122–133.
- [27] N. Normand, J.-P. Guédon, O. Philippe, and D. Barba, “Controlled redundancy for image coding and high-speed transmission,” *Proc. of the SPIE - The International Society for Optical Engineering*, vol. 2727, pp. 1070–1081, 1996.
- [28] J.-P. Guédon, N. Normand, A. Kingston, B. Parrein, M. Servières, P. Évenou, I. Svalbe, F. Atrousseau, T. Hamon, Y. Bizais, D. Coeurjolly, F. Boulos, and E. Graill, *The Mojette Transform: Theory and Applications*. ISTE-Wiley, 2009.
- [29] M. Servières, J. Idier, N. Normand, and J.-P. Guédon, “Conjugate gradient Mojette reconstruction,” *Proc. of the SPIE - The International Society for Optical Engineering*, vol. 5747, no. 1, pp. 2067–2074, 2005.
- [30] J. Boyd and J. Little, “Complementary data fusion for limited-angle tomography,” *Proc. 1994 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 288–294, 1994.
- [31] D. L. Donoho and P. B. Stark, “Uncertainty principles and signal recovery,” *SIAM Journal on Applied Mathematics*, vol. 49, p. 906, 1989.
- [32] D. L. Donoho and B. F. Logan, “Signal recovery and the large sieve,” *SIAM Journal on Applied Mathematics*, vol. 52, p. 577, 1992.
- [33] D. L. Donoho, “For most large underdetermined systems of linear equations the minimal  $\ell_1$ -norm solution is also the sparsest solution,” *Communications on Pure and Applied Mathematics*, vol. 59, no. 6, pp. 797–829, 2006.
- [34] J. Wright and Y. Ma, “Dense Error Correction Via  $\ell^1$ -Minimization,” *Information Theory, IEEE Transactions on*, vol. 56, no. 7, pp. 3540–3560, July 2010.
- [35] M. Vetterli, P. Marziliano, and T. Blu, “Sampling signals with finite rate of innovation,” *IEEE Transactions on Signal Processing*, vol. 50, no. 6, pp. 1417–1428, Jun. 2002.
- [36] A. Kuba, “The reconstruction of two-directionally connected binary patterns from their two orthogonal projections,” *Computer vision, graphics, and image processing*, vol. 27, no. 3, pp. 249–265, 1984.
- [37] S. Brunetti, P. Dulio, and C. Peri, “Characterization of -1, 0, +1 valued functions in discrete tomography under sets of 4 directions,” in *Lecture Notes in Computer Science (LNCS)*, vol. 6607, 2011, pp. 394–405.
- [38] E. Candès, J. Romberg, and T. Tao, “Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information,” *Information Theory, IEEE Transactions on*, vol. 52, no. 2, pp. 489–509, February 2006.
- [39] A. Ganesh, Z. Lin, J. Wright, L. Wu, M. Chen, and Y. Ma, “Fast algorithms for recovering a corrupted low-rank matrix,” in *Comp. Adv. in Multi-Sensor Adaptive Processing (CAMSAP), IEEE International Workshop on*, 2009, pp. 213–216.
- [40] G. Herman and R. Davidi, “Image reconstruction from a small number of projections,” *Inverse Problems*, vol. 24, p. 17, 2008.
- [41] P. J. Davis, *Circulant Matrices*. John Wiley & Sons, 1979.
- [42] S. S. Chandra and I. D. Svalbe, “Exact image representation via a Number-Theoretic Radon Transform,” *IET Computer Vision*, 2012, accepted.
- [43] S. S. Chandra, “The Finite Transform Library (FTL),” Monash University, Australia, Available at SourceForge.net, C/C++ Library (Open Source under GPL v3) 1.0, 2009. [Online]. Available: <http://finitetransform.sourceforge.net>
- [44] J. M. Pollard, “The Fast Fourier Transform in a Finite Field,” *Mathematics of Computation*, vol. 25, no. 114, pp. 365–374, 1971.
- [45] C. M. Rader, “Discrete Fourier transforms when the number of data samples is prime,” *Proceedings of the IEEE*, vol. 56, no. 6, pp. 1107–1108, June 1968.
- [46] H. J. Nussbaumer, “Overflow detection in the computation of convolutions by some number theoretic transforms,” *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 26, no. 1, pp. 108–109, Feb 1978.
- [47] S. S. Chandra, “Circulant Theory of the Radon Transform,” Ph.D. dissertation, School of Physics, Monash University, 2010, [Online Monash ARROW Repository](http://online.monash.edu.au/arrow/).
- [48] I. Svalbe, N. Normand, N. Nazareth, and S. S. Chandra, “On constructing minimal ghosts,” *Proceedings of the Digital Image Computing Techniques and Applications (DICTA)*, pp. 276 – 281, 2010.
- [49] I. Svalbe and N. Normand, “Properties of minimal ghosts,” in *Lecture Notes in Computer Science (LNCS)*. Springer Berlin / Heidelberg, 2010, vol. 6607, pp. 417–428.
- [50] I. Svalbe and S. S. Chandra, “Growth of discrete projection ghosts created by iteration,” in *Lecture Notes in Computer Science (LNCS)*, I. Debled-Rennesson, E. Domenjoud, B. Kerautret, and P. Even, Eds. Springer Berlin / Heidelberg, 2011, vol. 6607, pp. 406–416.
- [51] S. S. Chandra, N. Normand, A. Kingston, J.-P. Guédon, and I. Svalbe, “Fast Mojette Transform for Discrete Tomography,” *Journal of Mathematical Imaging and Vision*, 2010, submitted December (in Review), Available on [arXiv:1006.1965v1 \[physics.med-ph\]](http://arxiv.org/abs/1006.1965v1).
- [52] I. Svalbe and A. Kingston, “On correcting the unevenness of angle distributions arising from integer ratios lying in restricted portions of the Farey plane,” *Combinatorial Image Analysis. 10th International Workshop, IWCA 2004. Proceedings (LNCS)*, vol. 3322, pp. 110–121, 2004.
- [53] J. Franel, “Les suites de farey et le problème des nombres premiers,” *Göttinger Nachrichten*, pp. 191–201, 1924.



tomography.



an Invited Visiting Professor at the University of Nantes, and is currently an Associate Editor for Pattern Recognition Letters.



interests are centered around computed tomography and the associated Radon transform (both continuous and discrete forms).

**Shekhar S. Chandra** was born in Suva, Fiji Islands in 1980. He received a B.Sc. in computer science at the University of the South Pacific (Fiji), followed by a B.Sc. (Hons) in physics at Monash University (Australia) and a Ph.D in physics at Monash University (Australia) in 2010.

He is currently a post-doctoral fellow at the Australian e-Health Research Centre, CSIRO, where he is undertaking research in medical imaging. His current research interests include image/signal processing, number theory, biomedical imaging and discrete

**Imants D. Svalbe** was born in Ballarat, Australia, in 1952. He received the B.Sc. (Hons) degree in physics in 1974 and the Ph.D. degree in nuclear physics, in 1979, both from the University of Melbourne, Australia.

He is currently a Senior Lecturer in the School of Physics at the Clayton Campus of Monash University, Australia, where he teaches physics and medical imaging to radiography students. His principal research interests are focussed on discrete effects in digital image processing. He has spent several periods as

**Andrew M. Kingston** was born in Brisbane, Australia in 1978. He received a B.Sc. degree in 1999 and a B.Eng. (Hons.) degree in 2002 at Monash University. He was awarded a Ph.D in physics at Monash University in 2006.

He is currently a Research Fellow in the Department of Applied Mathematics in the Research School of Physics and Engineering at the Australian National University. Prior to that he held a Post-Doctoral position at the IRCCyN laboratory in Ecole polytechnique de l’université de Nantes. His research interests are centered around computed tomography and the associated Radon transform (both continuous and discrete forms).



**Jeanpierre Guédon** was born in Grosbreuil, France in 1962. He received the M.S degree and Ph.D degree from Ecole Centrale de Nantes and University of Nantes in 1986 and 1990 respectively, working on sampling in tomography with Yves Bizais. In 1991-92, he was a post-doc at CDRH FDA Rockville MD, USA with Kyle Myers and Bob Wagner. In 1994, he was assistant professor at Polytech Nantes then professor since 2002.

He developed the Mojette transform with several colleagues since 1994. His fields of interest concerns the theory around the Mojette transform and its numerous applications for storing and transmissions and for medical imaging. He invented the Mojette game in 2010.



**Nicolas Normand** was born in Nantes, France, in 1969. He received the Dipl.Ing. degree from INSA, Rennes, in 1992, the M.S. degree from the University of Nantes in 1993 and the Ph.D degree in computer science from the University of Nantes in 1997.

He has been holding a permanent position (Maître de Conférences) at the IRCCyN laboratory in Polytech Nantes, formerly IRESTE, since 1997. In 2009, he was awarded an International Fellowship grant ARC LX0989907 from the Australian Research Council and spent 12 months in the School of Physics, Monash University, Melbourne. His research fields of interest are discrete tomography, especially the Mojette transform and discrete Radon transform, and discrete distances.





Contents lists available at SciVerse ScienceDirect

Theoretical Computer Science

journal homepage: [www.elsevier.com/locate/tcs](http://www.elsevier.com/locate/tcs)

# Distance transform computation for digital distance functions

Robin Strand <sup>a,\*</sup>, Nicolas Normand <sup>b</sup>

<sup>a</sup> Centre for Image Analysis, Uppsala University, Box 337, SE-75105 Uppsala, Sweden

<sup>b</sup> LUNAM Université, Université de Nantes, IRCCyN UMR CNRS 6597, Polytech Nantes, La Chantrerie, 44306 Nantes Cedex 3, France

## ARTICLE INFO

### Article history:

Received 9 September 2011

Received in revised form 25 April 2012

Accepted 6 May 2012

Communicated by G. Ausiello

### Keywords:

Distance function

Distance transform

Weighted distances

Neighborhood sequences

## ABSTRACT

In image processing, the distance transform (DT), in which each object grid point is assigned the distance to the closest background grid point, is a powerful and often used tool. In this paper, distance functions defined as minimal cost-paths are used and a number of algorithms that can be used to compute the DT are presented. We give proofs of the correctness of the algorithms.

© 2012 Elsevier B.V. All rights reserved.

## 1. Introduction

An algorithm for computing distance transforms (DTs) using the basic city-block (horizontal and vertical steps are allowed) and chessboard (diagonal steps are allowed in conjunction with the horizontal and vertical steps) distance functions was presented in [1]. These distance functions are defined as shortest paths and the corresponding distance maps can be computed efficiently. Since these path-based distance functions are defined by the cost of discrete paths, we call them *digital distance functions*.

There are two commonly used generalizations of the city-block and chessboard distance functions, the *weighted distances* [2–4], and *distances based on neighborhood sequences* (ns-distances) [5–8]. The weighted distance is defined as the cost of a minimal cost-path and the ns-distance is defined as a shortest path in which the neighborhood that is allowed in each step is given by a neighborhood sequence. With weighted distances, a two-scan algorithm is sufficient for any point-lattice, see [2,9]. For ns-distances, three scans are needed for computing correct DTs on a square grid [10].

In this paper, we consider the *weighted ns-distance* [11–13] in which both weights and a neighborhood sequence are used to define the distance function. By using “optimal” parameters (weights and neighborhood sequence), the asymptotic shape of the discs with this distance function is a twelve-sided polygon, see [11]. The relative error is thus asymptotically  $(1/\cos(\pi/12) - 1)/((1/\cos(\pi/12) + 1)/2) \approx 3.5\%$  using only  $3 \times 3$  neighborhoods when computing the DT. In other words, we have a close to exact approximation of the Euclidean distance still using the path-based approach with connectivities corresponding to small neighborhoods. Some different algorithms for computing the distance transform using the weighted ns-distance functions are given in this paper.

The paper is organized as follows. First, some basic notions are given and the definition of weighted ns-distances is given. In Section 3, algorithms using an additional DT holding the *length* of the paths that define the distance values are presented. The notion of distance propagating path is introduced to prove that correct DTs are computed. In Section 4, a look-up table that holds the value that should be propagated in each direction is used to compute the DT. The third approach

\* Corresponding author. Tel.: +46 18 4713469.

E-mail addresses: [robin@cb.uu.se](mailto:robin@cb.uu.se) (R. Strand), [Nicolas.Normand@polytech.univ-nantes.fr](mailto:Nicolas.Normand@polytech.univ-nantes.fr) (N. Normand).



considered here works for metric distance functions with periodic neighborhood sequences. A large mask that holds all distance information corresponding to the first period of the neighborhood sequence is used.

## 2. Weighted distances based on neighborhood sequences

The distance function considered here is defined by a neighborhood sequence using two neighborhoods and two weights. The neighborhoods are defined as follows

$$\mathcal{N}_1 = \{(\pm 1, 0), (0, \pm 1)\} \quad \text{and} \quad \mathcal{N}_2 = \{(\pm 1, \pm 1)\}.$$

Two grid points  $\mathbf{p}_1, \mathbf{p}_2 \in \mathbb{Z}^2$  are strict  $r$ -neighbors,  $r \in \{1, 2\}$ , if  $\mathbf{p}_2 - \mathbf{p}_1 \in \mathcal{N}_r$ . Neighbors of higher order can also be defined, but in this paper, we will use only 1- and 2-neighbors. Let

$$\mathcal{N} = \mathcal{N}_1 \cup \mathcal{N}_2.$$

The points  $\mathbf{p}_1, \mathbf{p}_2$  are 2-neighbors (or *adjacent*) if  $\mathbf{p}_2 - \mathbf{p}_1 \in \mathcal{N}$ , i.e., if they are strict  $r$ -neighbors for some  $r$ . A ns  $B$  is a sequence  $B = (b(i))_{i=1}^{\infty}$ , where each  $b(i)$  denotes a neighborhood relation in  $\mathbb{Z}^2$ . If  $B$  is periodic, i.e., if for some finite, strictly positive  $l \in \mathbb{Z}_+$ ,  $b(i) = b(i + l)$  is valid for all  $i \in \mathbb{N}^*$ , then we write  $B = (b(1), b(2), \dots, b(l))$ .

The following notation is used for the number of 1:s and 2:s in the ns  $B$  up to position  $k$ .

$$\mathbf{1}_B^k = |\{i : b(i) = 1, 1 \leq i \leq k\}| \quad \text{and} \quad \mathbf{2}_B^k = |\{i : b(i) = 2, 1 \leq i \leq k\}|.$$

A path in a grid, denoted  $\mathcal{P}$ , is a sequence  $\mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_n$  of adjacent grid points. A path is a  $B$ -path of length  $\mathcal{L}(\mathcal{P}) = n$  if, for all  $i \in \{1, 2, \dots, n\}$ ,  $\mathbf{p}_{i-1}$  and  $\mathbf{p}_i$  are  $b(i)$ -neighbors. The number of 1-steps and strict 2-steps in a given path  $\mathcal{P}$  is denoted  $\mathbf{1}_{\mathcal{P}}$  and  $\mathbf{2}_{\mathcal{P}}$ , respectively.

**Definition 1.** Given the ns  $B$ , the ns-distance  $d(\mathbf{p}_0, \mathbf{p}_n; B)$  between the points  $\mathbf{p}_0$  and  $\mathbf{p}_n$  is the length of a shortest  $B$ -path between the points.

Let the real numbers  $\alpha$  and  $\beta$  (the *weights*) and a  $B$ -path  $\mathcal{P}$  of length  $n$ , where exactly  $l$  ( $l \leq n$ ) pairs of adjacent grid points in the path are strict 2-neighbors be given. The *cost* of the  $(\alpha, \beta)$ -weighted  $B$ -path  $\mathcal{P}$  is  $\mathcal{C}_{\alpha, \beta}(\mathcal{P}) = (n - l)\alpha + l\beta$ . The  $B$ -path  $\mathcal{P}$  between the points  $\mathbf{p}_0$  and  $\mathbf{p}_n$  is a  $(\alpha, \beta)$ -weighted minimal cost  $B$ -path between the points  $\mathbf{p}_0$  and  $\mathbf{p}_n$  if no other  $(\alpha, \beta)$ -weighted  $B$ -path between the points has lower cost than the  $(\alpha, \beta)$ -weighted  $B$ -path  $\mathcal{P}$ .

**Definition 2.** Given the ns  $B$  and the weights  $\alpha, \beta$ , the weighted ns-distance  $d_{\alpha, \beta}(\mathbf{p}_0, \mathbf{p}_n; B)$  is the cost of a  $(\alpha, \beta)$ -weighted minimal cost  $B$ -path between the points.

The following theorem is from [11].

**Theorem 1** (Weighted ns-distance in  $\mathbb{Z}^2$ ). Let the ns  $B$ , the weights  $\alpha, \beta$  s.t.  $0 < \alpha \leq \beta \leq 2\alpha$ , and the point  $(x, y) \in \mathbb{Z}^2$ , where  $x \geq y \geq 0$ , be given. The weighted ns-distance between  $\mathbf{0}$  and  $(x, y)$  is given by

$$d_{\alpha, \beta}(\mathbf{0}, (x, y); B) = (2k - x - y) \cdot \alpha + (x + y - k) \cdot \beta$$

where  $k = \min_l : l \geq \max(x, x + y - \mathbf{2}_B^l)$ .

Note if  $B = (1)$  then  $k = x + y$  so  $d(\mathbf{0}, (x, y); (1)) = (x + y)\alpha$  which is  $\alpha$  times the city-block distance whereas if  $B = (2)$  then  $k = x$  and  $d(\mathbf{0}, (x, y); (2)) = (x - y)\alpha + y\beta$  which is the  $(\alpha, \beta)$ -weighted distance.

## 3. Computing the distance transform using path-length information

In this section, the computation of DTs using the distance function defined in the previous section will be considered. Since the size of a digital image when stored in a computer is finite, we define the *image domain* as a finite subset of  $\mathbb{Z}^2$  denoted  $\mathcal{I}$ . In this paper we use image domains of the form

$$\mathcal{I} = [x_{\min}, x_{\max}] \times [y_{\min}, y_{\max}]. \tag{1}$$

**Definition 3.** We call the function  $F : \mathcal{I} \rightarrow \mathbb{R}_0^+$  an *image*.

Note that real numbers are allowed in the range of  $F$ . We denote the *object*  $X$  and the *background* is  $\bar{X} = \mathbb{Z}^2 \setminus X$ . We denote the distance transform for path-based distances with  $DT_{\mathcal{C}}$ , where the subscript  $\mathcal{C}$  indicates that costs of paths are computed.

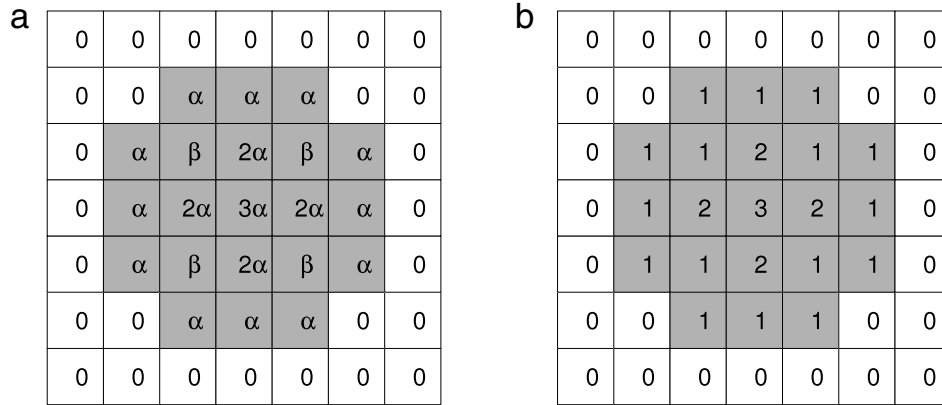


Fig. 1. Distance transforms for  $B = (2, 1)$  and  $\alpha \leq \beta \leq 2\alpha$ . The background is shown in white,  $DT_e$  is shown in (a) and a  $DT_L$  is shown in (b).

**Definition 4.** The distance transform  $DT_e$  of an object  $X \subset \mathcal{I}$  is the mapping

$$DT_e : \mathcal{I} \rightarrow \mathbb{R}_0^+ \quad \text{defined by}$$

$$\mathbf{p} \mapsto d(\mathbf{p}, \bar{X}), \quad \text{where}$$

$$d(\mathbf{p}, \bar{X}) = \min_{\mathbf{q} \in \bar{X}} \{d(\mathbf{p}, \mathbf{q})\}.$$

For weighted ns-distances, the size of the neighborhood allowed in each step is determined by the length of the minimal cost-paths (not the cost). In the first approach to compute the DT, an additional transform,  $DT_L$  that holds the length of the minimal cost path at each point is used.

**Definition 5.** The set of transforms  $\{DT_L^i\}$  of an object  $X \subset \mathbb{Z}^2$  is defined by all mappings  $DT_L^i$  that satisfy

$$DT_L^i(\mathbf{p}) = d_{1,1}(\mathbf{p}, \mathbf{q}; B), \quad \text{where}$$

$$\mathbf{q} \text{ is such that } d_{\alpha,\beta}(\mathbf{q}, \mathbf{p}; B) = d_{\alpha,\beta}(\mathbf{p}, \bar{X}; B).$$

See Fig. 1 for an example showing  $DT_e$  and some different  $DT_L$  (superscript omitted when it is not explicitly needed) of an object.

When  $\alpha = \beta = 1$ ,  $DT_L$  is uniquely defined and  $DT_e = DT_L$ . Example 1 illustrates that  $DT_L$  is not always uniquely defined when  $\alpha \neq \beta$ . We will see that despite this, the correct distance values are propagated by natural extensions of well-known algorithms when  $DT_e$  is used together with  $DT_L^i$  for any  $i$  are used to propagate the distance values.

We now introduce the notion of distance propagating path.

**Definition 6.** Given an object grid point  $\mathbf{p} \in X$ , a minimal cost  $B$ -path  $\mathcal{P}_{\mathbf{q},\mathbf{p}} = \langle \mathbf{q} = \mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_n = \mathbf{p} \rangle$ , where  $\mathbf{q} \in \bar{X}$  is a background grid point, is a distance propagating  $B$ -path if

- (i)  $\mathcal{C}_{\alpha,\beta}(\langle \mathbf{p}_0, \dots, \mathbf{p}_i \rangle) = DT_e(\mathbf{p}_i)$  for all  $i$  and
- (ii)  $\mathbf{p}_i, \mathbf{p}_{i+1}$  are  $b(DT_L^j(\mathbf{p}_i) + 1)$  – neighbors for all  $i$  and  $j$ .

If property (i) in the definition above is fulfilled, then we say that  $\mathcal{P}_{\mathbf{p},\mathbf{q}}$  is represented by  $DT_e$  and if property (ii) is fulfilled, then  $\mathcal{P}_{\mathbf{p},\mathbf{q}}$  is represented by  $DT_L^j$ . Note that when  $\alpha = \beta$ , then (i) implies (ii).

If we can guarantee that there is such a path for every object grid point, then the distance transform can be constructed by locally propagating distance information from  $\bar{X}$  to any  $\mathbf{p} \in X$ . Now, a number of definitions will be introduced. Using these definitions, we can show that there is always a distance propagating path when the weighted ns-distance function is used. The following definitions are illustrated in Examples 1 and 2.

**Definition 7.** Let  $\alpha, \beta$  such that  $0 < \alpha \leq \beta \leq 2\alpha$ , a ns  $B$ , an object  $X$ , and a point  $\mathbf{p} \in X$  be given. A minimal cost  $B$ -path  $\mathcal{P}_{\mathbf{q},\mathbf{p}}$ , where  $\mathbf{q} \in \bar{X}$ , such that  $d_{\alpha,\beta}(\mathbf{p}, \mathbf{q}; B) = d_{\alpha,\beta}(\mathbf{p}, \bar{X}; B)$  is a minimal cost  $B$ -path with minimal number of 2-steps if, for all paths  $\mathcal{Q}_{\mathbf{q}',\mathbf{p}}$  with  $\mathbf{q}' \in \bar{X}$  such that  $\mathcal{C}_{\alpha,\beta}(\mathcal{Q}_{\mathbf{q}',\mathbf{p}}) = \mathcal{C}_{\alpha,\beta}(\mathcal{P}_{\mathbf{q},\mathbf{p}})$ , we have

$$2_{\mathcal{P}_{\mathbf{q},\mathbf{p}}} \leq 2_{\mathcal{Q}_{\mathbf{q}',\mathbf{p}}}.$$

In other words, if there are several paths defining the distance at a point  $\mathbf{p}$ , the path with the least number of 2-steps is a minimal cost  $B$ -path with minimal number of 2-steps. See Examples 1 and 2.

**Remark 1.** A minimal cost-path with minimal number of 2-steps is a minimal cost-path of maximal length.

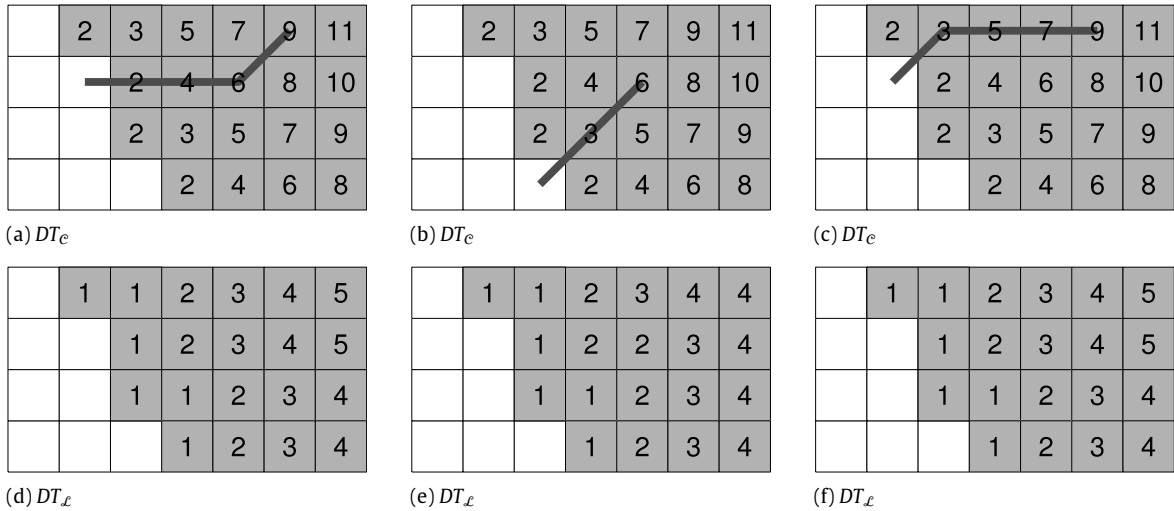


Fig. 2. Distance transform using  $(\alpha, \beta) = (2, 3)$  and  $B = (2, 2, 1)$  for a part of an object in  $\mathbb{Z}^2$ , see Example 1.

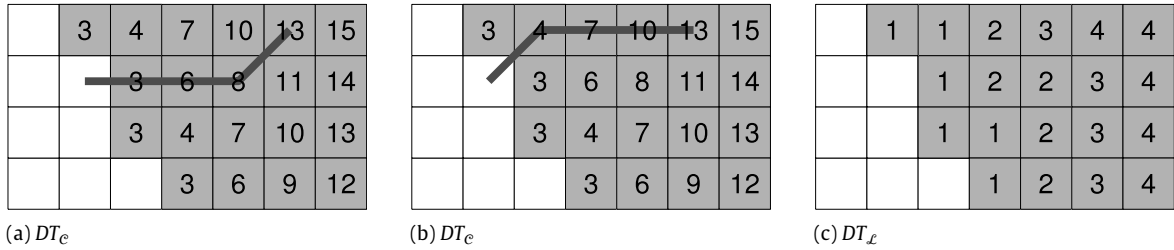


Fig. 3. Distance transform using  $(\alpha, \beta) = (3, 4)$  and  $B = (2, 2, 1)$  for a part of an object in  $\mathbb{Z}^2$ , see Example 2.

**Definition 8.** Let  $\alpha, \beta$  such that  $0 < \alpha \leq \beta \leq 2\alpha$ , a ns  $B$ , and points  $\mathbf{p}, \mathbf{q} \in \mathbb{Z}^2$  be given. The minimal cost  $(\alpha, \beta)$ -weighted  $B$ -path  $\mathcal{P}_{\mathbf{p}, \mathbf{q}} = \langle \mathbf{p} = \mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_n = \mathbf{q} \rangle$  is a *fastest* minimal cost  $(\alpha, \beta)$ -weighted  $B$ -path between  $\mathbf{p}$  and  $\mathbf{q}$  if there is an  $i, 0 \leq i \leq n$  such that

$$\mathbf{2}_{(\mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_i)} = \mathbf{2}_B^i \quad \text{and} \quad \mathbf{2}_{(\mathbf{p}_{i+1}, \mathbf{p}_{i+2}, \dots, \mathbf{p}_n)} = 0.$$

In other words, the minimal cost path between two points in which the 2-steps occur after as few steps as possible is a *fastest* minimal cost  $(\alpha, \beta)$ -weighted  $B$ -path. See Examples 1 and 2.

**Example 1.** This example illustrates that a path that is not a fastest path is not necessarily represented by  $DT_\ell^i$  for some  $i$ . Consider the (part of the) object showed in Fig. 2(a)–(f). The parameters  $(\alpha, \beta) = (2, 3)$  and  $B = (2, 2, 1)$  are used. In (d)–(f), some  $DT_\ell^i$ :s are shown. The path in (a) has a minimal number of 2-steps, but it is not a fastest path and is not represented by the  $DT_\ell$  in (e). The path in (b) does not have a minimal number of 2-steps. The path in (c) is a fastest path with minimal number of 2-steps and is also represented by all  $DT_\ell$  in (d)–(f). The paths shown in (b) and (c) are distance propagating paths, and the path in (a) is not a distance propagating path for  $DT_\ell$  in (e).

**Example 2.** In Fig. 3(a)–(c),  $B = (2, 2, 1)$  and  $(\alpha, \beta) = (3, 4)$  are used. In (a), the  $DT_e$  of an object and a minimal cost  $(\alpha, \beta)$ -weighted  $B$ -path with minimal number of 2-steps that is not distance propagating is shown. A distance propagating fastest minimal cost  $(\alpha, \beta)$ -weighted  $B$ -path with minimal number of 2-steps is shown in (b). The  $DT_\ell$  that corresponds to  $DT_e$  in (a)–(b) is shown in (c).

The following theorem says that a path satisfying Definitions 7 and 8 is a distance propagating path as defined in Definition 6. The theorem is proved in Lemma 2 and Lemma 3 below.

**Theorem 2.** If the  $B$ -path  $\mathcal{P}_{\mathbf{q}, \mathbf{p}} (\mathbf{p} \in X, \mathbf{q} \in \bar{X})$  such that  $d_{\alpha, \beta}(\mathbf{p}, \bar{X}) = C_{\alpha, \beta}(\mathcal{P}_{\mathbf{q}, \mathbf{p}})$  is a fastest minimal cost  $B$ -path with minimal number of 2-steps then  $\mathcal{P}_{\mathbf{q}, \mathbf{p}}$  is a distance propagating  $B$ -path.

Intuitively, we want the path to be of maximal length ( $B$ -path with minimal number of 2-steps, see Remark 1) and among the paths with this property, the path such that the 2-steps appear after as few steps as possible (*fastest  $B$ -path*). The algorithms we present will always be able to propagate correct distance values along such paths.

Lemma 1 will be used in the proofs of Lemmas 2 and 3. It is a direct consequence of Theorem 1. In Lemma 1,  $B(k) = (b(i))_{i=k}^\infty$ .

**Lemma 1.** Given  $\alpha, \beta$  such that  $0 < \alpha \leq \beta \leq 2\alpha$ , the ns  $B$ , the points  $\mathbf{p}, \mathbf{q}$ , and an integer  $k \geq 1$ , we have

$$d_{\alpha, \beta}(\mathbf{p}, \mathbf{q}; B) \leq d_{\alpha, \beta}(\mathbf{p}, \mathbf{q}; B(k)) + (2\alpha - \beta)(k - 1).$$



We consider the case  $\alpha < \beta$ . Lemmas 2 and 3 give the proof of Theorem 2 for weighted ns-distances.

**Lemma 2.** Let the weights  $\alpha, \beta$  such that  $0 < \alpha < \beta \leq 2\alpha$ , the ns  $B$ , and the point  $\mathbf{p} \in X$  be given. Any fastest minimal cost  $(\alpha, \beta)$ -weighted  $B$ -path  $\mathcal{P}_{\mathbf{q}, \mathbf{p}} = \langle \mathbf{q} = \mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_n = \mathbf{p} \rangle$  (for some  $\mathbf{q} \in \bar{X}$  such that  $d_{\alpha, \beta}(\mathbf{p}, \bar{X}) = C_{\alpha, \beta}(\mathcal{P}_{\mathbf{q}, \mathbf{p}})$ ) satisfies (i) in Definition 6, i.e.,

$$d_{\alpha, \beta}(\mathbf{p}_i, \bar{X}; B) = C_{\alpha, \beta}(\mathcal{P}_{\mathbf{p}_0, \mathbf{p}_i}) \quad \forall i : 0 \leq i \leq n. \quad (2)$$

**Proof.** First we note that there always exists a  $\mathbf{q} \in \bar{X}$  such that there is a fastest minimal cost  $(\alpha, \beta)$ -weighted  $B$ -path  $\mathcal{P}_{\mathbf{q}, \mathbf{p}} = \langle \mathbf{q} = \mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_n = \mathbf{p} \rangle$ . To prove (2), assume that there is a  $\mathbf{q}' \in \bar{X}$  and a path  $\mathcal{Q}_{\mathbf{q}', \mathbf{p}} = \langle \mathbf{q}' = \mathbf{p}'_0, \mathbf{p}'_1, \dots, \mathbf{p}'_k = \mathbf{p}_i, \mathbf{p}'_{k+1}, \dots, \mathbf{p}'_m = \mathbf{p} \rangle$  for some  $i$  such that

$$C_{\alpha, \beta}(\mathcal{P}_{\mathbf{q}, \mathbf{p}_i}) > C_{\alpha, \beta}(\mathcal{Q}_{\mathbf{q}', \mathbf{p}_i}).$$

Case i:  $\mathcal{L}(\mathcal{Q}_{\mathbf{q}', \mathbf{p}_i}) < \mathcal{L}(\mathcal{P}_{\mathbf{q}, \mathbf{p}_i})$ .

Case i (a):  $2_{\mathcal{Q}_{\mathbf{q}', \mathbf{p}_i}} > 2_{\mathcal{P}_{\mathbf{q}, \mathbf{p}_i}}$ .

Since  $\mathcal{P}_{\mathbf{q}, \mathbf{p}}$  is a fastest path,  $2_{\mathcal{P}_{\mathbf{p}_i, \mathbf{p}}} = 0$ . This implies that  $\mathcal{P}_{\mathbf{p}_i, \mathbf{p}}$  is a minimal cost  $(\alpha, \beta)$ -weighted  $B$ -path for  $B = (1)$  and since, by Theorem 1, any ns generates distances less than (or equal to)  $B = (1)$ ,  $C_{\alpha, \beta}(\mathcal{Q}_{\mathbf{p}_i, \mathbf{p}}) \leq C_{\alpha, \beta}(\mathcal{P}_{\mathbf{p}_i, \mathbf{p}})$ . Thus,  $C_{\alpha, \beta}(\mathcal{Q}_{\mathbf{q}', \mathbf{p}}) = C_{\alpha, \beta}(\mathcal{Q}_{\mathbf{q}', \mathbf{p}_i}) + C_{\alpha, \beta}(\mathcal{Q}_{\mathbf{p}_i, \mathbf{p}}) < C_{\alpha, \beta}(\mathcal{P}_{\mathbf{q}, \mathbf{p}_i}) + C_{\alpha, \beta}(\mathcal{P}_{\mathbf{p}_i, \mathbf{p}}) = C_{\alpha, \beta}(\mathcal{P}_{\mathbf{q}, \mathbf{p}})$ . This contradicts that  $\mathcal{P}_{\mathbf{q}, \mathbf{p}}$  is a minimal cost-path, so this case cannot occur.

Case i (b):  $2_{\mathcal{Q}_{\mathbf{q}', \mathbf{p}_i}} \leq 2_{\mathcal{P}_{\mathbf{q}, \mathbf{p}_i}}$

Since  $\mathcal{L}(\mathcal{Q}_{\mathbf{q}', \mathbf{p}_i}) = \mathcal{L}(\mathcal{P}_{\mathbf{q}, \mathbf{p}_i}) - L$  for some positive integer  $L$ , we have

$$1_{\mathcal{Q}_{\mathbf{q}', \mathbf{p}_i}} + 2_{\mathcal{Q}_{\mathbf{q}', \mathbf{p}_i}} = \mathcal{L}(\mathcal{Q}_{\mathbf{q}', \mathbf{p}_i}) = \mathcal{L}(\mathcal{P}_{\mathbf{q}, \mathbf{p}_i}) - L = 1_{\mathcal{P}_{\mathbf{q}, \mathbf{p}_i}} + 2_{\mathcal{P}_{\mathbf{q}, \mathbf{p}_i}} - L.$$

Using  $2_{\mathcal{Q}_{\mathbf{q}', \mathbf{p}_i}} \leq 2_{\mathcal{P}_{\mathbf{q}, \mathbf{p}_i}}$  and  $\alpha \leq \beta$ , we get

$$1_{\mathcal{Q}_{\mathbf{q}', \mathbf{p}_i}} \alpha + 2_{\mathcal{Q}_{\mathbf{q}', \mathbf{p}_i}} \beta \leq 1_{\mathcal{P}_{\mathbf{q}, \mathbf{p}_i}} \alpha + 2_{\mathcal{P}_{\mathbf{q}, \mathbf{p}_i}} \beta - L\alpha.$$

Thus,  $C_{\alpha, \beta}(\mathcal{Q}_{\mathbf{q}', \mathbf{p}_i}) \leq C_{\alpha, \beta}(\mathcal{P}_{\mathbf{q}, \mathbf{p}_i}) - L\alpha$ . By Lemma 1,  $C_{\alpha, \beta}(\mathcal{Q}_{\mathbf{p}_i, \mathbf{p}}) \leq C_{\alpha, \beta}(\mathcal{P}_{\mathbf{p}_i, \mathbf{p}}) + (2\alpha - \beta)L$ .

We use these results and get  $C_{\alpha, \beta}(\mathcal{Q}_{\mathbf{q}', \mathbf{p}}) = C_{\alpha, \beta}(\mathcal{Q}_{\mathbf{q}', \mathbf{p}_i}) + C_{\alpha, \beta}(\mathcal{Q}_{\mathbf{p}_i, \mathbf{p}}) \leq C_{\alpha, \beta}(\mathcal{P}_{\mathbf{q}, \mathbf{p}_i}) - L\alpha + C_{\alpha, \beta}(\mathcal{P}_{\mathbf{p}_i, \mathbf{p}}) + (2\alpha - \beta)L = C_{\alpha, \beta}(\mathcal{P}_{\mathbf{q}, \mathbf{p}_i) + C_{\alpha, \beta}(\mathcal{P}_{\mathbf{p}_i, \mathbf{p}}) + (\alpha - \beta)L < C_{\alpha, \beta}(\mathcal{P}_{\mathbf{q}, \mathbf{p}})$ . This contradicts that  $\mathcal{P}_{\mathbf{q}, \mathbf{p}}$  is a minimal cost-path.

Case ii:  $\mathcal{L}(\mathcal{Q}_{\mathbf{q}', \mathbf{p}_i}) \geq \mathcal{L}(\mathcal{P}_{\mathbf{q}, \mathbf{p}_i})$ .

Now,  $2_{\mathcal{Q}_{\mathbf{q}', \mathbf{p}_i}} < 2_{\mathcal{P}_{\mathbf{q}, \mathbf{p}_i}} \leq 2_B^i$ .

Construct the path (not a  $B$ -path)  $\mathcal{Q}'_{\mathbf{q}', \mathbf{p}} = \langle \mathbf{q}' = \mathbf{p}'_0, \mathbf{p}'_1, \dots, \mathbf{p}'_k = \mathbf{p}_i, \mathbf{p}_{i+1}, \dots, \mathbf{p}_n = \mathbf{p} \rangle$  of length  $n' = \mathcal{L}(\mathcal{Q}_{\mathbf{q}', \mathbf{p}_i}) + \mathcal{L}(\mathcal{P}_{\mathbf{p}_i, \mathbf{p}_n}) \geq \mathcal{L}(\mathcal{P}_{\mathbf{q}, \mathbf{p}_n}) = n$ . We have  $2_{\mathcal{Q}'_{\mathbf{q}', \mathbf{p}_n}} = 2_{\mathcal{Q}_{\mathbf{q}', \mathbf{p}_i}} + 2_{\mathcal{P}_{\mathbf{p}_i, \mathbf{p}_n}} < 2_B^i + 2_B^{n-i} = 2_B^n \leq 2_B^{n'}$ . This means that there is a  $B$ -path  $\mathcal{Q}''_{\mathbf{q}', \mathbf{p}_n}$  (obtained by permutation of the positions of the 1-steps and 2-steps in  $\mathcal{Q}'_{\mathbf{q}', \mathbf{p}_n}$ ) of length  $n'$  such that  $C_{\alpha, \beta}(\mathcal{Q}''_{\mathbf{q}', \mathbf{p}_n}) < C_{\alpha, \beta}(\mathcal{P}_{\mathbf{q}, \mathbf{p}_n})$ , which contradicts that  $\mathcal{P}_{\mathbf{q}, \mathbf{p}_n}$  is a minimal cost path.

The assumption is false, since a contradiction follows for all cases.  $\square$

Left to prove of Theorem 2 is that there is a path fulfilling the previous lemma that is represented by  $DT_{\mathcal{L}}$ . This is necessary for the path to be propagated correctly by an algorithm.

**Lemma 3.** Let the weights  $\alpha, \beta$  such that  $0 < \alpha < \beta \leq 2\alpha$ , the ns  $B$ , and the point  $\mathbf{p} \in X$  be given. Any fastest minimal cost  $(\alpha, \beta)$ -weighted  $B$ -path with minimal number of 2-steps  $\mathcal{P}_{\mathbf{q}, \mathbf{p}} = \langle \mathbf{q} = \mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_n = \mathbf{p} \rangle$  (for some  $\mathbf{q} \in \bar{X}$  such that  $d_{\alpha, \beta}(\mathbf{p}, \bar{X}) = C_{\alpha, \beta}(\mathcal{P}_{\mathbf{q}, \mathbf{p}})$ ) satisfies (ii) in Definition 6, i.e.,

$$\mathbf{p}_i, \mathbf{p}_{i+1} \text{ are } b(DT_{\mathcal{L}}(\mathbf{p}_i) + 1) - \text{neighbors for all } i.$$

**Proof.** Given a  $\mathbf{p} \in X$ , assume that there is a fastest minimal cost  $(\alpha, \beta)$ -weighted  $B$ -path  $\mathcal{P}_{\mathbf{q}, \mathbf{p}}$  with minimal number of 2-steps such that  $C_{\alpha, \beta}(\mathcal{P}_{\mathbf{q}, \mathbf{p}}) = d_{\alpha, \beta}(\mathbf{p}, \bar{X})$  and a  $K < n$  such that  $\mathcal{P}_{\mathbf{q}, \mathbf{p}_K}$  is not represented by  $DT_{\mathcal{L}}$ , i.e., that  $\mathbf{p}_{K-1}, \mathbf{p}_K$  are not  $b(DT_{\mathcal{L}}(\mathbf{p}_{K-1}) + 1)$ -neighbors. (Otherwise the value  $C_{\alpha, \beta}(\mathcal{P}_{\mathbf{q}, \mathbf{p}_K})$  is propagated from  $DT_{\mathcal{L}}(\mathbf{p}_{K-1})$ .) It follows that the values of  $DT_{\mathcal{L}}(\mathbf{p}_{K-1})$  and  $DT_{\mathcal{L}}(\mathbf{p}_K)$  are given by a path  $\mathcal{Q}_{\mathbf{q}', \mathbf{p}} = \langle \mathbf{q}' = \mathbf{p}'_0, \mathbf{p}'_1, \dots, \mathbf{p}'_k = \mathbf{p}_i, \mathbf{p}'_{k+1}, \dots, \mathbf{p}'_n = \mathbf{p} \rangle$  for some  $i$  and some  $\mathbf{q}' \in \bar{X}$  such that

$$C_{\alpha, \beta}(\mathcal{P}_{\mathbf{q}, \mathbf{p}_{K-1}}) = d_{\alpha, \beta}(\mathbf{p}_{K-1}, \mathbf{q}; B) = d_{\alpha, \beta}(\mathbf{p}_{K-1}, \mathbf{q}'; B) = C_{\alpha, \beta}(\mathcal{Q}_{\mathbf{q}', \mathbf{p}_{K-1}})$$

and

$$\mathcal{L}(\mathcal{P}_{\mathbf{q}, \mathbf{p}_{K-1}}) \neq \mathcal{L}(\mathcal{Q}_{\mathbf{q}', \mathbf{p}_{K-1}}).$$

We follow the cases in the proof of Lemma 2:

Case i:  $\mathcal{L}(\mathcal{Q}_{\mathbf{q}', \mathbf{p}_{K-1}}) < \mathcal{L}(\mathcal{P}_{\mathbf{q}, \mathbf{p}_{K-1}})$ .

Case i (a):  $2^{\mathcal{Q}_{\mathbf{q}', \mathbf{p}_{K-1}}} > 2^{\mathcal{P}_{\mathbf{q}, \mathbf{p}_{K-1}}}$ .

We get  $\mathcal{C}_{\alpha, \beta}(\mathcal{Q}_{\mathbf{q}', \mathbf{p}}) = \mathcal{C}_{\alpha, \beta}(\mathcal{P}_{\mathbf{q}, \mathbf{p}})$  from the proof of Lemma 2 and  $\mathcal{C}_{\alpha, \beta}(\mathcal{Q}_{\mathbf{q}', \mathbf{p}_{K-1}}) = \mathcal{C}_{\alpha, \beta}(\mathcal{P}_{\mathbf{q}, \mathbf{p}_{K-1}})$  by construction. Since  $\mathcal{P}_{\mathbf{q}, \mathbf{p}}$  is a fastest path,  $\mathbf{p}_{K-1}, \mathbf{p}_K$  is a 1-step, so it is also a  $b(DT_{\mathcal{L}}(\mathbf{p}_{K-1}) + 1)$ -step. Therefore,  $\mathbf{p}_{K-1}, \mathbf{p}_K$  are  $b(DT_{\mathcal{L}}(\mathbf{p}_{K-1}) + 1)$ -neighbors. Contradiction.

Case i (b):  $2^{\mathcal{Q}_{\mathbf{q}', \mathbf{p}_{K-1}}} \leq 2^{\mathcal{P}_{\mathbf{q}, \mathbf{p}_{K-1}}}$ .

Following the proof of Lemma 2, we get  $\mathcal{C}_{\alpha, \beta}(\mathcal{Q}_{\mathbf{q}', \mathbf{p}}) < \mathcal{C}_{\alpha, \beta}(\mathcal{P}_{\mathbf{q}, \mathbf{p}})$ .

Case ii:  $\mathcal{L}(\mathcal{Q}_{\mathbf{q}', \mathbf{p}_{K-1}}) > \mathcal{L}(\mathcal{P}_{\mathbf{q}, \mathbf{p}_{K-1}})$ .

This leads to a longer  $B$ -path  $\mathcal{Q}'_{\mathbf{q}', \mathbf{p}}$  with lower (or equal) cost by the construction in the proof of Lemma 2, so this case leads to a contradiction since  $\mathcal{P}_{\mathbf{q}, \mathbf{p}}$  is a  $B$ -path of maximal length (see Remark 1) by assumption.  $\square$

### 3.1. Algorithms

In this section, algorithms for computing DTs using the additional transform  $DT_{\mathcal{L}}$  are presented. First, we focus on a wavefront propagation algorithm. By Theorem 2, there is a distance propagating path for each  $\mathbf{p} \in X$ . This proves the correctness of Algorithm 1.

---

**Algorithm 1:** Computing  $DT_{\mathcal{C}}$  and  $DT_{\mathcal{L}}$  for weighted ns-distances by wave-front propagation.

---

**Input:**  $B, \alpha, \beta$ , neighborhoods  $\mathcal{N}_1$  and  $\mathcal{N}_2$ , and an object  $X \subset \mathbb{Z}^2$ .

**Output:** The distance transforms  $DT_{\mathcal{C}}$  and  $DT_{\mathcal{L}}$ .

**Initialization:** Set  $DT_{\mathcal{C}}(\mathbf{p}) \leftarrow 0$  for grid points  $\mathbf{p} \in \bar{X}$  and  $DT_{\mathcal{C}}(\mathbf{p}) \leftarrow \infty$  for grid points  $\mathbf{p} \in X$ . Set  $DT_{\mathcal{L}} = DT_{\mathcal{C}}$ . For all grid points  $\mathbf{p} \in \bar{X}$  adjacent to  $X$ : push  $(\mathbf{p}, DT_{\mathcal{C}}(\mathbf{p}))$  to the list  $L$  of ordered pairs sorted by increasing  $DT_{\mathcal{C}}(\mathbf{p})$ .

**Notation:**  $\omega_{\mathbf{v}}$  is  $\alpha$  if  $\mathbf{v} \in \mathcal{N}_1$  and  $\beta$  if  $\mathbf{v} \in \mathcal{N}_2$ .

```

while  $L$  is not empty do
  foreach  $\mathbf{p}$  in  $L$  with smallest  $DT_{\mathcal{C}}(\mathbf{p})$  do
    Pop  $(\mathbf{p}, DT_{\mathcal{C}}(\mathbf{p}))$  from  $L$ ;
    foreach  $\mathbf{q}: \mathbf{q}, \mathbf{p}$  are  $b(DT_{\mathcal{L}}(\mathbf{p}) + 1)$ -neighbors do
      if  $DT_{\mathcal{C}}(\mathbf{q}) > DT_{\mathcal{C}}(\mathbf{p}) + \omega_{\mathbf{p}-\mathbf{q}}$  then
         $DT_{\mathcal{C}}(\mathbf{q}) \leftarrow DT_{\mathcal{C}}(\mathbf{p}) + \omega_{\mathbf{p}-\mathbf{q}}$ ;
         $DT_{\mathcal{L}}(\mathbf{q}) \leftarrow DT_{\mathcal{L}}(\mathbf{p}) + 1$ ;
        Push  $(\mathbf{q}, DT_{\mathcal{C}}(\mathbf{q}))$  to  $L$ ;
      end
    end
  end
end
end
    
```

---

Now, the focus is on the raster-scanning algorithm. We will see that the DT can be computed correctly in three scans. Since a fixed number of scans is used and the time complexity is bounded by a constant for each visited grid point, the time complexity is linear in the number of grid points in the image domain.

We recall the following lemma from [14]:

**Lemma 4.** When  $\alpha < \beta \leq 2\alpha$ , any minimal cost-path between  $(0, 0)$  and  $(x, y)$ , where  $x \geq y \geq 0$ , consists only of the steps  $(1, 0)$ ,  $(1, 1)$ , and  $(0, 1)$ .

Since we consider only “rectangular” image domains, the following lemma holds.

**Lemma 5.** Given two points  $\mathbf{p}, \mathbf{q}$  in  $\mathcal{I}$ , a ns  $B$  and weights  $\beta > \alpha$ . Any point in any minimal cost  $(\alpha, \beta)$ -weighted  $B$ -path between  $\mathbf{p}$  and  $\mathbf{q}$  is in the image domain.

**Proof.** Consider the point  $\mathbf{p} = (x, y)$ , where  $x \geq y \geq 0$ . By Lemma 4, any  $(\alpha, \beta)$ -weighted  $B$ -path of minimal cost from  $\mathbf{0}$  to  $\mathbf{p}$  consists only of the local steps  $(0, 1)$ ,  $(1, 1)$ ,  $(1, 0)$ . The theorem follows from this result.  $\square$

Let  $\mathcal{N}^1 = \{(1, 0), (1, 1)\}$ ,  $\mathcal{N}^2 = \{(1, 1), (0, 1)\}, \dots, \mathcal{N}^8 = \{(1, -1), (1, 0)\}$ . In other words, the set  $\mathcal{N}$  is divided into set according to which octant they belong.

**Lemma 6.** Let  $\gamma$  be any permutation of  $1, 2, \dots, 8$ . Between any two points, there is a distance propagating  $B$ -path  $\mathcal{P}_{\mathbf{q}, \mathbf{p}} = \langle \mathbf{q} = \mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_n = \mathbf{p} \rangle$  and integers  $0 \leq K_1 \leq K_2 \leq \dots \leq K_8 = n$  such that

$$\begin{aligned} \mathbf{p}_i - \mathbf{p}_{i-1} &\in \mathcal{N}^{\gamma(1)} && \text{if } i \leq K_1 \\ \mathbf{p}_i - \mathbf{p}_{i-1} &\in \mathcal{N}^{\gamma(2)} && \text{if } i > K_1 \text{ and } i \leq K_2 \\ &\vdots && \\ \mathbf{p}_i - \mathbf{p}_{i-1} &\in \mathcal{N}^{\gamma(8)} && \text{if } i > K_7 \text{ and } i \leq K_8. \end{aligned}$$

**Proof.** Consider  $\mathbf{q} = \mathbf{0}$  and  $\mathbf{p} = (x, y)$  such that  $x \geq y \geq 0$ . Any minimal cost  $B$ -path consists only of the local steps  $(1, 0), (1, 1), (0, 1)$  by Lemma 4. Reordering the 1-steps does not affect the cost of the path. The path obtained by reordering the 1-steps in a fastest minimal cost  $(\alpha, \beta)$ -weighted  $B$ -path with minimal number of 2-steps is still a fastest minimal cost  $(\alpha, \beta)$ -weighted  $B$ -path with minimal number of 2-steps. Therefore, there are distance propagating  $B$ -path such that

$$\begin{aligned} \mathbf{p}_i - \mathbf{p}_{i-1} &\in \mathcal{N}^1 && \text{if } i \leq K_1 \\ \mathbf{p}_i - \mathbf{p}_{i-1} &\in \mathcal{N}^2 && \text{if } i > K_1 \text{ and } i \leq K_2 \end{aligned}$$

for some integers  $0 \leq K_1 \leq K_2 = n$  and

$$\begin{aligned} \mathbf{p}_i - \mathbf{p}_{i-1} &\in \mathcal{N}^2 && \text{if } i \leq K_1 \\ \mathbf{p}_i - \mathbf{p}_{i-1} &\in \mathcal{N}^1 && \text{if } i > K_1 \text{ and } i \leq K_2 \end{aligned}$$

for some integers  $0 \leq K_1 \leq K_2 = n$ . The general case follows from translation and rotation invariance.  $\square$

**Definition 9.** A scanning mask is a subset  $\mathcal{M} \subset \mathcal{N}$ .

**Definition 10.** A scanning order (so) is an enumeration of the  $M = \text{card}(\mathcal{J})$  distinct points in  $\mathcal{J}$ , denoted  $\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_M$ .

**Definition 11.** Let  $\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_M \in \mathcal{J}$  be a scanning order and  $\mathcal{M}$  a scanning mask. The scanning mask  $\mathcal{M}$  supports the scanning order if

$$\forall \mathbf{p}_i, \forall \mathbf{v} \in \mathcal{M}, ((\exists i' > i : \mathbf{p}_{i'} = \mathbf{p}_i + \mathbf{v}) \quad \text{or} \quad (\mathbf{p}_i + \mathbf{v} \notin \mathcal{J}_{\mathbb{C}})).$$

---

**Algorithm 2:** Computing  $DT_{\mathbb{C}}$  and  $DT_{\mathcal{L}}$  for weighted ns-distances by raster scanning.

---

**Input:**  $B, \alpha, \beta$ , scanning masks  $\mathcal{M}^i$ , and an object  $X \subset \mathbb{Z}^2$ .

**Output:** The distance transforms  $DT_{\mathbb{C}}$  and  $DT_{\mathcal{L}}$ .

**Initialization:** Set  $DT_{\mathbb{C}}(\mathbf{p}) \leftarrow 0$  for grid points  $\mathbf{p} \in \bar{X}$  and  $DT_{\mathbb{C}}(\mathbf{p}) \leftarrow \infty$  for grid points  $\mathbf{p} \in X$ . Set  $DT_{\mathcal{L}} = DT_{\mathbb{C}}$ .

**Comment:** The image domain  $\mathcal{J}$  defined by Eq. (1) is scanned  $L$  times using scanning orders such that the scanning mask  $\mathcal{M}^i$  supports the scanning order  $so_i, i \in \{1, \dots, L\}$ .

**Notation:**  $\omega_{\mathbf{v}}$  is  $\alpha$  if  $\mathbf{v} \in \mathcal{N}_1$  and  $\beta$  if  $\mathbf{v} \in \mathcal{N}_2$ .

```

for  $i = 1 : L$  do
  foreach  $\mathbf{p} \in \mathcal{J}$  following  $so_i$  do
    if  $DT_{\mathbb{C}}(\mathbf{p}) < \infty$  then
      foreach  $\mathbf{v} \in \mathcal{M}^i$  do
        if  $\mathbf{p}$  and  $\mathbf{p} + \mathbf{v}$  are  $b(DT_{\mathcal{L}}(\mathbf{p}) + 1)$ -neighbors then
          if  $DT_{\mathbb{C}}(\mathbf{p} + \mathbf{v}) > DT_{\mathbb{C}}(\mathbf{p}) + \omega_{\mathbf{v}}$  then
             $DT_{\mathbb{C}}(\mathbf{p} + \mathbf{v}) \leftarrow DT_{\mathbb{C}}(\mathbf{p}) + \omega_{\mathbf{v}};$ 
             $DT_{\mathcal{L}}(\mathbf{p} + \mathbf{v}) \leftarrow DT_{\mathcal{L}}(\mathbf{p}) + 1;$ 
          end
        end
      end
    end
  end
end

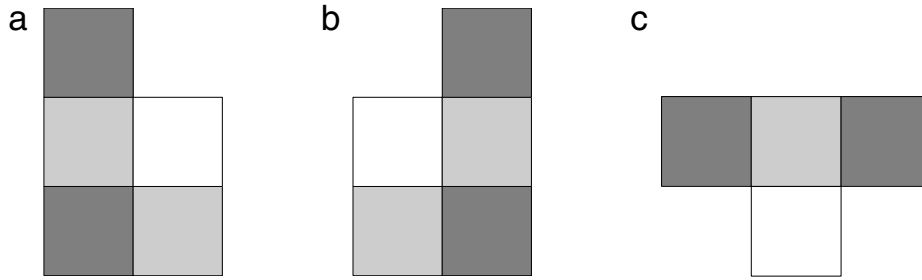
```

---

**Theorem 3.** If

- each of the sets  $\mathcal{N}^1, \mathcal{N}^2, \dots, \mathcal{N}^8$  is represented by at least one scanning mask and
- the scanning masks support the scanning orders,

then Algorithm 2 computes correct distance maps.



**Fig. 4.** Masks that can be used with Algorithm 2. The white pixel is the center of the mask and the two gray levels correspond to the elements in  $\mathcal{N}_1$  and  $\mathcal{N}_2$ , respectively.

**Proof.** Any distance propagating path between any pair of grid points in  $\mathcal{I}$  is also in  $\mathcal{I}$  by Lemma 5. Since the scanning masks support the scanning orders, there is, by Lemma 6, a distance propagating path that is propagated by the scanning masks.  $\square$

**Corollary 1.** Algorithm 2 with, e.g., the masks

$$\begin{aligned} \mathcal{M}^1 &= \{(-1, 1), (-1, 0), (-1, -1), (0, -1)\}, \\ \mathcal{M}^2 &= \{(0, -1), (1, -1), (1, 0), (1, 1)\}, \quad \text{and} \\ \mathcal{M}^3 &= \{(-1, 1), (0, 1), (1, 1)\}, \end{aligned}$$

see Fig. 4, gives correct distance transforms.

#### 4. Computing the distance transform using a look-up table

The look-up table  $LUT_{\mathbf{v}}(k)$  gives the value to be propagated in the direction  $\mathbf{v}$  from a grid point with distance value  $k$ . We will see that by using this approach, the additional distance transform  $DT_{\mathcal{L}}$  is *not* needed for computing  $DT_e$ . Thus, we get an efficient algorithm in this way. In this section, we assume that integer weights are used.

The LUT-based approach to compute the distance transform first appeared in [15]. The distance function considered in [15] uses neighborhood sequences, but is non-symmetric. The non-symmetry allows to compute the DT in one scan. The same LUT-based approach is used for binary mathematical morphology with convex structuring elements in [16]. This approach is efficient for, e.g., binary erosion in one scan with a computational per-pixel cost independent of the size of the structuring element.

For the algorithm in [15,16] the following formula is used in one raster scan:

$$DT_e(\mathbf{p}) = \min_{\mathbf{v} \in \mathcal{N}} (LUT_{\mathbf{v}}(DT_e(\mathbf{p} + \mathbf{v}))),$$

where  $\mathcal{N}$  is a non-symmetric neighborhood. In this section, we will extend this approach and allow the (symmetric) weighted ns-distances by allowing more than one scan.

##### 4.1. Construction of the look-up table

Given a distance value  $k$ , the look-up table at position  $k$  with subscript-vector  $\mathbf{v}$ ,  $LUT_{\mathbf{v}}(k)$ , holds information about the maximal distance value that can be found in a distance map in direction  $\mathbf{v}$ . See Examples 3 and 4.

**Example 3.** For a distance function on  $\mathbb{Z}^2$  defined by  $\alpha = 2, \beta = 3$  and  $B = (1, 2)$ , the LUT with  $D_{\max} = 10$  is the following:

	$j$	<u>0</u>	1	<u>2</u>	3	<u>4</u>	<u>5</u>	<u>6</u>	<u>7</u>	<u>8</u>	<u>9</u>	<u>10</u>
$\mathbf{v} \in \mathcal{N}_1$	$LUT_{\mathbf{v}}(j)$	2	3	4	5	6	7	8	9	10	11	12
$\mathbf{v} \in \mathcal{N}_2$	$LUT_{\mathbf{v}}(j)$	4	4	5	6	7	9	9	10	11	12	14

Only the values that are underlined are attained by the distance functions. See also Fig. 5. The values in the look-up tables can be extracted from these DTs by, for each distance value 0 to 10, finding the corresponding maximal value in the subscript-direction.

**Example 4.** For a distance function on  $\mathbb{Z}^2$  defined by  $\alpha = 4, \beta = 5$  and  $B = (1, 2, 1, 2, 2)$ , the LUT (only showing values that are attained by the distance function) with  $D_{\max} = 23$  is the following:

	$j$	0	4	8	9	12	13	16	17	18	20	21	22	23
$\mathbf{v} \in \mathcal{N}_1$	$LUT_{\mathbf{v}}(j)$	4	8	12	13	16	17	20	21	22	24	25	26	27
$\mathbf{v} \in \mathcal{N}_2$	$LUT_{\mathbf{v}}(j)$	8	9	13	17	17	18	21	22	23	25	26	27	31

The values in the LUT are given by the formula in Lemma 7.

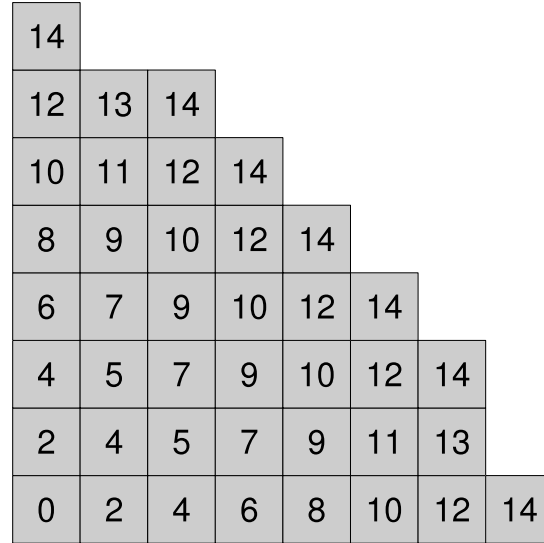


Fig. 5. Each pixel above is labeled with the distance to the pixel with value 0. The parameters  $B = (1, 2)$ ,  $(\alpha, \beta) = (2, 3)$  are used. See also Example 3.

**Lemma 7.** Let  $\alpha, \beta$  such that  $0 < \alpha \leq \beta \leq 2\alpha$ , the ns  $B$ , and the integer value  $k$  be given. Then

$$\left\{ \begin{array}{l} \max_{\mathbf{v} \in \mathcal{N}_1} \\ \mathbf{p}: d_{\alpha, \beta}(\mathbf{0}, \mathbf{p}; B) = k \end{array} \right\} (d_{\alpha, \beta}(\mathbf{0}, \mathbf{p} + \mathbf{v}; B) - k) = \alpha \text{ and}$$

$$\left\{ \begin{array}{l} \max_{\mathbf{v} \in \mathcal{N}_2} \\ \mathbf{p}: d_{\alpha, \beta}(\mathbf{0}, \mathbf{p}; B) = k \end{array} \right\} (d_{\alpha, \beta}(\mathbf{0}, \mathbf{p} + \mathbf{v}; B) - k) = \begin{cases} 2\alpha & \text{if } \exists n : b(n+1) = 1 \\ & \text{and } k = \mathbf{1}_B^n \alpha + \mathbf{2}_B^n \beta \\ \beta & \text{else.} \end{cases}$$

**Proof.** When  $\mathbf{v}$  is a 1-step, then the maximum difference between  $d_{\alpha, \beta}(\mathbf{0}, \mathbf{p} + \mathbf{v}; B)$  and  $d_{\alpha, \beta}(\mathbf{0}, \mathbf{p}; B)$  is  $\alpha$  by definition. There is a local step  $\mathbf{v} \in \mathcal{N}_1$  that increases the length of the minimal cost  $B$ -path (for any  $B$ ) by 1, so the maximum difference  $\alpha$  is always attained.

When  $\mathbf{v}$  is a strict 2-step,  $\mathbf{v} \in \mathcal{N}_2$  is the sum of two local steps from  $\mathcal{N}_1$ . Intuitively, if there are “enough” 2s in  $B$ , then the maximum difference is  $\beta$ . Otherwise, two 1-steps are used and the maximum difference is  $2\alpha$ . To prove this, let  $\mathcal{P}_{\mathbf{0}, \mathbf{p}} = \langle \mathbf{0} = \mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_n = \mathbf{p} \rangle$  be a minimal cost  $B$ -path and let  $\mathbf{p} = (x, y)$  be such that  $x \geq y \geq 0$ . We have the following conditions on  $B$ :

- (i)  $b(n+1) = 1$  and
- (ii)  $d_{\alpha, \beta}(\mathbf{0}, \mathbf{p}; B) = \mathbf{1}_B^n \alpha + \mathbf{2}_B^n \beta$ .

We note that (i) implies that  $\mathcal{P}_{\mathbf{0}, \mathbf{p}} \cdot \langle \mathbf{p} + \mathbf{w} \rangle$  is a  $B$ -path iff  $\mathbf{w} \in \mathcal{N}_1$  and a minimal cost  $B$ -path if  $\mathbf{w}$  is either  $(1, 0)$  or  $(0, 1)$ . Also, (ii) implies that the number of 2:s in  $B$  up to position  $n$  equals the number 2-steps in  $\mathcal{P}_{\mathbf{0}, \mathbf{p}}$ .

If both (i) and (ii) are fulfilled, since  $b(n+1) = 1$ , the 2-step  $\mathbf{v} = (1, 1)$  is divided into two 1-steps  $(1, 0)$  and  $(0, 1)$  giving a minimal cost  $B$ -path, so  $d_{\alpha, \beta}(\mathbf{0}, \mathbf{p} + \mathbf{v}; B) = d_{\alpha, \beta}(\mathbf{0}, \mathbf{p}; B) + 2\alpha$ .

If (i) is not fulfilled, then there is a 2-step  $\mathbf{v}$  such that  $\mathcal{P}_{\mathbf{0}, \mathbf{p}} \cdot \langle \mathbf{p} + \mathbf{v} \rangle$  is a minimal cost  $B$ -path of cost  $d_{\alpha, \beta}(\mathbf{0}, \mathbf{p} + \mathbf{v}; B) = d_{\alpha, \beta}(\mathbf{0}, \mathbf{p}; B) + \beta$ .

If (i), but not (ii) is fulfilled, then for any minimal cost  $B$ -path  $\mathcal{Q}_{\mathbf{0}, \mathbf{p}}$ , we have

$$k = \mathbf{1}_{\mathcal{Q}_{\mathbf{0}, \mathbf{p}}} \alpha + \mathbf{2}_{\mathcal{Q}_{\mathbf{0}, \mathbf{p}}} \beta \neq \mathbf{1}_B^{\mathcal{L}(\mathcal{Q}_{\mathbf{0}, \mathbf{p}})} \alpha + \mathbf{2}_B^{\mathcal{L}(\mathcal{Q}_{\mathbf{0}, \mathbf{p}})} \beta.$$

It follows that  $\mathbf{2}_{\mathcal{Q}_{\mathbf{0}, \mathbf{p}}} < \mathbf{2}_B^{\mathcal{L}(\mathcal{Q}_{\mathbf{0}, \mathbf{p}})}$ . Therefore, there is a 1-step in  $\mathcal{Q}_{\mathbf{0}, \mathbf{p}}$  that can be swapped with the 2-step  $\mathbf{v}$  giving a minimal cost  $B$ -path of cost  $d_{\alpha, \beta}(\mathbf{0}, \mathbf{p}; B) + \beta$ .  $\square$

The formula in Lemma 7 gives an efficient way to compute the look-up table, see Algorithm 3. The algorithm gives a correct LUT by Lemma 7. The output of Algorithm 3 for some parameters is shown in Examples 3 and 4.

Lemma 8 shows that the distance values are propagated correctly along distance propagating paths by using the look-up table.

**Algorithm 3:** Computing the look-up table for weighted ns-distances.

**Input:** Neighborhoods  $\mathcal{N}_1, \mathcal{N}_2$ , weights  $\alpha$  and  $\beta$  ( $0 < \alpha \leq \beta \leq 2\alpha$ ), a ns  $B$ , and the largest distance value  $D_{\max}$ .

**Output:** The look-up table  $LUT$ .

```

for  $k = 1 : D_{\max}$  do
  foreach  $\mathbf{v} \in \mathcal{N}_1$  do
     $LUT_{\mathbf{v}}(k) \leftarrow k + \alpha$ ;
  end
  foreach  $\mathbf{v} \in \mathcal{N}_2$  do
     $LUT_{\mathbf{v}}(k) \leftarrow k + \beta$ ;
  end
end
 $n \leftarrow 0$ ;
while  $\mathbf{1}_B^n \alpha + \mathbf{2}_B^n \beta \leq D_{\max}$  do
  if  $b(n+1) == 1$  then
     $LUT_{\mathbf{v}}(\mathbf{1}_B^n \alpha + \mathbf{2}_B^n \beta) \leftarrow (\mathbf{1}_B^n + 2) \alpha + \mathbf{2}_B^n \beta$ ;
  end
   $n \leftarrow n + 1$ ;
end

```

**Lemma 8.** Let  $\mathcal{P}_{\mathbf{p}_0, \mathbf{p}_n} = \langle \mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_n \rangle$  be a distance propagating  $B$ -path. Then

$$C_{\alpha, \beta}(\langle \mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_{i+1} \rangle) = LUT_{\mathbf{p}_{i+1} - \mathbf{p}_i}(C_{\alpha, \beta}(\langle \mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_i \rangle)) \forall i < n.$$

**Proof.** Assume that the lemma is false and let  $i$  be the minimal index such that

$$C_{\alpha, \beta}(\mathcal{P}_{\mathbf{p}_0, \mathbf{p}_{i+1}}) \neq LUT_{\mathbf{p}_{i+1} - \mathbf{p}_i}(C_{\alpha, \beta}(\mathcal{P}_{\mathbf{p}_0, \mathbf{p}_i})).$$

Then there is a path  $\mathcal{Q}_{\mathbf{q}_0, \mathbf{q}_j} = \langle \mathbf{q}_0, \mathbf{q}_1, \dots, \mathbf{q}_j \rangle$  such that

$$C_{\alpha, \beta}(\mathcal{P}_{\mathbf{p}_0, \mathbf{p}_i}) = C_{\alpha, \beta}(\mathcal{Q}_{\mathbf{q}_0, \mathbf{q}_j}) \quad \text{and} \quad \mathcal{L}(\mathcal{P}_{\mathbf{p}_0, \mathbf{p}_i}) \neq \mathcal{L}(\mathcal{Q}_{\mathbf{q}_0, \mathbf{q}_j})$$

defining the value in the LUT, i.e.,

$$C_{\alpha, \beta}(\mathcal{Q}_{\mathbf{q}_0, \mathbf{q}_j} \cdot \langle \mathbf{q}_j + \mathbf{v} \rangle) = LUT_{\mathbf{v}}(C_{\alpha, \beta}(\mathcal{P}_{\mathbf{p}_0, \mathbf{p}_i})),$$

where  $\mathbf{v} = \mathbf{p}_{i+1} - \mathbf{p}_i$ . Since the LUT stores the maximal local distances that are attained,

$$C_{\alpha, \beta}(\mathcal{Q}_{\mathbf{q}_0, \mathbf{q}_j} \cdot \langle \mathbf{q}_j + \mathbf{v} \rangle) > C_{\alpha, \beta}(\mathcal{P}_{\mathbf{p}_0, \mathbf{p}_{i+1}}).$$

It follows from Lemma 7 that  $\mathbf{v}$  is a strict 2-step and that  $LUT_{\mathbf{v}}(C_{\alpha, \beta}(\mathcal{P}_{\mathbf{p}_0, \mathbf{p}_i})) = 2\alpha$  and

$$C_{\alpha, \beta}(\mathcal{P}_{\mathbf{p}_0, \mathbf{p}_{i+1}}) - C_{\alpha, \beta}(\mathcal{P}_{\mathbf{p}_0, \mathbf{p}_i}) = \beta.$$

Since  $\mathcal{P}_{\mathbf{p}_0, \mathbf{p}_i}$  is a distance propagating  $B$ -path and  $\mathbf{v}$  is a strict 2-step,

$$\mathbf{2}_B^{\mathcal{L}(\mathcal{P}_{\mathbf{p}_0, \mathbf{p}_i})} = \mathbf{2}_{\mathcal{P}_{\mathbf{p}_0, \mathbf{p}_i}} \tag{3}$$

and  $\mathbf{2}_B^{\mathcal{L}(\mathcal{P}_{\mathbf{p}_0, \mathbf{p}_{i+1}})} = \mathbf{2}_{\mathcal{P}_{\mathbf{p}_0, \mathbf{p}_{i+1}}}$ .

case i  $\mathcal{L}(\mathcal{P}_{\mathbf{p}_0, \mathbf{p}_i}) > \mathcal{L}(\mathcal{Q}_{\mathbf{q}_0, \mathbf{q}_j})$ .

It follows that  $\mathbf{2}_{\mathcal{P}_{\mathbf{p}_0, \mathbf{p}_i}} < \mathbf{2}_{\mathcal{Q}_{\mathbf{q}_0, \mathbf{q}_j}} \leq \mathbf{2}_B^{\mathcal{L}(\mathcal{Q}_{\mathbf{q}_0, \mathbf{q}_j})} \leq \mathbf{2}_B^{\mathcal{L}(\mathcal{P}_{\mathbf{p}_0, \mathbf{p}_i})}$  which contradicts (3).

case ii  $\mathcal{L}(\mathcal{P}_{\mathbf{p}_0, \mathbf{p}_i}) < \mathcal{L}(\mathcal{Q}_{\mathbf{q}_0, \mathbf{q}_j})$ .

This implies that  $\mathbf{2}_{\mathcal{P}_{\mathbf{p}_0, \mathbf{p}_i}} > \mathbf{2}_{\mathcal{Q}_{\mathbf{q}_0, \mathbf{q}_j}}$ . Then  $\mathcal{Q}_{\mathbf{q}_0, \mathbf{q}_j} \cdot \langle \mathbf{q}_j + \mathbf{v} \rangle$  is not a distance propagating path (there are more elements 2 in  $B$  than 2-steps in the path). It follows from Lemma 7 that there is a distance propagating path from  $\mathbf{q}_0$  to  $\mathbf{q}_j + \mathbf{v}$  of cost  $C_{\alpha, \beta}(\mathcal{Q}_{\mathbf{q}_0, \mathbf{q}_j}) + \beta$ . Since  $\mathcal{Q}_{\mathbf{q}_0, \mathbf{q}_j}$  is arbitrary, it follows that  $LUT_{\mathbf{v}}(C_{\alpha, \beta}(\mathcal{P}_{\mathbf{p}_0, \mathbf{p}_i})) = LUT_{\mathbf{v}}(C_{\alpha, \beta}(\mathcal{Q}_{\mathbf{q}_0, \mathbf{q}_j})) = \beta$ . Contradiction.  $\square$

---

**Algorithm 4:** Computing  $DT_e$  for weighted ns-distances by wave-front propagation using a look-up table.

---

**Input:** LUT and an object  $X \subset \mathbb{Z}^2$ .

**Output:** The distance transform  $DT_e$ .

**Initialization:** Set  $DT_e(\mathbf{p}) \leftarrow 0$  for grid points  $\mathbf{p} \in \bar{X}$  and  $DT_e(\mathbf{p}) \leftarrow \infty$  for grid points  $\mathbf{p} \in X$ . For all grid points  $\mathbf{p} \in \bar{X}$  adjacent to  $X$ : push  $(\mathbf{p}, DT_e(\mathbf{p}))$  to the list  $L$  of ordered pairs sorted by increasing  $DT_e(\mathbf{p})$ .

```

while  $L$  is not empty do
  foreach  $\mathbf{p}$  in  $L$  with smallest  $DT_e(\mathbf{p})$  do
    Pop  $(\mathbf{p}, DT_e(\mathbf{p}))$  from  $L$ ;
    foreach  $\mathbf{v} \in \mathcal{N}$  do
      if  $DT_e(\mathbf{p} + \mathbf{v}) > LUT_{\mathbf{v}}(DT_e(\mathbf{p}))$  then
         $DT_e(\mathbf{p} + \mathbf{v}) \leftarrow LUT_{\mathbf{v}}(DT_e(\mathbf{p}))$ ;
        Push  $(\mathbf{p} + \mathbf{v}, DT_e(\mathbf{p} + \mathbf{v}))$  to  $L$ ;
      end
    end
  end
end

```

---

#### 4.2. Algorithms for computing the DT using look-up tables

In this section, we give algorithms that can be used to compute the distance transform using the LUT-approach. By Lemma 8, distance values are propagated correctly along distance propagating paths, so Algorithm 4 produces correct distance maps.

**Theorem 4.** *If*

- each of the sets  $\mathcal{N}^1, \mathcal{N}^2, \dots, \mathcal{N}^8$  is represented by at least one scanning mask and
- the scanning masks support the scanning orders,

then Algorithm 5 computes correct distance maps.

**Proof.** Since the same paths are propagated using this technique, the same conditions on the masks, scanning orders, and image domain are needed for Algorithm 5 to produce distance transforms without errors as when the additional distance transform  $DT_{\ell}$  is used.  $\square$

Note that Algorithms 4 and 5 derive from the work in [15,16], but here, symmetrical distance functions are allowed due to the increased number of scans.

---

**Algorithm 5:** Computing  $DT_e$  for weighted ns-distances by raster scanning using a look-up table.

---

**Input:** LUT, scanning masks  $\mathcal{M}^i$ , scanning orders  $so_i$  and an object  $X \subset \mathbb{Z}^2$ .

**Output:** The distance transform  $DT_e$ .

**Initialization:** Set  $DT_e(\mathbf{p}) \leftarrow 0$  for grid points  $\mathbf{p} \in \bar{X}$  and  $DT_e(\mathbf{p}) \leftarrow \infty$  for grid points  $\mathbf{p} \in X$ .

**Comment:** The image domain  $\mathcal{I}$  defined by Eq. (1) is scanned  $L$  times using scanning orders such that the scanning masks  $\mathcal{M}^i$  supports the scanning order  $so_i$ ,  $i \in \{1, \dots, L\}$

```

for  $i = 1 : L$  do
  foreach  $\mathbf{p} \in \mathcal{I}$  following  $so_i$  do
    if  $DT_e(\mathbf{p}) < \infty$  then
      foreach  $\mathbf{v} \in \mathcal{N}^i$  do
        if  $DT_e(\mathbf{p} + \mathbf{v}) > LUT_{\mathbf{v}}(DT_e(\mathbf{p}))$  then
           $DT_e(\mathbf{p} + \mathbf{v}) \leftarrow LUT_{\mathbf{v}}(DT_e(\mathbf{p}))$ ;
        end
      end
    end
  end
end

```

---

We remark that the computational cost of Algorithm 3 is linear with respect to the maximal radius  $D_{\max}$  and that the LUT can be computed on the fly when computing the DT. In other words, if it turns out during the DT computation that the LUT is



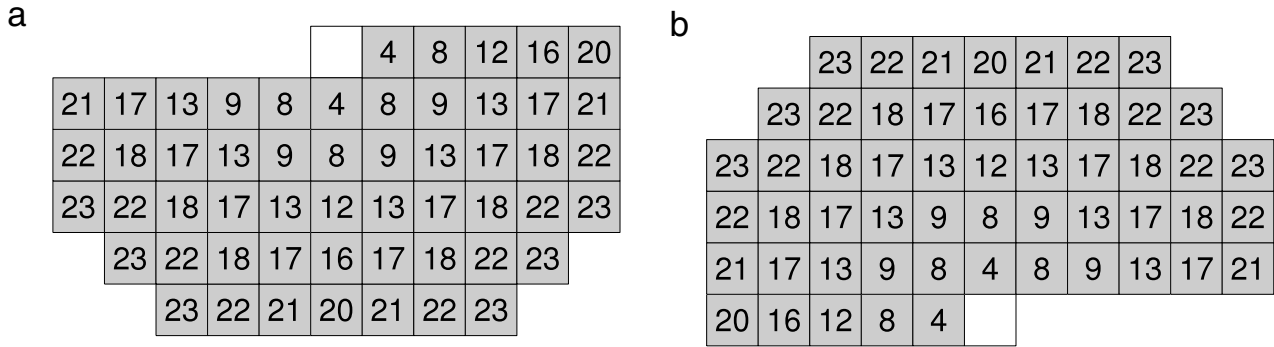


Fig. 6. Masks that can be used by a two-scan algorithm to compute a DT using the weighted ns-distance defined by  $B = (1, 2, 1, 2, 2)$ ,  $(\alpha, \beta) = (4, 5)$ .

too short, it can be extended by using Algorithm 3 with the modification that the loop variable starts from the missing value. Note also that for short neighborhood sequences, the LUT can sometimes be replaced by a modulo operator. For example, when  $(\alpha, \beta) = (4, 5)$  and  $B = (1, 2)$ , then by propagating distances to 2-neighbors only when  $DT_c(\mathbf{p})$  is not divisible by nine gives a very fast algorithm the computes a DT with low rotational dependency.

### 5. Computing the distance transform in two scans using a large mask

In [6], it is proved that if the weights and neighborhood sequence are such that the generated distance function is a metric, then the distance function is generated by constant neighborhood using a (large) neighborhood. This implies that the 2-scan chamfer algorithm can be used to compute the DT in  $\mathbb{Z}^2$ , see [12].

The following theorem is proved in [11]:

**Theorem 5.** *If*

$$\sum_{i=1}^N b(i) \leq \sum_{i=j}^{j+N-1} b(i) \quad \forall j, N \geq 1 \text{ and} \tag{4}$$

$$0 < \alpha \leq \beta \leq 2\alpha \tag{5}$$

then  $d_{\alpha, \beta}(\cdot, \cdot; B)$  is a metric.

In [11], the distance function generated by  $B = (1, 2, 1, 2, 2)$ ,  $(\alpha, \beta) = (4, 5)$  is suggested. In Fig. 6, the masks that can be used by a two-scan algorithm to compute the DT with this distance function are shown.

In this section we assume that the weights  $\alpha$  and  $\beta$  and the ns  $B$  are such that

- $B$  is periodic and
- the distance function generated by  $\alpha$ ,  $\beta$ , and  $B$  is a metric.

For this family of distance functions, a two-scan chamfer algorithm with large scanning masks can be used instead of the three-scan algorithm with small scanning masks using  $DT_{length}$  or the three-scan algorithm with small scanning masks using a LUT.

Let  $\overline{\mathcal{N}}$  be the set of grid points such that the distance value from  $\mathbf{0}$  is defined by the first period of  $B$ .

We now define two sets that are used in Algorithm 6.

$$\mathcal{M}^1 = \overline{\mathcal{N}} \cap \{(x, y) : y < 0 \text{ or } y = 0 \text{ and } x \geq 0\} \text{ and}$$

$$\mathcal{M}^2 = \overline{\mathcal{N}} \cap \{(x, y) : y > 0 \text{ or } y = 0 \text{ and } x \leq 0\}.$$

The following theorem is proved in [6]:

**Theorem 6.** *If  $d_{\alpha, \beta}(\cdot, \cdot; B)$  is a metric, then the weighted ns-distance defined by  $B$  and  $(\alpha, \beta)$  defines the same distance function as the weighted distance defined by the weighted vectors*

$$\left\{ \left( \mathbf{v}, d_{\alpha, \beta}(\mathbf{0}, \mathbf{0} + \mathbf{v}; B) \right) : \mathbf{v} \in \overline{\mathcal{N}} \right\}.$$

**Theorem 7.** *If the scanning masks  $\mathcal{M}^1$  and  $\mathcal{M}^2$  support the scanning orders then Algorithms 6 and 7 compute correct distance maps.*



---

**Algorithm 6:** Computing  $DT_e$  for weighted ns-distances by wave-front propagation using a large weighted mask.

---

**Input:** The mask  $\overline{\mathcal{N}}$ , and an object  $X \subset \mathbb{Z}^2$ .

**Output:** The distance transform  $DT_e$ .

**Initialization:** Set  $DT_e(\mathbf{p}) \leftarrow 0$  for grid points  $\mathbf{p} \in \overline{X}$  and  $DT_e(\mathbf{p}) \leftarrow \infty$  for grid points  $\mathbf{p} \in X$ . For all grid points  $\mathbf{p} \in \overline{X}$  adjacent to  $X$ : push  $(\mathbf{p}, DT_e(\mathbf{p}))$  to the list  $L$  of ordered pairs sorted by increasing  $DT_e(\mathbf{p})$ .

```

while  $L$  is not empty do
  foreach  $\mathbf{p}$  in  $L$  with smallest  $DT_e(\mathbf{p})$  do
    Pop  $(\mathbf{p}, DT_e(\mathbf{p}))$  from  $L$ ;
    foreach  $\mathbf{v} \in \overline{\mathcal{N}}$  do
      if  $DT_e(\mathbf{p} + \mathbf{v}) > DT_e(\mathbf{p}) + \omega_{\mathbf{v}}$  then
         $DT_e(\mathbf{p} + \mathbf{v}) \leftarrow DT_e(\mathbf{p}) + \omega_{\mathbf{v}}$ ;
        Push  $(\mathbf{p} + \mathbf{v}, DT_e(\mathbf{p} + \mathbf{v}))$  to  $L$ ;
      end
    end
  end
end

```

---

**Algorithm 7:** Computing  $DT_e$  for weighted ns-distances by two scans using a large weighted mask.

---

**Input:** Scanning masks  $\mathcal{M}^i$ , scanning orders  $so_i$ , weights, and an object  $X \subset \mathbb{Z}^2$ .

**Output:** The distance transform  $DT_e$ .

**Initialization:** Set  $DT_e(\mathbf{p}) \leftarrow 0$  for grid points  $\mathbf{p} \in \overline{X}$  and  $DT_e(\mathbf{p}) \leftarrow \infty$  for grid points  $\mathbf{p} \in X$ .

**Comment:** The image domain  $\mathcal{I}$  defined by Eq. (1) is scanned two times using scanning orders such that the scanning mask defined by  $\mathcal{M}^i$  supports the scanning order  $so_i$ ,  $i \in \{1, \dots, 2\}$

```

for  $i = 1 : 2$  do
  foreach  $\mathbf{p} \in \mathcal{I}$  following  $so_i$  do
    foreach  $\mathbf{v} \in \mathcal{M}^i$  do
      if  $DT_e(\mathbf{p} + \mathbf{v}) > DT_e(\mathbf{p}) + \omega_{\mathbf{v}}$  then
         $DT_e(\mathbf{p} + \mathbf{v}) \leftarrow DT_e(\mathbf{p}) + \omega_{\mathbf{v}}$ ;
      end
    end
  end
end

```

---

**Proof.** Any path consists of steps from  $\overline{\mathcal{N}}$  and the order of the steps is arbitrary. Consider the point  $\mathbf{p} = (x, y)$  such that  $x \geq y \geq 0$ . All points in any minimal cost path between  $\mathbf{0}$  and  $\mathbf{p}$  have non-negative coordinates. Also, all local steps are in  $\mathcal{M}^2$  except  $(1, 0)$ . Thus, the local steps in any minimal cost path between  $\mathbf{0}$  and  $\mathbf{p}$  can be rearranged such that the steps from  $\mathcal{M}^1$  are first and the steps from  $\mathcal{M}^2$  are last or vice versa. It follows that a minimal cost path is propagated from  $\mathbf{0}$  to each point  $\mathbf{p}$  such that  $x \geq y \geq 0$ . The theorem holds by translation and rotation invariance.  $\square$

## 6. Conclusions

We have examined the DT computation for weighted ns-distances. Three different, but related, algorithms have been presented and we have proved that the resulting DTs are correct.

We have shown that using the additional transform  $DT_{\mathcal{L}}$  is not needed for computing the DT  $DT_e$ . This extra information can, however, be useful when extracting medial representations, see [14].

We note that when the LUT-approach is used, a fast and efficient algorithm is obtained. This approach can also be used for computing the *constrained* DT. When the constrained DT is computed, there are obstacle grid points that are not allowed to intersect with the minimal cost paths that define the distance values. The path-based approach is well-suited for such algorithms. When the Euclidean distance is used, the corresponding algorithm must keep track of *visible* point, i.e., points which can be given the distance value by adding the length of the straight line segment between already visited points. Such algorithms, see [17], are slow and computationally heavy compared to the distance functions used in this paper.

For short sequences, the LUT can be replaced by a modulo function: consider  $B = (1, 2)$  and weights  $(\alpha, \beta)$ , then  $\beta$  is propagated to a two neighbor only from grid points with distance values that are divisible by  $\alpha + \beta$ . This approach gives a fast and efficient algorithm.

The LUT can be computed “on-the-fly” by using Algorithm 3. In other words, if it turns out during the DT computation that the LUT is too short, Algorithm 3 can be used to find the missing values in time that is proportional to the number of added values.

For long sequences, the two-scan Algorithms 6 and 7 is not efficient since the size of the masks depend on the length of the sequence. Also, this approach is valid only for metric distance functions.

Due to the low rotational dependency and the efficient algorithms presented here, we expect that the weighted ns-distance has the potential of being used in several image processing-applications where the DT is used: matching [18], morphology [19], and more recent applications such as separating arteries and veins in 3D pulmonary CT, [20] and traffic sign recognition, [21].

## References

- [1] A. Rosenfeld, J. L. Pfaltz, Sequential operations in digital picture processing, *Journal of the ACM* 13 (4) (1966) 471–494.
- [2] U. Montanari, A method for obtaining skeletons using a quasi-Euclidean distance, *Journal of the ACM* 15 (4) (1968) 600–624.
- [3] G. Borgefors, Distance transformations in digital images, *Computer Vision, Graphics, and Image Processing* 34 (1986) 344–371.
- [4] G. Borgefors, On digital distance transforms in three dimensions, *Computer Vision and Image Understanding* 64 (3) (1996) 368–376.
- [5] A. Rosenfeld, J. L. Pfaltz, Distance functions on digital pictures, *Pattern Recognition* 1 (1968) 33–61.
- [6] M. Yamashita, T. Ibaraki, Distances defined by neighbourhood sequences, *Pattern Recognition* 19 (3) (1986) 237–246.
- [7] B. Nagy, Metric and non-metric distances on  $\mathbb{Z}^n$  by generalized neighbourhood sequences, in: *Proceedings of 4th International Symposium on Image and Signal Processing and Analysis, ISPA 2005, Zagreb, Croatia, 2005*, pp. 215–220.
- [8] B. Nagy, Distances with neighbourhood sequences in cubic and triangular grids, *Pattern Recognition Letters* 28 (1) (2007) 99–109.
- [9] C. Fouard, R. Strand, G. Borgefors, Weighted distance transforms generalized to modules and their computation on point lattices, *Pattern Recognition* 40 (9) (2007) 2453–2474.
- [10] R. Strand, B. Nagy, C. Fouard, G. Borgefors, Generating distance maps with neighbourhood sequences, in: *Proceedings of 13th International Conference on Discrete Geometry for Computer Imagery, DGCI 2006, Szeged, Hungary*, in: *Lecture Notes in Computer Science*, vol. 4245, Springer-Verlag, 2006, pp. 295–307.
- [11] R. Strand, Weighted distances based on neighbourhood sequences, *Pattern Recognition Letters* 28 (15) (2007) 2029–2036.
- [12] A. Hajdu, L. Hajdu, R. Tijdeman, General neighborhood sequences in  $\mathbb{Z}^n$ , *Discrete Applied Mathematics* 155 (2007) 2507–2522.
- [13] R. Strand, B. Nagy, G. Borgefors, Digital distance functions on three-dimensional grids, *Theoretical Computer Science* 412 (15) (2011) 1350–1363. *theoretical Computer Science Issues in Image Analysis and Processing*.
- [14] R. Strand, Shape representation with maximal path-points for path-based distances, in: *Proceedings of 5th International Symposium on Image and Signal Processing and Analysis, ISPA 2007, Istanbul, Turkey, 2007*, pp. 397–402.
- [15] N. Normand, C. Viard-Gaudin, A background based adaptive page segmentation algorithm, in: *Proceedings of 3rd International Conference on Document Analysis and Recognition, ICDAR 1995, Washington, DC, USA, Vol. 1, 1995*, pp. 138–141.
- [16] N. Normand, Convex structuring element decomposition for single scan binary mathematical morphology, in: *Proceedings of 11th International Conference on Discrete Geometry for Computer Imagery, DGCI 2003, Naples, Italy*, in: *Lecture Notes in Computer Science*, Vol. 2886, Springer-Verlag, 2003, pp. 154–163.
- [17] D. Coeurjolly, S. Miguet, L. Tougne, 2D and 3D visibility in discrete geometry: an application to discrete geodesic paths, *Pattern Recognition Letters* 25 (5) (2004) 561–570.
- [18] G. Borgefors, Hierarchical chamfer matching: A parametric edge matching algorithm, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 10 (6) (1988) 849–865.
- [19] P. Nacken, Chamfer metrics in mathematical morphology, *Journal of Mathematical Imaging and Vision* 4 (1994) 233–253.
- [20] P. Saha, Z. Gao, S. Alford, M. Sonka, E. Hoffman, Topomorphologic separation of fused isointensity objects via multiscale opening: Separating arteries and veins in 3-D pulmonary CT, *IEEE Transactions on Medical Imaging* 29 (3) (2010) 840–851.
- [21] A. Ruta, Y. Li, X. Liu, Real-time traffic sign recognition from video by class-specific discriminative features, *Pattern Recognition* 43 (1) (2010) 416–430.

# Fast recursive grayscale morphology operators: from the algorithm to the pipeline architecture

Olivier Déforges · Nicolas Normand ·  
Marie Babel

Received: 13 July 2009 / Accepted: 12 July 2010  
© Springer-Verlag 2010

**Abstract** This paper presents a new algorithm for efficient computation of morphological operations for gray images and the specific hardware. The method is based on a new recursive morphological decomposition method of 8-convex structuring elements by only causal two-pixel structuring elements (2PSE). Whatever the element size, erosion or/and dilation can then be performed during a unique raster-like image scan involving a fixed reduced analysis neighborhood. The resulting process offers low computation complexity combined with easy description of the element form. The dedicated hardware is generic and fully regular, built from elementary interconnected stages. It has been synthesized into an FPGA and achieves high-frequency performances for any shape and size of structuring element.

**Keywords** Mathematical morphology ·  
8-Convex structuring element operators ·  
Regular dedicated pipeline architecture

## 1 Introduction

Mathematical morphology was first introduced as a method to measure binary objects, but soon became a complete theory based on set operations [14]. Morphology relies on the use of set operators (intersection, union, inclusion, complement) to transform an image. The transformed image usually has fewer details, but its main characteristics are still present. Once the image has been simplified, measurements can be computed to give a quantitative analysis of the image. Morphological transformation is based on a structuring element ( $B$ ) characterized by its shape, size and center location, also called origin. Each pixel in an image is compared with  $B$  by moving  $B$  so that its center hits the pixel. Depending on the type of morphological transformation, the pixel value is set to the minimal or maximal value of the pixels in the translated structuring element.

Performing a morphological operation directly from its formal definition would imply that all the neighbors present within the structuring element should be sought for each pixel. Of course, this process becomes time consuming for large elements. Fortunately, mathematical morphology properties led to the introduction of methods which partially solve this problem.

The main goal of the decomposition algorithms is to reduce the number of basic dilations needed to generate the desired structuring element. Different ways to decompose structuring elements have been investigated exploiting two fundamental properties: dilations [2, 19] and set unions of elements [9, 18]. More particularly, Chen has defined a decomposition method using only two non-connected pixel elements. On the other hand, Xu demonstrated how any 8-convex polygon could be constructed from a set of elements included in a 4-neighborhood.

---

O. Déforges (✉) · M. Babel  
UMR CNRS 6164 IETR, INSA Rennes, 20 av.  
des Buttes de Coesmes, 35043 Rennes, France  
e-mail: odeforge@insa-rennes.fr

M. Babel  
e-mail: mabel@insa-rennes.fr

N. Normand  
UMR CNRS 6597 IRCCYN, Ecole Polytechnique Nantes,  
La Chantrerie, BP 60601, 44306 Nantes, France  
e-mail: nnormand@polytech.univ-nantes.fr

Mathematical morphology is a tool commonly used in embedded vision systems, so a great deal of work has been done to design hardware implementation. The quality of an architecture dedicated to mathematical morphology is generally measured by its ability to describe different element shapes versus hardware efficiency, e.g., memory and computation requirements, or number of image scans. Of course, decomposition element methods can help the design of efficient dedicated hardware by limiting the number of operations.

Chien et al. [3] proposed a comparative table among existing architectures for a dilation by a  $7 \times 7$  structuring element (see Table in one pass 1). Chien architecture relies on the principle of PRR (Partial-Result-Reuse), trying to avoid redundant computations. This concept has been used earlier by Pitas Ong and Sunwoo [11] and [12], but their architectures were not fully optimized. Coltuc and Pitas [5] proposed an optimal solution but only for rectangular elements, and Chien architecture is an extension of some other shapes. A comparison between PRR and the proposed method will be discussed later.

Diamantaras and Kung [6] architecture provides the best flexibility, supporting both non-flat and arbitrarily shaped elements. However, it is at the expense of complexity. The architecture is built as a systolic array, and one Processor Element is defined for each pixel of the element. Ruetz and Brodersen [13] and Sheu et al. [15] both proposed pipelined implementations deduced from decomposition element techniques that are also limited to flat rectangular elements. The performance and capability of their approach will be further discussed in the Sect. 4.

Other works concern computational efficiency for grayscale morphological operators. Van Herk [17] proposed a very efficient method called HGW for grayscale erosions and dilations, requiring only three comparisons for 1D structuring elements, whatever the element size. It relies on a recursive forward and backward ranking of pixel values inside temporary buffers. Extension to 2D rectangular structuring elements is achieved by consecutively applying the same process to rows and columns, for a total of six comparisons. 2D circular symmetric filtering can also be considered by running the process along the two diagonal directions, doubling the number of operations. Soille et al. [16] introduced an extension of this algorithm to any angle 1D structuring elements. Gil and Kimmel [8] further improved the initial algorithm to reach 1.5 comparisons for 1D erosions and dilations, thanks to a sliding window technique. The major drawback is that this algorithm induces non-regular data flow processing, as well as the generation of ordered lists, which are not compatible with efficient implementations. In Clienti et al. [4], the authors proposed a modification of the original HGW, with

a block mirroring scheme, in order to design a low-memory requirements architecture. The other drawback is that all HGW-based techniques have been designed and optimized in 1D space: 2D extension requires a second pass and is mainly restricted to rectangular elements shapes. The technique proposed in this paper performs one-pass erosion or dilation, considering more generalized 2D convex shapes.

In this paper, we address the problem of grayscale morphology with a restricted class of discrete convex polygons (2). We then introduce a fast recursive technique for grayscale dilations, relying on this particular decomposition scheme. The process regularity also enables the design of a pipeline architecture (Sect. 3). We summarize the results in Sect. 4.

## 2 Fast recursive morphological operators

### 2.1 Binary morphology

Applying morphological operations on a set of points involves basic templates called structuring elements, also defined as sets of points, used as shape parameters for the operations.

The two most fundamental morphological operations are the dual dilation and erosion. The erosion (resp. dilation) of the set of points  $A$  with the structuring element  $B$ , denoted by  $A \ominus B$  (resp.  $A \oplus B$ ) is defined as (following notations of Chen and Haralick):

$$\begin{aligned}
 A \ominus B &= \{p \mid (B)_p \subset A\} = \bigcap_{p \in B} (A)_p \\
 A \oplus B &= \{p \mid (B)_p \cap A \neq \emptyset\} = \bigcup_{p \in B} (A)_p
 \end{aligned}
 \tag{1}$$

where  $(A)_p$  is equal to the set  $B$  translated by  $p$ :

$$(A)_p = \{a + p, a \in A\}.$$

A set of integer points  $A$  can be conveniently described as a binary image (i.e. its indicator function):

$$\begin{aligned}
 \mathbf{1}_A : Z &\rightarrow \{0, 1\} \\
 x &\rightarrow \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{if } x \notin A \end{cases}
 \end{aligned}$$

The erosion and dilation by the structuring element  $B$  can be rewritten using the min and max functions as

$$\begin{aligned}
 A \ominus B &= \min\{(A)_p \mid p \in B\} = \min_{p \in B} (A)_p, \\
 A \oplus B &= \max\{(A)_p \mid p \in B\} = \max_{p \in B} (A)_p.
 \end{aligned}
 \tag{2}$$

where the functional notation is deliberately dropped for convenience.

## 2.2 Grayscale morphology

Grayscale morphology is a natural extension of binary morphology, using the Umbra and Top operators. In most of the cases, grayscale morphology is generally restricted to flat structuring elements leading to the same simple dilation and erosion definitions based on *min* and *max* functions.

A grayscale image  $A$  maps each point in a subset of the two-dimensional digitized space  $Z \times Z$  to a intensity value. Some useful properties that hold both for binary morphology and grayscale morphology with flat structuring elements must be presented to justify the algorithm proposed later.

### 2.2.1 Translation

The dilation by the translated of a structuring element can be easier to compute than the dilation by the element itself. The following properties demonstrate the equivalence of operations (except for the translation):

$$\begin{aligned} A \ominus (B)_p &= (A)_p \ominus B = (A \ominus B)_p \\ A \oplus (B)_p &= (A)_p \oplus B = (A \oplus B)_p \end{aligned} \tag{3}$$

### 2.2.2 Associativity

Dilations (respectively erosions) by a series of structuring elements are equivalent to a dilation (resp. erosion) by the dilated of all the elements of the series:

$$\begin{aligned} (A \ominus B) \ominus C &= A \ominus (B \oplus C) \\ (A \oplus B) \oplus C &= A \oplus (B \ominus C) \end{aligned} \tag{4}$$

This property justifies a lot of algorithms since, as long as a structuring element can be decomposed by dilations of smaller elements, it is generally advantageous from a computational point of view to dilate or erode the image successively by all these elements. This property is often used conjointly with the previous one. Hence, dilating by a series of translated elements is equivalent to the combination of one global translation and dilations by the elements themselves. The number of operations per pixel can then be drastically reduced, but the major drawback is that the image has to be scanned several times.

### 2.2.3 Union of structuring elements

The other way to split a structuring element is to realize a set union of smaller elements. The dilation (resp. erosion) by a set union of structuring elements is then equal to the union of the dilations (resp. intersection of the erosions) by all the structuring elements:

$$\begin{aligned} A \ominus (B \cup C) &= (A \ominus B) \cap (A \ominus C) \\ A \oplus (B \cup C) &= (A \oplus B) \cup (A \oplus C) \end{aligned} \tag{5}$$

### 2.2.4 Opening/closing operators

Erosion and dilation operations are frequently combined to provide powerful filters. An erosion (resp. dilation) by  $B$  followed by a dilation (resp. erosion) by  $\tilde{B}$ , the symmetrical element of  $B$ , is called an opening (resp. closing) operation:

$$\begin{aligned} \text{Opening} : A \circ B &= (A \ominus B) \oplus \tilde{B} \\ \text{Closing} : A \bullet B &= (A \oplus B) \ominus \tilde{B} \end{aligned} \tag{6}$$

## 2.3 H-convex polygons

A shape in  $R \times R$  is said to be H-convex if it corresponds to the (possibly unbounded) intersection of a set of half-planes [1], or, equivalently, if it contains all line segments defined by a couple of its points. A discrete set of points (in  $Z \times Z$ ) is said to be H-convex if it is equal to the intersection of a H-convex shape of  $R \times R$  with  $Z \times Z$  [7]. As a consequence, a discrete H-convex shape is not necessarily connected in the usual sense (4- or 8-connexity). A bounded discrete H-convex shape is by definition the convex hull of a finite set of discrete points and hence is a polygon. In Normand [10], a method to decompose arbitrary discrete H-convex polygons is presented and applied to the construction of structuring elements for binary morphology.

We call 8-convex polygon, a 8-connected discrete H-convex polygon whose edge directions are restricted to  $0^\circ, 45^\circ, 90^\circ,$  and  $135^\circ$  like the structuring elements used in Xu [19]. In the following, we show that 8-convex polygons can be decomposed with a fixed (4) number of 2PSEs, whereas this amount varies for general H-convex polygons.

## 2.4 Xu decomposition

Xu [19] proved that any 8-convex polygon can be built by dilations with a restricted set of generating structuring elements. The 13 generating elements proposed by Xu are shown in Fig. 1. They constitute the minimal set to generate any 8-convex polygon, i.e., any convex polygon whose border is 8-connected. Their origin is normally the center of the cross, but in order to restrict later considerations to causal relationships, we replace the elements by their translated version whose origin is the last point in a raster scan order. According to (3) it introduces only a translation of the resulting image. In the following the translated version of an element  $Q_x$  will be referred to  $Q'_x$ . Figure 2 provides an example of an 8-convex structuring element and its Xu decomposition.



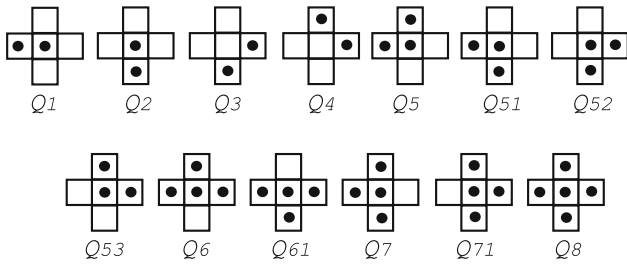


Fig. 1 Generating set of structuring elements for 8-convex polygons

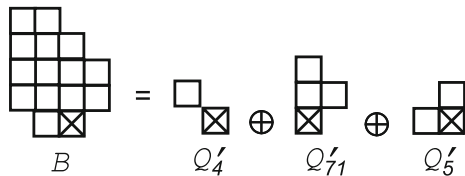


Fig. 2 Example of structuring element decomposition

### 2.5 Causal recursive decomposition of structuring elements

We have demonstrated in Normand [10] that all the Xu’s decomposition elements can themselves be built as unions and dilations of the four two-pixel structuring elements (2PSEs) depicted in Fig. 3a. By extension, it implies that these 2PSEs are also sufficient to build any 8-convex element as a series of both set unions and dilations. The main advantage of using 2PSEs is that they involve a restricted 8-connected causal neighborhood of analysis defined by four elementary translations  $p_k, k \in \{1, \dots, 4\}$  (see Fig. 3b).

We have established in Normand [10] that any convex structuring element can be iteratively constructed: for each step of the construction, already-built elements can be joined together in any combination of translations and set unions. We have also introduced a deterministic method for convex shape decomposition.

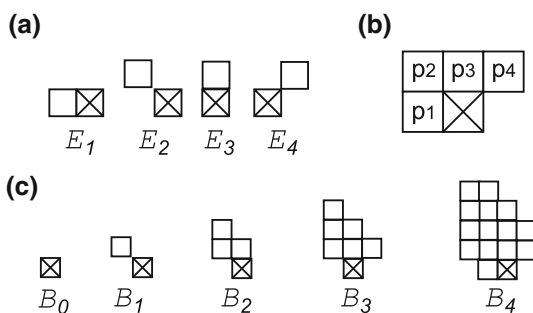


Fig. 3 2PSE decomposition a Elementary elements, b corresponding translations and c example of growing structuring family

Restricting elementary structuring elements to the set of four 2PSEs depicted in Fig. 3b not only leads to consideration of only 8-convex elements but also limits the analysis space to an 8-connected causal neighborhood. In this case, each 8-convex  $B_i$  can be recursively constructed according to

$$B_{-1} = \emptyset; B_0 = \{O\}; \forall i > 1, B_i = \bigcup_{k \in \{1, \dots, 4\}} B_{I_k(i)} \oplus E_k, \quad (7)$$

where  $I_k(i)$  ( $I_k(i) < i$ ) denotes which previous element belonging to the family and dilated by  $E_k$  is used to build  $B_i$ . The empty set is represented by  $I_k(i) = -1$  and  $B_0$  only contains the origin.  $\{B_i\}_{i=0 \dots N}$  forms an increasing family of structuring elements as long as  $B_{i-1} \subset B_i$ .

Going on with the previous example we see that the translated Xu elements ( $Q'_4$ ,  $Q'_{71}$ , and  $Q'_5$ ) can be built by dilations and unions of 2PSEs:

$$Q'_4 = E_2, Q'_{71} = (E_3 \oplus E_3) \cup E_4, Q'_5 = E_1 \cup E_3 \quad (8)$$

In most of the cases (9/13), the  $Q'$  are expressed by a unique set of unions, like  $Q'_4$  and  $Q'_5$ , and the recursive relationship from (7) is straightforward. To change over from the Xu decomposition scheme to the proposed one for the four left structuring elements, we introduce an intermediate step: let us assume that  $B_i = B_{i-2} \oplus Q'_{71}$ , according to (5) and (8) we get

$$B_i = (B_{i-2} \oplus E_3 \oplus E_3) \cup (B_{i-2} \oplus E_4) \\ = (B_{i-1} \oplus E_3) \cup (B_{i-2} \oplus E_4) \quad (9)$$

where  $B_{i-1} = B_{i-2} \oplus E_3$ . From relation (7) it gives  $I_3(i-1) = i-2$ ,  $I_3(i) = i-1$  and  $I_4(i) = i-2$ . The corresponding growing family is given below, and the iterative construction is depicted in Fig. 3c.

Dil. by $Q'_4$	$B_1 = B_0 \oplus E_2$
Dil. by $Q'_{71}$ (step 1)	$B_2 = B_1 \oplus E_3$
Dil. by $Q'_{71}$ (step 2)	$B_3 = (B_2 \oplus E_3) \cup (B_1 \oplus E_4)$
Dil. by $Q'_5$	$B_4 = (B_3 \oplus E_1) \cup (B_3 \oplus E_3)$

(10)

A growing structuring element family is then very easily described by providing only the four  $I_k(i)$  indices per level. In practice most of the structuring elements used are symmetrical ones. In that particular case, all of them (except crosses) can be obtained when restricting formulae (7) to only elementary dilations. The simplified associated expression is given by

$$B_i = B_{i-1} \oplus E_i, E_i \in \{E_1, E_2, E_3, E_4\} \quad (11)$$

Only one index per level is necessary here for the decomposition description. The alternative use of  $E_1, E_2, E_3, E_4$  leads to a circular structuring element, while limiting the construction to  $E_1$  and  $E_3$  generates rectangular ones.

2.6 Grayscale image recursive dilation and erosion

In Normand [10], we took advantage of the previous structuring element decomposition method by introducing a fast binary erosion operator. The binary case, which presents a fixed number of operations whatever the structuring element, cannot be used to expand the grayscale morphology. Even if comparable performances cannot be achieved, we propose here a fast algorithm for grayscale erosion and dilation operators. The principle will be detailed only for dilation.

First, consider that as the four 2PSEs contain the origin, an elementary dilation by  $E_k$  is equivalent to the union of the element and the element translated by  $p_k$ :

$$B \oplus E_k = B \cup (B)_{p_k} \tag{12}$$

Let  $\{B_i\}_{i=0,\dots,N}$  be a family of growing structuring elements constructed recursively by union sets and elementary dilations according to (7). In the following  $D_{B_i}$  states the dilation of  $A$  by  $B_i$ . Based on the 2PSE decomposition a recursive computation of  $D_{B_i}$  is possible:

$$\begin{aligned} D_{B_i} &= A \oplus B_i \\ &= A \oplus \left( \bigcup_{k \in \{1,\dots,4\}} B_{I_k(i)} \oplus E_k \right) \\ &= A \oplus \left( \bigcup_{k \in \{1,\dots,4\}} B_{I_k(i)} \cup (B_{I_k(i)})_{p_k} \right) \\ &= \text{Max} \left\{ (A)_p \mid p \in \bigcup_{k \in \{1,\dots,4\}} B_{I_k(i)} \cup (B_{I_k(i)})_{p_k} \right\} \\ &= \text{Max}_{k \in \{1,\dots,4\}} \left( \text{Max} \left\{ (A)_p \mid p \in (B_{I_k(i)} \cup (B_{I_k(i)})_{p_k}) \right\} \right) \\ &= \text{Max}_{k \in \{1,\dots,4\}} \left( \text{Max} \left\{ (A)_p \mid p \in B_{I_k(i)} \right\}, \right. \\ &\quad \left. \text{Max} \left\{ (A)_p \mid p \in (B_{I_k(i)})_{p_k} \right\} \right) \\ &= \text{Max}_{k \in \{1,\dots,4\}} \left( D_{B_{I_k(i)}}, (D_{B_{I_k(i)}})_{p_k} \right) \end{aligned} \tag{13}$$

These results show that the dilated values when considering successively all the elements in the family, are recursively obtained by performing a maximum extraction between the values of the dilation by the elements involved to build the current one, at the current position and at positions  $p_k$  in the 8-connected causal neighborhood. The process is fully regular because it uses a raster scan of the image.

In this scheme, resources requirements are linear.

- **Complexity:** Processing requirements concern Max operators. If  $N_D$  and  $N_U$  are, respectively, the total number of elementary dilations and unions used to

build  $B_i$ , then the algorithm requires only  $N_D + N_U$  comparisons to perform the dilation of an image by  $B_i$  whatever the shape of the structuring element.

- **Memory:** Only intermediate results  $\{D_{B_i}\}_{i=0,\dots,N}$  have to be recorded, so the number of stored elements is given by the number of decomposition levels. When only  $p_1$  is involved for one level, the intermediate value corresponds to the previous computed one, and an unique variable is necessary to store the result. In other cases, a line-memory per level is necessary to store intermediate results from the previous line.

Figure 4 shows the progressive steps to get dilated values up to the structuring element  $B_3$  from Fig. 3.  $D_{B_3}$  computation requires five comparisons, and  $D_{B_4}$  eight.

When  $B$  is a structuring element composed of only dilations [(11) for symmetrical elements], the operations at each stage are reduced to a unique comparison between the dilated values by the previous element at the current position and at position  $p_i$ ,  $p_i \in \{p_1, p_2, p_3, p_4\}$ , so that

$$\begin{aligned} B_i &= B_{i-1} \cup (B_{i-1})_{p_i} \\ \Rightarrow D_{B_i} &= \text{Max} \left( D_{B_{i-1}}, (D_{B_{i-1}})_{p_i} \right) \end{aligned} \tag{14}$$

In this particular case, the number of operations is also given by the number of decomposition levels.

2.6.1 Image borders

To handle the problem of image borders, one has generally to expand the image or to introduce conditional processing. For a dilation operation, a suitable way consists of considering values as 0 outside borders. This can be done here without preliminary image expansion as explained below.

- If  $L$  is the image width, then the line-memories have to be of size  $(L+2)$ .
- Memory-lines have to be initialized to 0 once at the beginning of the process and variables to store results in

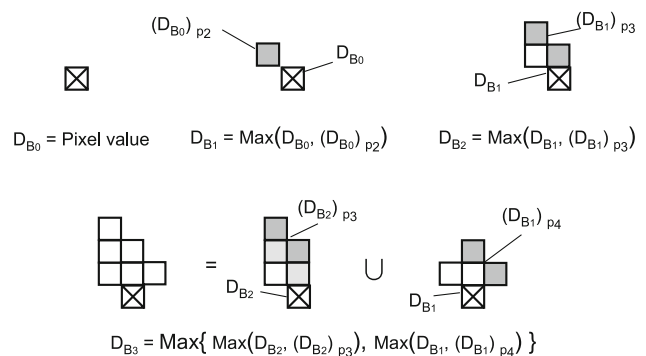


Fig. 4 Recursive dilation process

$p_1$  position have to be initialized at the beginning of each line process.

- The position values for pixels in the image are considered in the range  $[1, \dots, L]$ . Thus, for pixel position 1 (resp.  $L$ ) and for  $p_2$  (resp.  $p_4$ ) neighborhood, line-memory is accessed at address 0 (resp.  $L + 1$ ) and provides 0 as a dilated value.

Considering 0 values outside the borders has of course an impact on results near these borders. Assuming that that  $D_{B_i} = 0$  if  $B_i$  is totally outside the image, it modifies (13):

$$D_{B_i} = \text{Max}_{k \in \{1, \dots, 4\}} \begin{pmatrix} D_{B_{I_k(i)}}, (D_{B_{I_k(i)}})_{p_k} & \text{if } B_{I_k(i)} \subset A \\ D_{B_{I_k(i)}} & \text{otherwise.} \end{pmatrix} \quad (15)$$

From the previous formulae, one can see that the process automatically adapts the shape of the structuring element near borders and that the actual structuring element expression is

$$B_i = \bigcup_{k \in \{1, \dots, 4\}} B_{I'_k(i)} \cup (B_{I''_k(i)})_{p_k}, \text{ with} \quad (16)$$

$$I'_k(i) = \text{argmax}(j \in \{0, \dots, I_k(i)\} | B_j \subset A)$$

$$I''_k(i) = \text{argmax}(j \in \{0, \dots, I_k(i)\} | (B_j)_{p_k} \subset A)$$

To sum up for this part, the proposed method enables an efficient software implementation of dilation and erosion operators with the following main features:

- an unique scan of the image whatever the shape and size of the structuring element,
- an easy and systematic way to describe any 8-convex structuring element,
- limited resources requirements in terms of both memory and computation, and
- automatic adaptation to borders without image expansion nor conditional processing.

Further improvements could be made, in particular concerning the number of comparisons. In Chien et al. [3] introduced the concept of Partial-Result-Reuse (PRR). The idea is to limit the number of operations by avoiding redundant comparisons as much as possible and by exploiting already computed data. The example of a 1D structuring element of 8 pixels (called  $B_4$ ) given by Chien is illustrated Fig. 5.  $B_4$  is recursively constructed by

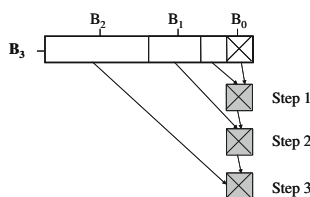


Fig. 5 PRR principle for a 1D structuring element of 8 pixels

gathering non-overlapped structuring elements. Intermediate results have to be stored at different distances from the current position: 1 for  $B_0$ , 2 for  $B_1$ , and 4 for  $B_3$ , giving a total of seven recorded values.

If we compare this with our approach, the same structuring element would be built according to

$$B_7 = B_6 \cup (B_6)_{p_1}$$

$$\dots$$

$$B_1 = B_0 \cup (B_0)_{p_1} \quad (17)$$

The dilation is performed here after seven comparisons. Actually, 1D dilations are the worst case for our method in terms of size of the element versus number of comparisons. Concerning memory, both methods require the storage of seven values, but our method involves a unique distance (equal to 1), whereas the depth for PRR is dependant on the element decomposition.

The PRR concept has been extended to square and circular elements. Square representation is the simplest shape leading to the greatest computational reduction. For circular elements (13 pixel elements), Chien's heuristic method requires six computation steps, whereas the proposed method requires eight. However, the authors do not propose a systematic method of decomposition. For a dichotomic approach, the problem can be reformulated by trying to find the minimum element  $B_j$  of the family (where  $j \in [0, \dots, i - 1]$ ) such that  $B_i$  can be expressed as

$$B_i = B_j \cup (B_j)_{T \times p_i} \quad (18)$$

$$T \in \left\{ 1, \dots, \left\lfloor \frac{i}{2} \right\rfloor \right\}, \quad p_i \in \{p_1, p_2, p_3, p_4\}$$

The 1D case is simple. Formula (17) can first be expanded, and, by factoring process, it is easy to find that there are four possible solutions meeting the condition set out in (17). Among them,  $B_3$  is the minimum element.

$$B_7 = B_6 \cup (B_6)_{p_1} \quad B_7 = B_5 \cup (B_5)_{2p_1} \quad (19)$$

$$B_7 = B_4 \cup (B_4)_{3p_1} \quad B_7 = B_3 \cup (B_3)_{4p_1}$$

The approach can be extended to 2D space but only for simple element shapes. For more complex elements, PRR is more difficult to determine, and as mentioned earlier, the benefit of PRR decreases when the complexity of the pattern increases.

### 2.6.2 Opening and closing

Opening and closing operations imply the need to consider symmetrical elements. If  $B$  is causal,  $\tilde{B}$  is necessarily anti-causal, which would lead to an inverse scan of the image in order to preserve the sequential feature of the process. A better solution takes advantage of the property (3) by defining  $B_T = (\tilde{B})_{p_B}$ , with  $p_B$  a fixed vector such that  $B_T$  is



causal. Then the use of  $B_T$  instead of  $\tilde{B}$  simply requires a global image translation of  $p_B$  once processing has been completed. For a regular data flow it consists merely of introducing a constant delay in the output flow.

When  $B$  is symmetrical, deduced from (11), then  $B$  and  $\tilde{B}$  have the same shape, and the same structuring element can be used.

### 3 Specific pipeline architecture

The regularity of the algorithm and the restriction of the relationship area to the 8-connected causal neighborhood enable the design of a fast generic and regular specific architecture, requiring few resources in terms of memory and logic. A dilation implementation suited for an FPGA target is presented in the following paragraphs:

The whole architecture is synchronized by a global clock with a data flow rate of one input and output pixel data for each clock pulse. Each elementary stage computing  $D_{B_i}$  from previous values, requires two types of components: a memory module to store and to provide previous results and a combinatory operative part to realize the maximum extraction.

For the following, let  $T$  be the clock period,  $D_{B_i}(n)$  be the dilation value by the structuring element  $B_i$  at time  $n \times T$  ( $n$ th pixel),  $L$  be the size of an image row and  $M$  be the number of bits to encode a value.

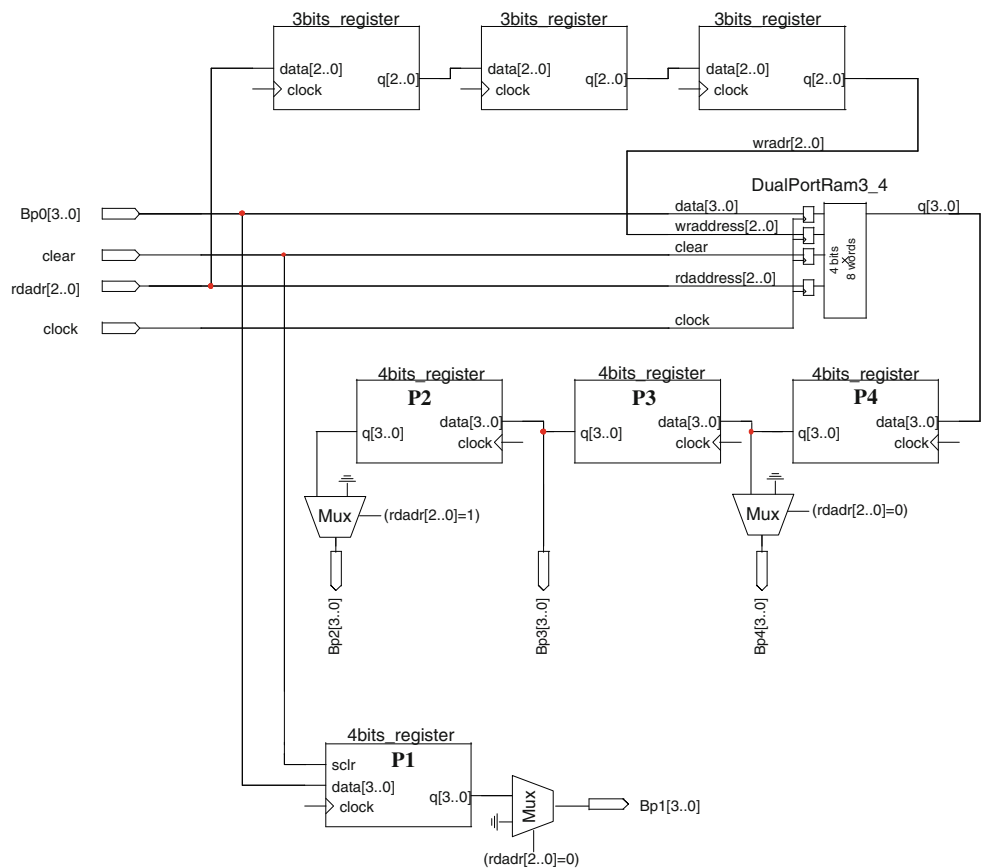
#### 3.1 Memory module

For easier systematic design of the architecture, a generic memory module has been designed. The knowledge of a previous result in the  $\{p_2, p_3, p_4\}$  neighborhood implies the storage of one row at each step of the decomposition. We have chosen a “dual port RAM” memory solution allowing both a reading and a writing access during one clock cycle. The following temporal relationships avoid the need for multiple address management process in each of the various stages:

$$\begin{aligned} (D_{B_i}(n))_{p_1} &= D_{B_i}(n-1) \\ (D_{B_i}(n))_{p_4} &= D_{B_i}(n-(L-1)) \\ (D_{B_i}(n))_{p_3} &= D_{B_i}(n-L) = (D_{B_i}(n-1))_{p_4} \\ (D_{B_i}(n))_{p_2} &= D_{B_i}(n-(L+1)) = (D_{B_i}(n-1))_{p_3} \end{aligned} \tag{20}$$

Figure 6 presents the general solution for the memory module. For this example we have considered  $L = 7$

Fig. 6 Generic elementary memory module



(3 bits to encode the address) and  $M = 4$ . The storage  $D_{B_i}(n - 1)$  requires only a simple register. The memory is addressed at a fixed relative position  $[(n + 2) \text{ Modulo } L]$  to provide  $(D_{B_i})_{p_4}$  at the next clock pulse. Then the dilated values at the two other positions are available by a temporal pipeline through two registers. The reading and writing addresses have a constant difference (equal to 3) which can be also seen as a temporal delay in terms of clock cycle. Therefore the simplest architectural solution consists of the implementation of three registers.

Of course, all the resources displayed in the memory module are not always necessary. For instance, when only  $p_1$  among the neighborhood is involved it requires only one register. However, these kinds of optimizations are generally automatically performed by synthesis tools removing all unused resources.

### 3.2 Image borders

As mentioned earlier, image borders induce specific processing which can break the data flow regularity of the architecture. To avoid such problem, a solution has been designed. Only one extra storage element is required for the memory-lines, and as borders for both the  $p_2$  and  $p_4$  neighborhoods. Figure 7 illustrates the behavior of the output memory module on right and left borders. To flush the pipeline at the end of each row, and to load values at the beginning, an additional clock pulse is performed. Then a control signal (*wren*) is generated to avoid to overwrite stored value (0) at address  $L$ , and to reset value in  $p_1$ .

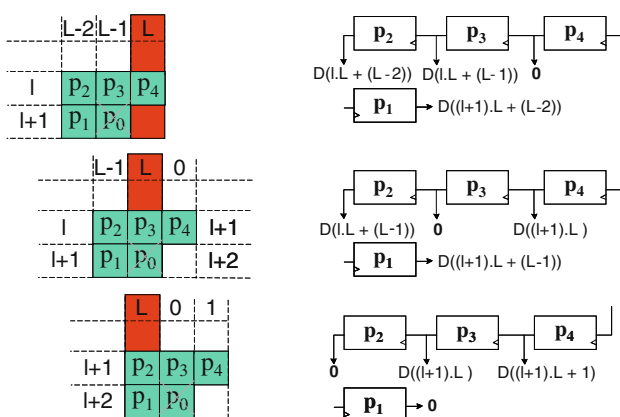


Fig. 7 Border processing

### 3.3 Global architecture

The global architecture (see Fig. 8) is composed of a simple global control (counter) providing the address and the set of pipelined elementary stage of dilation. The configuration of each stage consists of determining the number of elementary dilations and connecting the output of each memory module according to the  $B_i$  definition. The “Max extraction module” designates an elementary subtractor whose carry-out output drives the selection input of a multiplexer. The temporal performances of the system are determined by the maximal data path time fixing the minimal period. Thus, the maximal frequency is dependent on  $(N_D + N_I)$ . A conventional solution relies on the design of a multi-layer synchronous architecture by introducing additional registers between stages to reduce the critical data path. Then, the critical data-path will be determined by the longest path between two elementary layers, leading to a faster clock frequency, independently of the number of stages. The latency of the global system is simply given in terms of clock pulse by the number of layers to cross. The registers are represented by gray boxes in the global architecture diagram. In practice, registers are available in FPGA at the output of all logic modules. So, additional registers for a pipeline implementation are only required for data paths with no operation between stages such as the address of the counter or the propagation of  $B_2$  towards  $B_4$ .

Figure 9 shows expected results for a dilation of an input image by  $B_2$ , and Fig. 10 gives the results of the simulation.

### 3.4 Implementation

Comparative implementation efficiency with existing architectures is summarized in Table 1. Except for the Diamantaras solution, all the other techniques have been optimized for only rectangular elements. For memory and cycle requirements, the proposed method achieves the same performances as the best solutions. Only Chien architecture provides a better solution in terms of comparators, but as mentioned earlier, the PRR concept is essentially efficient for 1D and rectangular elements. Moreover, PRR and the proposed methods are not incompatible: additional improvements in order to reduce the number of operations, for simple shapes, can be a posteriori realized after the decomposition definition.

The proposed architecture has been synthesized into FPGAs. As the architecture requires few resources, a medium FPGA such as an APEX 20KC (Altera) used in these experiments is sufficient to support dilations by large

Fig. 8 Global architecture

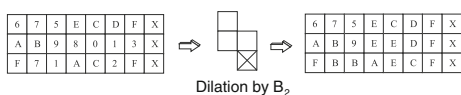
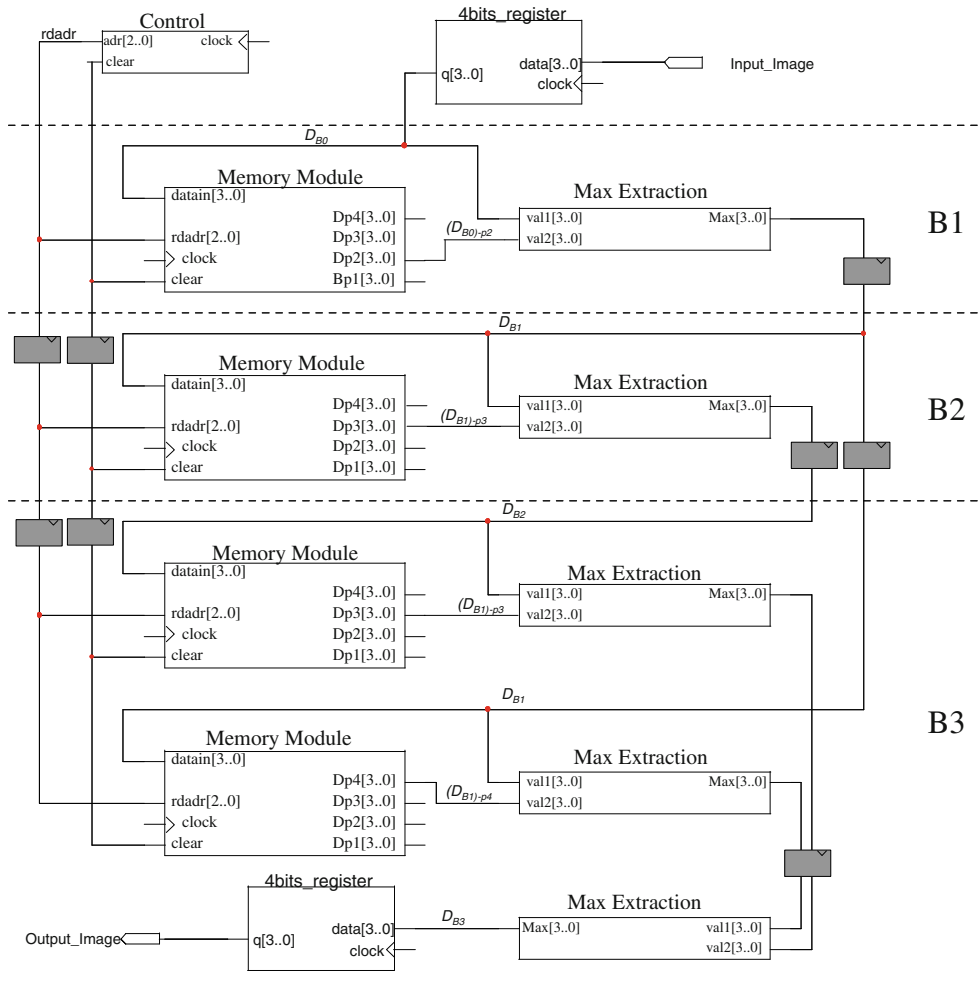


Fig. 9 Expected values for a dilation by  $B_2$

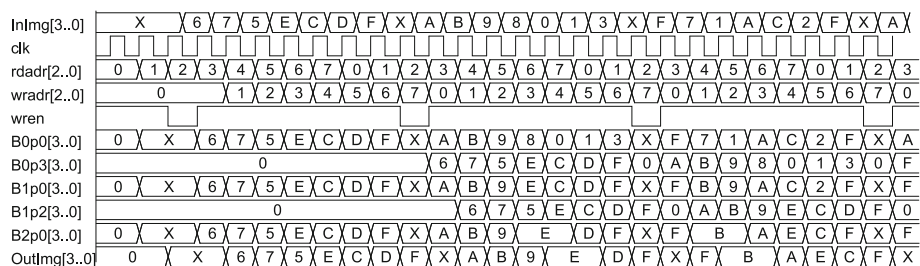
structuring elements. Maximum clock rate is dependant only on one combinatorial stage and thus dependent only on  $M$ . For  $M = 8$  the maximum clock frequency is 50 MHz. For instance, a  $512 \times 512$  grayscale image is

computed during one pass in 5 ms whatever the structuring element shape and size.

### 4 Conclusion

In this paper, we have presented an algorithm enabling the design of fast mathematical grayscale morphology operations. We have proposed a systematic way to describe any 8-convex structuring elements through a generic decomposition process using 2-pixel structuring elements. Then these

Fig. 10 Simulation results for a dilation by  $B_2$



**Table 1** Flexibility and implementation efficiency comparison between state-of-the-art architectures for a one pass dilation by a  $7 \times 7$  square structuring element

Architecture	Support non flat structuring element	Support non rectangular structuring element	Comparator counts	Number of delay elements	Cycles required per frames
Pitas	No	No	8	$7L + 7$	$L(H + 7) + 6$
Coltuc	No	No	6	$6L + 6$	$L(H + 6) + 5$
Ong	Yes	No	7	$[6L + 6] + 1$	$7LH$
Diamantaras	Yes	Yes	48	$[6L + 42]$	$L(H + 6)$
Ruetz	No	No	12	$6L + 6$	$L(H + 6) + 5$
Sheu	Yes	No	13	$6L + 20$	$7LH + 8$
Chien	No	Partially	6	$6L + 6$	$L(H + 6) + 5$
This work	No	Yes	12	$6L + 6$	$L(H + 6) + 5$

8-convex structuring elements are recursively constructed by elementary dilations and union sets, based on a fixed and close neighborhood. This decomposition system has been used to define the efficient implementation of elementary morphological operators in both software and hardware. Computation complexity is directly correlated to the number of elementary dilations involved in the decomposition.

## References

- Boltyanskii, V.G., Soltan, P.S.: Combinatorial geometry and convexity classes. *Russian Mathematical Surveys* **33**(1), 1–45 (1978)
- Chen, S., Haralick, R.: Recursive erosion, dilation, opening, and closing transforms. *IEEE Trans. Image Process.* **4**(3), 335–345 (1995)
- Chien, S., Ma, S., Chen, L.: Partial-result-reuse architecture and its design technique for morphological operations with flat structuring elements. *IEEE Trans. Circuits Syst. Video Technol.* **15**(9), 1156–1169 (2005)
- Clienti, C., Bilodeau, M., Beucher, S.: An efficient hardware architecture without line memories for morphological image processing. In: *ACIVS '08: Proceedings of the 10th International Conference on Advanced Concepts for Intelligent Vision Systems*, pp. 147–156 (2008)
- Coltuc, D., Pitas, I.: Fast computation of a class of running filters. *IEEE Trans. Signal. Process.* **46**(6), 549–553 (1998)
- Diamantaras, K., Kung, S.: A linear systolic array for real-time morphological image processing. *J. VLSI Signal Process.* **17**, 43–55 (1997)
- Eckhardt, U.: Digital lines and digital convexity. *Digital Image Geom.* **2243**, 209–228 (2001)
- Gil, J., Kimmel, R.: Efficient dilation, erosion, opening, and closing algorithms. *IEEE Pattern Anal. Mach. Intell.* **24**(12), 1607–1617 (2002)
- Ji, L., Piper, J., Tang, J.: Erosion and dilation of binary images by arbitrary structuring elements using interval coding. *Pattern Recogn. Lett.* **(9)**, 201–249 (1989)
- Normand, N.: Convex structuring element decomposition for single scan binary mathematical morphology. In: *Conference on Discrete Geometry for Computer Imagery, Naples* (2003)
- Ong, S., Sunwoo, M.: A new cost-effective morphological filter chip. In: *IEEE Workshop Design Signal Processing Systems*, pp. 421–430 (1997)
- Pitas, I.: Fast algorithms for running ordering and max/min calculation. *IEEE Trans. Circuits Syst.* **36**(6), 795–804 (1989)
- Ruetz, P., Brodersen, R.: Architectures and design techniques for real-time image processing IC'S. *IEEE J. Solid State Circuit* **22**(2), 233–250 (1987)
- Serra, J.: *Image Analysis and Mathematical Morphology*. Academic Press, London (1982)
- Sheu, M., Wang, J., Chen, J., Suen, A., Jeang, Y., Lee, J.: A data-reuse architecture for gray-scale morphologic operations. *IEEE Trans. Circuits and Syst II, Analog Digital Signal Process* **39**(10), 753–756 (1992)
- Soille, P., Breen, E.J., Jones, R.: Recursive implementation of erosions and dilations along discrete lines at arbitrary angles. *IEEE Pattern Anal. Mach. Intell.* **18**(5), 562–567 (1996)
- Van Herk, M.: A fast algorithm for local minimum and maximum filters on rectangular and octagonal kernels. *Pattern Recogn. Lett.* **13**(7), 517–521 (1992)
- Wang, X., Bertrand, G.: An algorithm for a generalized distance transformation based on minkowski operations. In: *9th International Conference on Pattern Recognition*, pp. 1164–1168 (1988)
- Xu, J.: Decomposition of convex polygonal morphological structuring elements into neighborhoods subsets. *IEEE Trans. Pattern Anal. Mach. Intell.* **13**(2), 153–162 (1991)

## Author Biographies

**Olivier Déforges** is a professor at National Institute of Applied Sciences of Rennes (INSA). He graduated in electronic engineering from the Polytechnique University of Nantes, France, in 1992, where he also received a Ph.D. degree in image processing in 1995. In 1996, he joined the Department of Electronic Engineering at the INSA, Rennes Scientific and Technical University. He is a member of the UMR CNRS 6164 IETR Laboratory in Rennes. His principal research interests are image and video lossy and lossless compression, image understanding, fast prototyping, and parallel architectures.

**Nicolas Normand** was born in 1969. He received the MS degree in 1992 and his Ph.D. from the university of Nantes, France, in 1997. He has been an assistant professor in the computer science department of École polytechnique de l'université de Nantes from 1997. His areas of interest include digital geometry, data networks and the related Mojette transform.

**Marie Babel** is an associate professor at the National Institute of Applied Sciences (INSA) of Rennes. She graduated in Electronic Engineering from the INSA of Rennes and in Signal and Image Processing from the University of Rennes 1. She received a Ph.D. degree in image processing in 2005. She is a member of the UMR CNRS 6164 IETR Laboratory in Rennes. Her principal research interests include image and video lossy and lossless coding, advanced video schemes and image transmission securization processes.



Contents lists available at ScienceDirect

## Pattern Recognition

journal homepage: [www.elsevier.com/locate/pr](http://www.elsevier.com/locate/pr)

# Medial axis lookup table and test neighborhood computation for 3D chamfer norms

Nicolas Normand\*, Pierre Evenou

IRCCyN UMR CNRS 6597, École polytechnique de l'Université de Nantes, Rue Christian Pauc, La Chantrerie, 44306 Nantes Cedex 3, France

## ARTICLE INFO

### Article history:

Received 28 June 2008

Received in revised form 5 November 2008

Accepted 10 November 2008

### Keywords:

Chamfer distances  
Weighted distances  
Medial axis  
Test neighborhood  
Polytopes

## ABSTRACT

Chamfer distances are discrete distances based on the propagation of local distances, or weights, defined in a mask. The medial axis, i.e. the centers of maximal balls (balls which are not contained in any other ball), is a powerful tool for shape representation and analysis. The extraction of maximal disks is performed in the general case by testing the inclusion of a ball in a local neighborhood with covering relations usually represented by lookup tables.

The proposed method determines if a mask induces a norm and in this case, computes the lookup tables and the test neighborhood based on geometric properties of the balls of chamfer norms, represented as  $\mathcal{H}$ -polytopes. The method does not need to repeatedly scan the image space, and improves the computation time of both the test neighborhood detection and the lookup table computation.

© 2008 Elsevier Ltd. All rights reserved.

## 1. Introduction

The distance transform  $DT_X$  of a binary image  $X$  is a function that maps each point  $p$  to its distance to the closest background pixel i.e. with the radius of the largest open disk centered in  $p$  included in the image. Such a disk is said to be maximal if it is not contained in another disk also included in  $X$ . The set of centers of maximal disks, also called the medial axis, is a convenient description of binary images for many applications ranging from image coding to shape recognition. Its attractive properties are reversibility and (relative) compactness.

Algorithms for computing the distance transform are known for various discrete distances [1–5]. In this paper, we will focus on chamfer (or weighted) distances which are defined by a set of weighted vectors described by a mask, called the *chamfer mask*. The classical medial axis extraction method is based on the removal of non-maximal disks in the distance transform. It is thus mandatory to describe the covering relation of disks, or at least the transitive reduction of this relation. For simple distances, this knowledge is summarized in a local maximum criterion [1] but a most general method uses lookup tables (LUT) for that purpose [6].

In this paper, we propose a two-phase method that determines if a mask induces a norm and in this case, computes the LUT and

the test neighborhood based on geometric properties of the balls of chamfer norms.

The first phase starts from the 3D chamfer mask and produces a triangulation of the chamfer neighbors inversely pondered by the chamfer weights. The algorithm conceptually operates on the convex hull of these weighted vectors and creates a triangulation of this convex hull. It produces two results: a norm condition check and a description of the geometry of the chamfer balls for the given chamfer mask. The second phase is performed only for norms; from the general geometry obtained during the previous phase, each chamfer ball is described as the intersection of a set of half-spaces ( $\mathcal{H}$ -description). The norm condition is sufficient for the balls to be convex, ensuring the validity of their  $\mathcal{H}$ -description. This  $\mathcal{H}$ -description is then used to compute the test neighborhood  $\mathcal{T}$  and the LUT values.

Basic notions, definitions and known results about chamfer disks and medial axis LUT are recalled in Section 2. Then Section 3 justifies the use of polytope formalism in our context and presents the principles of the method. In Section 4, a triangulation algorithm is given for the 3D case. The computation of the test neighborhood  $\mathcal{T}$  and of the LUT values is explained in Section 5. Finally, Section 6 gives speed-up figures for the overall algorithm compared to a reference implementation [7]. Notice that we do not address the actual computation of the medial axis which remains unchanged.

This paper is a more detailed version of [8] with an added method for the triangulation of 3D balls and the detection of norm conditions. The algorithms for LUT and  $\mathcal{T}$  computation were adapted to the 3D case from [8].

\* Corresponding author. Tel.: +33 2 40 68 32 07; fax: +33 2 40 68 32 32.

E-mail addresses: [Nicolas.Normand@polytech.univ-nantes.fr](mailto:Nicolas.Normand@polytech.univ-nantes.fr) (N. Normand), [Pierre.Evenou@polytech.univ-nantes.fr](mailto:Pierre.Evenou@polytech.univ-nantes.fr) (P. Evenou).

## 2. Chamfer distances and medial axis

### 2.1. Discrete distances

**Definition 1** (*Discrete distance, metric and norm*). Consider a function  $d : \mathbb{Z}^n \times \mathbb{Z}^n \rightarrow \mathbb{N}$  and the following properties  $\forall x, y, z \in \mathbb{Z}^n, \forall \lambda \in \mathbb{Z}$ :

1. *positive definiteness*  $d(x, y) \geq 0$  and  $d(x, y) = 0 \Leftrightarrow x = y$ ,
2. *symmetry*  $d(x, y) = d(y, x)$ ,
3. *triangle inequality*  $d(x, z) \leq d(x, y) + d(y, z)$ ,
4. *translation invariance*  $d(x + z, y + z) = d(x, y)$ ,
5. *positive homogeneity*  $d(\lambda x, \lambda y) = |\lambda| \cdot d(x, y)$ .

$d$  is called a *distance* if it verifies conditions 1 and 2, a *metric* with conditions 1–3 and induces a *norm* if it also satisfies conditions 4 and 5.

Most discrete distances are built from a definition of neighborhood and connected paths (path-based distances), the distance between two points being equal to the length of the shortest path that joins them [9]. Distance functions differ by the neighborhoods used to build paths and by the way path lengths are measured. For the simple distance  $d_4$  (denoted by  $d$  in [1]), defined in the square grid  $\mathbb{Z}^2$ , each pixel has four neighbors located at its top, left, bottom and right edges. Similarly, for distance  $d_8$  ( $d^*$  in [1]), each pixel has four extra diagonally located neighbors. In both cases,  $d_4$  and  $d_8$ , the length of a path is defined as its number of displacements, whereas it is measured as a weighted sum of displacements for chamfer distances [2,3] or by the displacements allowed at each step for neighborhood sequence distances [3,9], or even by a mixed approach of weighted neighborhood sequence paths [5].

For a given distance  $d$ , the closed ball  $B_{\leq}$  and open ball  $B_{<}$  of center  $c$  and radius  $r$  are the sets of points of  $\mathbb{Z}^n$ :

$$B_{<}(c, r) = \{p : d(c, p) < r\},$$

$$B_{\leq}(c, r) = \{p : d(c, p) \leq r\}. \quad (1)$$

Since the codomain of  $d$  is  $\mathbb{N}$ , we have  $\forall r \in \mathbb{N}, d(c, p) \leq r \Leftrightarrow d(c, p) < r + 1$ . So

$$\forall r \in \mathbb{N}, B_{\leq}(c, r) = B_{<}(c, r + 1). \quad (2)$$

**Definition 2** (*Distance transform*). The distance transform  $DT_X$  of the binary image  $X$  is a function that maps each point  $p$  to its distance to the closest background pixel:

$$DT_X : \mathbb{Z}^n \rightarrow \mathbb{N},$$

$$DT_X(p) = \min\{d(p, q) : q \in \mathbb{Z}^n \setminus X\}. \quad (3)$$

Alternatively, since all points at a distance less than  $DT_X(p)$  from  $p$  belongs to  $X$  ( $B_{<}(p, DT_X(p)) \subset X$ ) and at least one point at a distance equal to  $DT_X(p)$  is not in  $X$  ( $B_{\leq}(p, DT_X(p)) = B_{<}(p, DT_X + 1) \not\subset X$ ) then  $DT_X(p)$  is the radius of the largest open disk centered in  $p$  included in the image:

$$DT_X(p) = \max\{r : B_{<}(p, r) \subset X\}. \quad (4)$$

**Definition 3** (*Medial axis*). The medial axis  $MA_X$  of the binary image  $X$  is the set of centers of maximal open balls in  $X$  valued with their radii:

$$MA_X : \mathbb{Z}^n \rightarrow \mathbb{N},$$

$$MA_X(p) = \begin{cases} 0 & \text{if } \exists q, r' \text{ s.t. } B_{<}(p, DT_X(p)) \subsetneq B_{<}(q, r') \subset X, \\ DT_X(p) & \text{otherwise.} \end{cases} \quad (5)$$

### 2.2. Chamfer distances

**Definition 4** (*Chamfer mask, Thiel [10]*). A weighting  $M = (\vec{v}; w)$  is a vector  $\vec{v}$  of  $\mathbb{Z}^n$  associated with a weight  $w$  (or local distance). A chamfer mask  $\mathcal{M}$  is a central-symmetric set of weightings having positive weights and non-null displacements, and containing at least one basis of  $\mathbb{Z}^n$ :  $\mathcal{M} = \{M_i \in \mathbb{Z}^n \times \mathbb{N}^* \}_{1 \leq i \leq m}$ .

The grid  $\mathbb{Z}^n$  is symmetric with respect to the hyperplanes normal to the axes and to the bisectors (G-symmetry). This divides  $\mathbb{Z}^n$  into  $2^n \cdot n!$  subspaces (there are eight of them for  $\mathbb{Z}^2$  and 48 for  $\mathbb{Z}^3$ ). The particular subspace  $x_n \geq \dots \geq x_1 \geq 0$  is called the generator cone or simply generator and is denoted  $\mathcal{G}$ . From every point in  $\mathbb{Z}^n$ , we can determine its unique G-symmetrical point in  $\mathcal{G}$  by ordering the absolute values of its components in decreasing order. Conversely, from any point  $p$  in  $\mathcal{G}$ , we can derive all its G-symmetrical points by generating all  $n_1!$  permutations of the components of  $p$  and all  $2^{n_2}$  combinations of their signs where  $n_1$  and  $n_2$  are, respectively, the number of different absolute values of the components of  $p$  and the number of non-null components.

Chamfer masks are usually restricted to  $\mathcal{G}$  for simplicity. Weightings are then only given in  $\mathcal{G}$  and we denote  $\mathcal{M}|_{\mathcal{G}} = \{M_i \in \mathcal{G} \times \mathbb{N}^* \}_{1 \leq i \leq m}$ , the chamfer mask restricted to  $\mathcal{G}$ . A usual ordering of the points in  $\mathcal{G}$  is the lexicographical order;  $p$  is before  $q$  if for some  $i$ :  $p_0 \dots p_{i-1} = q_0 \dots q_{i-1}$  and  $p_i < q_i$ . According to this ordering, a common naming scheme assigns alphabetic letters to visible points (such that  $\gcd(p) = 1$ ). For instance, in the 2D grid:  $\vec{a} = (1 \ 0)$ ,  $\vec{b} = (1 \ 1)$ ,  $\vec{c} = (2 \ 1)$ ,  $\vec{d} = (3 \ 1)$ ,  $\vec{e} = (3 \ 2) \dots$ . The corresponding chamfer weights are, respectively, named  $a, b, c, d, e \dots$ .

**Definition 5** (*Chamfer distance, Thiel [10]*). Consider the chamfer mask  $\mathcal{M} = \{(\vec{v}_i; w_i) \in \mathbb{Z}^n \times \mathbb{N}^* \}_{1 \leq i \leq m}$ . The chamfer (or weighted) distance between two points  $p$  and  $q$  is

$$d(p, q) = \min \left\{ \sum \lambda_i w_i : p + \sum \lambda_i \vec{v}_i = q, \lambda_i \in \mathbb{N}, 1 \leq i \leq m \right\}. \quad (6)$$

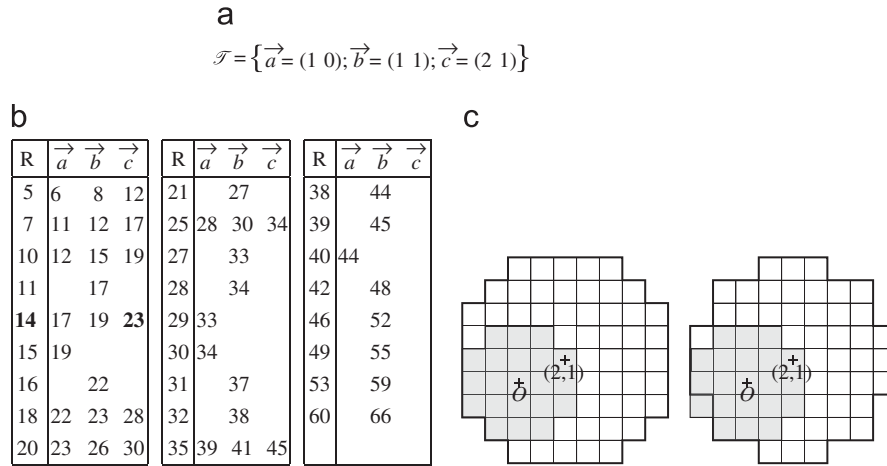
Paths between two points  $p$  and  $q$  can be produced by chaining displacements. The length of a path is the sum of the weights associated with the displacements and the distance between  $p$  and  $q$  is the length of the shortest path.

Any chamfer masks defines a metric [11]. However, a chamfer mask only generates a norm when proper conditions on the mask neighbors and on the corresponding weights permits a triangulation of the ball in influence cones [10,12]. When a mask induces a norm then all its balls are convex and therefore can be represented as polytopes.

### 2.3. Chamfer medial axis

For simple distances  $d_4$  and  $d_8$ , the medial axis extraction can be performed by the detection of local maxima in the distance map [1]. Chamfer distances raise a first complication even for small masks as soon as the weights are not unitary. Since all possible values of distance are not achievable, two different radii  $r$  and  $r'$  may correspond to the same set of discrete points. The radii  $r$  and  $r'$  are said to be *equivalent*. Since the distance transform labels pixels with the greatest equivalent radius, criteria based on radius difference fail to recognize equivalent disks as being covered by other disks. In the case of  $3 \times 3$  2D masks or  $3 \times 3 \times 3$  3D masks, a simple relabeling of distance map values with the smallest equivalent radius is sufficient [13,14]. However, this method fails for greater masks and the most general method for medial axis extraction from the distance map involves LUT that represent for each neighbor  $\vec{v}_i$  in a set called the test neighborhood  $\mathcal{T}$  and for each radius  $r_1$ , the minimal open





**Fig. 1.** (a) Test neighborhood and (b) lookup table for distance  $d_{5,7,11}$  extracted from [7, Fig. 14]. The value  $\text{Lut}_{\vec{c}}(14) = 23$  (in boldface) means that  $B_{\leq}(O, 14) \subseteq B_{\leq}(O + \vec{c}, 23)$  as shown in (c) but  $B_{\leq}(O, 14) \not\subseteq B_{\leq}(O + \vec{c}, 22)$  because of the point  $(-2, 1)$  (d). In terms of closed balls,  $B_{\leq}(O, 13) \subseteq B_{\leq}(O + \vec{c}, 22)$  but  $B_{\leq}(O, 13) \not\subseteq B_{\leq}(O + \vec{c}, 21)$ .

ball covering  $B_{\leq}(O, r_1)$  in direction  $\vec{v}_i$  [6]:

$$\text{Lut}_{\vec{v}_i}(r_1) = \min\{r_2 : B_{\leq}(O + \vec{v}_i, r_1) \subseteq B_{\leq}(O, r_2)\}.$$

Equivalently, using closed balls (considering (2)):

$$\text{Lut}_{\vec{v}_i}(r_1) = 1 + \min\{r_2 : B_{\leq}(O + \vec{v}_i, r_1 - 1) \subseteq B_{\leq}(O, r_2)\}. \quad (7)$$

As noticed by Thiel, the test neighborhood  $\mathcal{F}$  (Fig. 1(a)) is not necessarily equal to the chamfer mask  $\mathcal{M}$  [10].

### 2.3.1. Medial axis LUT coefficients

A general method for LUT coefficient computation was given by Rémy and Thiel [10, 15, 16]. The idea is that the disk covering relation can be extracted directly from values of distance to the origin. If  $d(O, p) = r_1$  and  $d(O, p + \vec{v}_i) = r_2$ , we can deduce the following:

$$p \in B_{\leq}(O, r_1) = B_{\leq}(O, r_1 + 1),$$

$$p + \vec{v}_i \notin B_{\leq}(O, r_2 - 1) = B_{\leq}(O, r_2),$$

hence  $B_{\leq}(O + \vec{v}_i, r_1 + 1) \not\subseteq B_{\leq}(O, r_2)$  and  $\text{Lut}_{\vec{v}_i}(r_1 + 1) > r_2$ . If  $\forall p, d(O, p) \leq r_1 \Rightarrow d(O, p + \vec{v}_i) \leq r_2$  then  $\text{Lut}_{\vec{v}_i}(r_1 + 1) = r_2 + 1$ .

Finally,  $\text{Lut}_{\vec{v}_i}(r) = 1 + \max\{d(O, p + \vec{v}_i) : d(O, p) < r\}$ .

This method only requires one scan of the distance function for each displacement  $\vec{v}_i$ . Moreover, the visited area may be restricted according to the symmetries of the chamfer mask. The order of complexity is about  $O(mL^n)$  for  $m$  neighbors if we limit the computation of the distance function to an image of size  $L^n$ .

### 2.3.2. Medial axis test neighborhood

Thiel observed that the chamfer mask is not adequate to compute the medial axis [15, p. 81]. For instance, with  $d_{14,20,31,44}$ ,  $\text{Lut}_{(2,1)}(291) = 321$  and  $\text{Lut}_{(2,1)}(321) = 352$ , but the smallest open ball of center  $O$  that covers  $B_{\leq}((4, 2), 291)$  is  $B_{\leq}(O, 351)$ . This inclusion relation is neither detected with the vector  $\vec{c} = (2\ 1)$  nor with the other vectors of the chamfer mask. Remy and Thiel then introduced an LUT mask (called test neighborhood and denoted by  $\mathcal{F}(R)$  here) for that purpose [12].  $\mathcal{F}(R)$  is the minimal set of vectors sufficient to detect the medial axis for shapes whose inner radius (the radius of a greatest ball) is less than or equal to  $R$ . In the previous  $d_{14,20,31,44}$  example, the point  $(4, 2)$  is not in the chamfer mask but should be in  $\mathcal{F}(R)$  for  $R$  greater than 350.

A test neighborhood incompleteness produces extra points in the medial axis (undetected ball coverings). A general method for both detecting and validating  $\mathcal{F}$  is based on the computation of the medial axis of all disks [7]. When  $\mathcal{F}$  is complete, the medial axis is restricted to the center of the disk, when extra points remains, they are added to  $\mathcal{F}$ . This neighborhood determination was proven to work in any dimension  $n \geq 2$ . However, it is time consuming even when taking advantage of the mask symmetries.

## 3. $\mathcal{H}$ -polytopes and chamfer balls

### 3.1. General $\mathcal{H}$ -polytopes [17]

**Definition 6 (Polyhedron).** A convex polyhedron is the intersection of a finite set of half-hyperplanes.

**Definition 7 (Polytope).** A polytope is the convex hull of a finite set of points.

**Theorem 1 (Weyl–Minkowski).** A subset of Euclidean space is a polytope if and only if it is a bounded convex polyhedron.

As a result, a polytope in  $\mathbb{R}^n$  can be represented either as the convex hull of its  $k$  vertices ( $\mathcal{V}$ -representation) or by a set of  $l$  half-planes ( $\mathcal{H}$ -representation):

$$P = \text{conv}(\{p_i\}_{1 \leq i \leq k}) = \left\{ p = \sum_{i=1}^k \alpha_i p_i : \alpha_i \in \mathbb{R}_+ \text{ and } \sum_{i=1}^k \alpha_i = 1 \right\}, \quad (8)$$

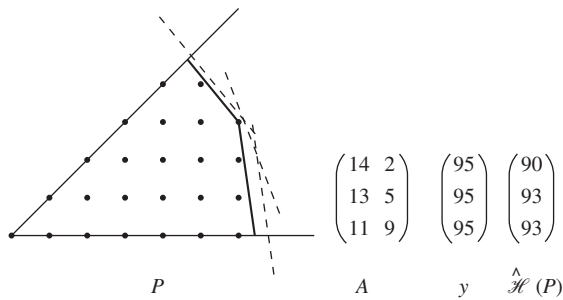
$$P = \{x : Ax \leq y\}, \quad (9)$$

where  $A$  is an  $l \times n$  matrix,  $y$  a vector of  $n$  values that we name  $\mathcal{H}$ -coefficients of  $P$ . Given two vectors  $\vec{u}$  and  $\vec{v}$ , we denote  $\vec{u} \leq \vec{v}$  if and only if  $\forall i, u_i \leq v_i$ .

**Definition 8 (Simplicial cone).** A simplicial cone  $C_{o,U}$  from a point  $o$  is a cone of dimension  $m$  defined by a set  $U$  of  $m$  independent vectors.

In a simplicial cone, each point is representable by a unique (up to a permutation) non-negative combination of the vectors of the cone, i.e.:

$$p \in C_{o,U} \Rightarrow \vec{op} = \sum_{i=1}^m \alpha_i \vec{u}_i \quad \text{where } (\alpha_i) \in \mathbb{R}_+^m \text{ is unique.}$$



**Fig. 2.**  $\mathcal{H}$ -representations of a discrete G-symmetrical polytope  $P$  (restricted to the first octant). Dashed lines: an  $\mathcal{H}$ -representation of  $P$ . Thick lines:  $\widehat{\mathcal{H}}$ -representation of  $P$ . In the  $\widehat{\mathcal{H}}$  case, the three equalities are verified for the same point  $(6, 3)$ . Notice that although coefficient values are minimal, this representation is still redundant: the second inequality could be removed.

**Definition 9 (Discrete polytope).** A discrete polytope  $\mathcal{Q}$  is the intersection of a polytope  $\mathcal{P}$  in  $\mathbb{R}^n$  with  $\mathbb{Z}^n$  (Gauss discretization of  $\mathcal{P}$ ).

**Definition 10 (Unimodular cone).** A unimodular cone  $C_{o,U}$  is a simplicial cone defined by a set of integral vectors  $U$  that generate all the integral points of  $C_{o,U}$ .

In a unimodular cone, all discrete points are representable as a unique non-negative integral combination of the vectors of the cone. A cone  $C_{o,U}$  of dimension  $n$  is unimodular if and only if  $|\det(U)| = 1$ . Then  $U$  is the basis of a unimodular point lattice, equivalent to  $\mathbb{Z}^n$ , so each integral point can be reached.

### 3.1.1. Minimal representation

Many operations on  $\mathbb{R}^n$  polytopes in either  $\mathcal{V}$  or  $\mathcal{H}$  representation often require a minimal representation. The *redundancy removal* is the elimination of unnecessary points in the  $\mathcal{V}$ -representation or unnecessary inequalities in the  $\mathcal{H}$ -representation of polytopes. Since our purpose is mainly to compare  $\mathcal{H}$ -polytopes defined with the same matrix  $A$ , no inequality removal is needed. However, for some operations,  $\mathcal{H}$ -representations of discrete polytopes must be minimal in terms of  $\mathcal{H}$ -coefficients.

**Definition 11 (Minimal parameter representation).** A minimal parameter  $\mathcal{H}$ -representation of a discrete polytope  $P$ , denoted  $\widehat{\mathcal{H}}$ -representation, is an  $\mathcal{H}$ -representation of  $P = \{x : Ax \leq y\}$  such that  $y$  is minimal:

$$P = \{x \in \mathbb{Z}^n : Ax \leq y\} \text{ and } \forall i \in [1 \dots l], \exists x \in P : A_i x = y_i, \quad (10)$$

where  $A_i$  stands for the  $i$ th line of the matrix  $A$ .

The  $\widehat{\mathcal{H}}$  function, introduced for convenience, gives the minimal parameter vector for a given polytope  $P$ :  $\widehat{\mathcal{H}}(P) = \max\{Ax : x \in P\}$ . As a consequence,  $\{x : Ax \leq \widehat{\mathcal{H}}(P)\}$  is the  $\widehat{\mathcal{H}}$ -representation of  $P = \{x : Ax \leq y\}$ . Fig. 2 depicts two representations of the same polytope  $P$  in  $\mathbb{Z}^2$ , one of each is minimal.

### 3.1.2. $\mathcal{H}$ -polytope translation

Let  $P = \{x : Ax \leq y\}$  be an  $\mathcal{H}$ -polytope. The translated of  $P$  by  $\vec{v}$  which is also the Minkowski sum of  $P$  and  $\{\vec{v}\}$  is

$$(P)_{\vec{v}} = P \oplus \{\vec{v}\} = \{x + \vec{v} : Ax \leq y\} = \{x : Ax \leq y + A\vec{v}\}. \quad (11)$$

The translation of a minimal representation gives a minimal representation.

### 3.1.3. Covering test

Let  $P = \{x : Ax \leq y\}$  and  $Q = \{x : Ax \leq z\}$  be two polyhedra represented by the same matrix  $A$  but having different sets of  $\mathcal{H}$ -coefficients  $y$  and  $z$ . We have

$$y \leq z \Rightarrow P \subseteq Q. \quad (12)$$

Furthermore, if the  $\mathcal{H}$ -description of the enclosed polyhedron has minimal coefficients, the condition is also necessary:

$$y = \widehat{\mathcal{H}}(P) \leq z \Leftrightarrow P \subseteq Q. \quad (13)$$

## 3.2. Geometry of the chamfer ball [10,12]

Describing balls of chamfer norms as polygons in 2D and polyhedra in higher dimensions is not new [11]. Thiel and others have extensively studied chamfer ball geometry from this point of view [12,15,18]. Our purpose is to introduce properties specific to the  $\mathcal{H}$ -representation of these convex balls.

We can deduce from (1) and (6) a recursive construction of chamfer balls:

$$B_{\leq}(O, r) = B_{\leq}(O, r-1) \cup \bigcup_{0 \leq i \leq m} B_{\leq}(O + \vec{v}_i, r - w_i). \quad (14)$$

This construction is sufficient to obtain an exact description of the balls in the case of chamfer norms as we will see later (Section 5.1).

**Definition 12 (Rational ball).** Consider a chamfer mask  $\mathcal{M}$ , the rational unitary ball or simply rational ball  $B_R$  is the convex hull of the rational points  $\vec{v}_i/w_i$ ,  $(\vec{v}_i, w_i) \in \mathcal{M}$ :

$$B_R = \text{conv} \left( \frac{\vec{v}_i}{w_i} : (\vec{v}_i, w_i) \in \mathcal{M} \right) = \left\{ \sum_{i=1}^m \alpha_i \frac{\vec{v}_i}{w_i} : \alpha_i \in \mathbb{R}_+, \sum_{i=1}^m \alpha_i = 1 \right\}. \quad (15)$$

Note that the rational ball, as it is introduced here, is convex by definition and is different, in this regard, from the *equivalent rational ball* described in [10,12,19].

**Proposition 1.** The homothetic of  $B_R$ ,  $rB_R$  contains  $B_{\leq}(O, r)$ :  $B_{\leq}(O, r) \subseteq rB_R$ .

**Proof.** Let  $p$  be a point of  $B_{\leq}(O, r)$  and  $\sum_{i=1}^m \lambda_i \vec{v}_i$ ,  $\lambda_i \in \mathbb{N}$  be a minimal path between  $O$  and  $p$ . Then  $d(O, p) \leq r$  and  $d(O, p) = \sum_{i=1}^m \lambda_i w_i$ . We can describe  $p$  as a combination of  $O$ ,  $r\vec{v}_1/w_1, \dots, r\vec{v}_n/w_n$ :

$$p = \frac{r-d(O,p)}{r} \cdot O + \sum_{i=1}^m \frac{\lambda_i w_i}{r} \cdot \frac{r}{w_i} \vec{v}_i \quad \text{with } \frac{r-d(O,p)}{r} \geq 0, \frac{\lambda_i w_i}{r} \geq 0 \forall i$$

and

$$\frac{r-d(O,p)}{r} + \sum_{i=1}^m \frac{\lambda_i w_i}{r} = 1.$$

Then, by definition of  $\text{conv}$ :  $p \in \text{conv}(O, (r/w_1)\vec{v}_1, \dots, (r/w_n)\vec{v}_n) \subset rB_R$  and every point of  $B_{\leq}(O, r)$  is in  $rB_R$ .  $\square$

**Definition 13 (Normal vector, Thiel [10]).** We call normal vector to a facet  $\mathcal{F}$  of  $B_R$ , the unique vector  $\vec{\mathcal{F}}$  orthogonal to the facet such that  $\forall p \in \mathcal{F}, \vec{\mathcal{F}} \cdot p = 1$ .

In the simplicial cone spanned by vectors  $\vec{v}_1/w_1 \dots \vec{v}_n/w_n$ ,  $\vec{\mathcal{F}} = (\vec{v}_1/w_1 \dots \vec{v}_n/w_n) = (1 \dots 1)$ . This implies

$$\vec{\mathcal{F}} = (w_1, \dots, w_n) \cdot (\vec{v}_1 \dots \vec{v}_n)^{-1}. \quad (16)$$



The vector  $\alpha = (\vec{v}_1 | \dots | \vec{v}_n)^{-1} p$  give the unique representation of  $p$  as a combination of  $\vec{v}_1, \dots, \vec{v}_n$ . If  $\alpha$  has integer coefficients then there is a path  $\alpha(\vec{v}_1 | \dots | \vec{v}_n)$  to  $p$  and its length is  $(w_1, \dots, w_n) \cdot \alpha = \vec{\mathcal{F}} \cdot p$ . The normal vector is equivalent to the elementary displacement and discrete gradient of the cone intercepted by  $\mathcal{F}$  as defined for chamfer norms in [10,12]. Note that the normal vector is defined here for general chamfer masks, whether they induce a norm or not.

For instance, with the chamfer norm  $d_{5,7,11}$ , the point  $(3, 1)$  is in the cone spanned by the vectors  $\vec{u} = (1 \ 0)$  and  $\vec{v} = (2 \ 1)$  and the weights involved are 5 and 11. The distance between the origin and the point  $(3, 1)$  is then [10, (4.32)]:

$$d(O, (3, 1)) = (5 \ 11) \cdot \begin{pmatrix} 1 & 2 \\ 0 & 1 \end{pmatrix}^{-1} \cdot \begin{pmatrix} 3 \\ 1 \end{pmatrix} = (5 \ 1) \cdot \begin{pmatrix} 3 \\ 1 \end{pmatrix} = 16.$$

### 3.3. Mask conditions for chamfer norms

**Proposition 2** (Distance lower bound). *If  $\mathcal{F}$  is a facet of the rational ball and  $\vec{\mathcal{F}}$  its normal vector, then:*

$$d(O, p) \geq \vec{\mathcal{F}} \cdot p. \tag{17}$$

**Proof.** Let  $p = \sum_i \lambda_i \vec{v}_i$  be a minimal path to  $p$ , so  $d(O, p) = \sum_i \lambda_i w_i$ . The facet  $\mathcal{F}$  is supported by the hyperplane  $\{x \in \mathbb{R}^n : \vec{\mathcal{F}} \cdot x = 1\}$ . Due to convexity, the unitary ball is included in the half-space  $\{x \in \mathbb{R}^n : \vec{\mathcal{F}} \cdot x \leq 1\}$ . This applies to the vertices of the unitary ball  $\vec{v}_i/w_i$ :  $\vec{\mathcal{F}} \cdot \vec{v}_i/w_i \leq 1$ . By linearity of the dot product,  $\vec{\mathcal{F}} \cdot \vec{v}_i \leq w_i$  and  $\vec{\mathcal{F}} \cdot (\sum_i \lambda_i \vec{v}_i) = \sum_i \lambda_i \vec{\mathcal{F}} \cdot \vec{v}_i \leq \sum_i \lambda_i w_i$ . Hence,  $\vec{\mathcal{F}} \cdot \sum_i \lambda_i \vec{v}_i$  is always less than or equal to the length of the path  $\sum_i \lambda_i \vec{v}_i$ .  $\square$

This lower bound applies for any point  $p$ , whether or not it is located in a cone that intercepts  $\mathcal{F}$ ;  $d$  does not need to induce a norm.

**Proposition 3** (Homogeneity in a unimodular cone). *Let  $C_{O,U}$  be a unimodular cone that intercepts a unique facet  $\mathcal{F}$  of  $B_R$ . The distance  $d$  is homogenous in the cone and equal to  $\vec{\mathcal{F}} \cdot p$ ,  $\forall p \in C_{O,U}$  if and only if:*

$$d(O, O + \vec{u}) = \vec{\mathcal{F}} \cdot \vec{u}, \quad \forall \vec{u} \in U. \tag{18}$$

**Proof.** According to Proposition 2,  $d(O, p) \geq \vec{\mathcal{F}} \cdot p$ ,  $\forall p \in C_{O,U}$ . Since  $C_{O,U}$  is unimodular,  $p$  is reachable by a non-negative integral combination of vectors of  $U$ ,  $\vec{p} = \sum \lambda_i \vec{u}_i$ . By triangle inequality and composition hypotheses,  $d(O, p) \leq \sum \lambda_i d(O, O + \vec{u}_i) = \sum \lambda_i \vec{\mathcal{F}} \cdot \vec{u}_i = \vec{\mathcal{F}} \cdot p$ . So  $\vec{\mathcal{F}} \cdot p$  is simultaneously a lower and upper bound to  $d(O, p)$  then the equality holds and  $d$  is homogenous in  $C_{O,U}$ .

Conversely, consider a vector  $\vec{u}$  of  $U$  such that  $d(O, p) \neq p \cdot \vec{\mathcal{F}}$ , so  $d(O, p) > p \cdot \vec{\mathcal{F}}$ , by Proposition 2. In the facet  $\mathcal{F}$ , we always can find  $n = \dim(\mathcal{F}) + 1$  vertices  $\vec{v}_1/w_1 \dots \vec{v}_n/w_n$  such that  $\vec{u}$  lies in the simplicial cone spanned by the vectors  $\vec{v}_1 \dots \vec{v}_n$ . As a consequence  $\vec{u}$  can be represented by a unique non-negative combination of  $n$  linearly independent vectors:  $\vec{u} = \alpha \cdot (\vec{v}_1 | \dots | \vec{v}_n)$ . The components of  $\alpha = (\vec{v}_1 | \dots | \vec{v}_n)^{-1} \cdot \vec{u}$  are not necessarily integers (they are if  $\Delta = |\det(\vec{v}_1 | \dots | \vec{v}_n)| = 1$ ). However,  $\Delta(\vec{v}_1 | \dots | \vec{v}_n)^{-1}$  has integral coefficients hence  $\Delta \vec{u}$  can be uniquely represented as an integral combination of the vectors  $\vec{v}_i$ :  $\Delta \vec{u} = \Delta(\vec{v}_1 | \dots | \vec{v}_n) \cdot \alpha$ , so there is a path to  $O + \Delta \vec{u}$  whose length is:  $\Delta(w_1 \dots w_n) \cdot \alpha = \Delta(w_1 \dots w_n) \cdot (\vec{v}_1 | \dots | \vec{v}_n)^{-1} \cdot \vec{u} = \Delta \vec{\mathcal{F}} \cdot p \geq d(O, \Delta \vec{u})$ . Thus  $d(O, O + \Delta \vec{u}) < \Delta d(O, O + \vec{u})$  and the distance is not homogenous in the cone.  $\square$

According to this proposition, the homogeneity in an  $nD$  unimodular cone can be determined by only testing the vectors of the cone (which are not necessarily in the chamfer mask). We derive a

necessary and sufficient condition for norms (a sufficient condition, based on a unimodular triangulation, is given in [10]):

**Corollary 1** (Norm condition). *Let  $\{U_i\}$  be a set of vectors sets  $U_i$  with the following properties:*

- each unimodular cone  $C_i = C_{O,U_i}$  intersects a unique facet of  $B_R$ ;
- the union of the cones covers the entire space  $\mathbb{Z}^n$ .

The chamfer mask  $\mathcal{M}$  induces a norm if and only if:

$$d(O, O + \vec{u}) = \vec{\mathcal{F}} \cdot \vec{u}, \quad \forall \vec{u} \in \bigcup_i U_i. \tag{19}$$

**Proposition 4** (Direct distance formulation). *If  $d$  induces a norm, the distance from  $O$  to any point  $p$  is*

$$d(O, p) = \max_{1 \leq i \leq l} \{\vec{\mathcal{F}}_i \cdot p\}, \tag{20}$$

where  $l$  is the number of facets of  $B_R$  and  $\vec{\mathcal{F}}_i$  is the normal vector to the  $i$ th facet of  $B_R$ . This formula does not require to determine in which cone lies  $p$ .

**Proof.** The inequality  $d(O, p) \geq \vec{\mathcal{F}}_i \cdot p$  holds for all facets due to Proposition 2 and the equality  $d(O, p) = \vec{\mathcal{F}}_i \cdot p$  holds, according to Proposition 3, for all the facets of  $B_R$  (there is at least one of them) that intersect  $[O, p]$ .  $\square$

### 3.4. $\mathcal{H}$ -representation of chamfer norm balls

The  $\mathcal{H}$ -representation of chamfer balls is directly derived from (20):

$$p \in B_{\leq}(O, r) \Leftrightarrow \max_{1 \leq i \leq l} \{\vec{\mathcal{F}}_i \cdot p\} \leq r \Leftrightarrow A_{\mathcal{M}} \cdot p \leq \widehat{y}, \tag{21}$$

where  $A_{\mathcal{M}}$  is an  $\mathcal{H}$ -representation matrix depending only on the chamfer mask  $\mathcal{M}$ . The number of rows in  $A_{\mathcal{M}}$  is equal to the number  $l$  of facets of  $B_R$ , each line of the matrix  $A_{\mathcal{M}}$  is computed with (16) and  $y$  is a column vector whose values are  $r$ . For instance, the  $\mathcal{H}$ -representation matrix of  $d_{5,7,11}$  balls restricted to  $\mathcal{G}$  is  $A_{\mathcal{M}} = \begin{pmatrix} 5 & 1 \\ 4 & 3 \end{pmatrix}$  where  $(5 \ 1)$  and  $(4 \ 3)$  are the normal vectors of the two facets of  $B_R$  and  $B_{\leq}(O, r) = \{p \in \mathbb{Z}^n : \begin{pmatrix} 5 & 1 \\ 4 & 3 \end{pmatrix} \cdot p \leq \begin{pmatrix} r \\ r \end{pmatrix}\}$ .

Note that if a polytope is not centered in  $O$ , the simplification due to symmetries do not hold and the full set of  $\mathcal{H}$ -coefficients is needed, unless we ensure that the  $\mathcal{H}$ -coefficients for the hyperplanes in the generator  $\mathcal{G}$  are greater than  $\mathcal{H}$ -coefficients for the corresponding symmetric cones. This is the case when a G-symmetric polytope is translated by a vector in  $\mathcal{G}$ .

**Proposition 5** (Furthest point). *Let  $A_{\mathcal{M}}$  be the matrix defined by a chamfer mask  $\mathcal{M}$  generating a norm. The furthest point from the origin in the  $\mathcal{H}$ -polytope  $P = \{p : A_{\mathcal{M}} \cdot p \leq \widehat{\mathcal{H}}(P)\}$  is at a distance equal to the greatest component of  $\widehat{\mathcal{H}}(P)$ .*

**Proof.** By construction of  $A_{\mathcal{M}}$ , (20) is equivalent to  $d(O, p) = \max_i \{A_{\mathcal{M}_i} \cdot p\}$ , thus

$$\max_{p \in P} \{d(O, p)\} = \max_{p \in P} \left\{ \max_{1 \leq i \leq l} \{A_{\mathcal{M}_i} \cdot p\} \right\} = \max_{1 \leq i \leq l} \{\widehat{\mathcal{H}}_i(P)\}. \quad \square$$

**Proposition 6** (Minimal covering ball). *The radius of the minimal ball centered in  $O$  that contains all points of a discrete  $\mathcal{H}$ -polytope  $P$  represented by the matrix  $A_{\mathcal{M}}$  and the vector  $\widehat{\mathcal{H}}(P)$  is equal to the greatest component of  $\widehat{\mathcal{H}}(P)$ .*

**Proof.** The smallest ball that covers the polytope  $P$  must contain its furthest point from the origin:

$$\min\{r \in \mathbb{N} : P \subseteq B_{\leq}(O, r)\} = \max_{p \in P} \{d(O, p)\} = \max_{1 \leq i \leq l} \{\widehat{\mathcal{H}}_i(P)\}. \quad \square$$

**Definition 14** (Covering function). We call covering function of a set  $X$  of points of  $\mathbb{Z}^n$  the function  $\mathcal{C}_X$  which assigns to each point  $p$  of  $\mathbb{Z}^n$ , the radius of the minimal ball centered in  $p$  covering  $X$ :

$$\mathcal{C}_X : \mathbb{Z}^n \rightarrow \mathbb{N},$$

$$\mathcal{C}_X(p) = \min\{r : X \subseteq B_{\leq}(p, r)\}.$$

The covering function of the chamfer ball  $B_{\leq}(O, r)$  at point  $p$  gives the radius of the minimal ball centered in  $p$  that contains  $B_{\leq}(O, r)$  or, by central symmetry of the chamfer balls, to the radius of the minimal ball centered in  $O$  covering  $B_{\leq}(p, r)$  and therefore it is the maximal component of the  $\widehat{\mathcal{H}}$ -representation of  $B_{\leq}(p, r)$ :

$$\mathcal{C}_{B_{\leq}(O, r)}(p) = \max\{\widehat{\mathcal{H}}(B_{\leq}(p, r))\} = \max\{\widehat{\mathcal{H}}(B_{\leq}(O, r)) + A_{\mathcal{M}} \cdot p\}. \quad (22)$$

One can notice that the covering function of the zero radius disk is equal to the distance function, as is the distance transform of the complement of this disk:

$$\mathcal{C}_{B_{\leq}(O, 0)}(p) = \text{DT}_{\mathbb{Z}^n \setminus \{0\}}(p) = d(p, O) = d(O, p).$$

**Definition 15** (Covering cone). A covering cone  $C_{o,U}$  in the covering function  $\mathcal{C}_X$  is a unimodular cone defined by a vertex  $o$  and a set of vectors  $U$ , such that each ball centered in  $p$  is included in the ball centered in  $p + \vec{u}$ ,  $\vec{u} \in U$ :

$$\forall p \in C_{o,U}, \forall \vec{u} \in U, B_{\leq}(p, \mathcal{C}_X(p)) \subseteq B_{\leq}(p + \vec{u}, \mathcal{C}_X(p + \vec{u})).$$

**Proposition 7.** If  $C_{o,U}$  is a covering cone in  $\mathcal{C}_{B_{\leq}(O, r)}$  then for any point  $q$  in  $C_{o,U} \setminus U \setminus \{o\}$  there exists  $p$  such that the ball centered in  $q$  includes the ball centered in  $p$ :

$$B_{\leq}(O, r) \subseteq B_{\leq}(p, \mathcal{C}_{B_{\leq}(O, r)}(p)) \subseteq B_{\leq}(q, \mathcal{C}_{B_{\leq}(O, r)}(q)).$$

**Proof.** If  $U$  is not empty, there is  $\vec{u} \in U$  such that  $p = q - \vec{u}$  is in  $C_{o,U} \setminus \{o\}$ . Then, by definition of  $C_{o,U}$ ,  $B_{\leq}(p, \mathcal{C}_{B_{\leq}(O, r)}(p)) \subseteq B_{\leq}(q, \mathcal{C}_{B_{\leq}(O, r)}(q))$ . In addition,  $B_{\leq}(O, r) \subseteq B_{\leq}(p, \mathcal{C}_{B_{\leq}(O, r)}(p))$  always holds by definition of  $\mathcal{C}_{B_{\leq}(O, r)}$ .  $\square$

Proposition 7 is used to limit the search for balls directly covering  $B_{\leq}(O, r)$  in the covering function  $\mathcal{C}_{B_{\leq}(O, r)}$ . When a covering cone is detected, only its vertex  $o$  or the vectors of the cone (if  $o = O$ ) have to be tested.

**Proposition 8.** If there is an integer  $j \in [1 \dots l]$  and a point  $o$  such that  $A_{\mathcal{M}j} \cdot \vec{u}_i, \forall \vec{u}_i \in U$  and  $\widehat{\mathcal{H}}_j(B_{\leq}(o, r))$  are maximal then  $C_{o,U}$  is a covering cone in  $\mathcal{C}_{B_{\leq}(O, r)}$ .

**Proof.** Let  $j$  be the row number of a maximal component of  $\widehat{\mathcal{H}}(B_{\leq}(o, r))$  and  $A_{\mathcal{M}j} \cdot \vec{u}_i, \forall \vec{u}_i \in U$ , then  $j$  is a maximal component of any positive linear combination of these vectors. Let  $p$  be any point in  $C_{o,U}$ ,  $p = o + \sum_i \lambda_i \vec{u}_i, \lambda_i \in \mathbb{N}$ . From (22), we deduce that, in  $C_{o,U}$ ,  $\mathcal{C}_{B_{\leq}(O, r)}$  takes the form of an affine function of the components  $\lambda_i$ :

$$\mathcal{C}_{B_{\leq}(O, r)}(p) = \max \left\{ \widehat{\mathcal{H}}(B_{\leq}(o, r)) + \sum \lambda_i A_{\mathcal{M}j} \vec{u}_i \right\}$$

$$= \widehat{\mathcal{H}}_j(B_{\leq}(o, r)) + \sum \lambda_i A_{\mathcal{M}j} \vec{u}_i.$$

In the same way, the radius of the minimal ball centered in  $p + \vec{u}$ ,  $\vec{u} \in U$  covering  $B_{\leq}(O, r)$  is

$$\mathcal{C}_{B_{\leq}(O, r)}(p + \vec{u}) = \widehat{\mathcal{H}}_j(B_{\leq}(o, r)) + \sum \lambda_i A_{\mathcal{M}j} \vec{u}_i + A_{\mathcal{M}j} \vec{u}$$

$$= \mathcal{C}_{B_{\leq}(p, \mathcal{C}_{B_{\leq}(O, r)}(p))}(O + \vec{u}).$$

In other words,  $B_{\leq}(p, \mathcal{C}_{B_{\leq}(O, r)}(p))$  is contained in  $B_{\leq}(p + \vec{u}, \mathcal{C}_{B_{\leq}(O, r)}(p + \vec{u}))$  and  $C_{o,U}$  is a covering cone according to Definition 15.  $\square$

#### 4. Triangulation of 3D chamfer balls

The first stage of the method consists in decomposing the generator cone  $\mathcal{G}$  in unimodular cones that each intercepts a unique facet of the rational ball. This triangulation has three goals:

1. compute the distance gradients which are vectors normal to the facets of the rational ball. These distance gradients are the rows of the matrix  $A_{\mathcal{M}}$ ;
2. provide a set of unimodular cones suitable for the recursive partition of the generator  $\mathcal{G}$  used in the next stage;
3. verify that the chamfer mask generates a norm by checking homogeneity in each cone.

##### 4.1. Convex hull vertices in the generator cone

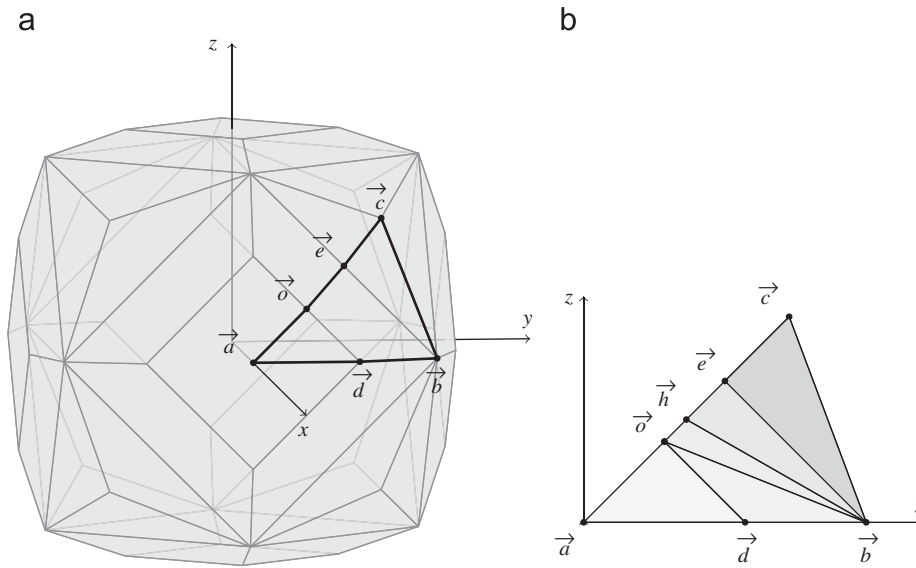
The expected result is a triangulation of the part of the rational ball  $B_R$  that lies in the generator  $\mathcal{G}$ . The input data consist of vectors from the chamfer mask  $\mathcal{M}|_{\mathcal{G}}$  divided by their corresponding weights (to form the equivalent unitary ball). In the algorithm, these points in  $\mathbb{Q}^n$  are represented as weightings, and all computations are based on integer arithmetics. The weightings of the chamfer mask are not sufficient alone because

$$\mathcal{G} \cap \text{conv} \left( \frac{\vec{v}}{w}, (\vec{v}; w) \in \mathcal{M} \right) \neq \text{conv} \left( \frac{\vec{v}}{w}, (\vec{v}; w) \in \mathcal{M}|_{\mathcal{G}} \right).$$

The intersection of  $B_R$  with  $\mathcal{G}$  introduces new vertices that are points in edges of  $B_R$  that intersect the faces of  $\mathcal{G}$ . This is illustrated in Fig. 3(a); vectors  $\vec{e} = (2 \ 1 \ 1)$  and  $\vec{d} = (4 \ 1 \ 1)$  are not in the chamfer mask and thus cannot be vertices of  $B_R$ , but they appear in the intersection of  $B_R$  with  $(O, \{\vec{d}, \vec{e}\})$ . Since the edges of  $B_R$  are not known yet, all the intersections of the faces of  $\mathcal{G}$  with couples of rational vertices of  $B_R$  must be added to the input of the algorithm. These points are rational combinations of the vertices of the intersected edges, but are also represented as weightings with integer coefficients. Among those, the majority of the points that lie in the interior of  $B_R$  will be discarded during the computation of the convex hull.

##### 4.2. Gift-wrapping convex hull and triangulation of the rational ball

Many different algorithms are available for convex hull computation. We chose to implement an modified version of the algorithm known as *gift wrapping* [20] for two reasons: first, it naturally produces a triangulation of the convex hull i.e. a partition in simplicial cones and second, a subpartition of these simplicial cones in unimodular cones and the face lattice of the cones can be easily integrated as by-products of the algorithm.



**Fig. 3.** The rational ball  $B_R$  of the 3D chamfer mask 7,8,11,14 and a unimodular triangulation of  $B_R \cap \mathcal{G}$ . Unimodular triangulation in 3D cones (represented by triangles) of  $B_R \cap \mathcal{G}$  (thick lines in (a)). Each  $\mathbb{Z}^3$  point  $(x,y,z)$  is projected onto  $(y/x, z/x)$  (Farey space).

**Algorithm 1.** (Simplified) gift wrapping algorithm [20].

**Input:** A set of points  $S$

- 1 find a starting edge of the convex hull  $(p,q)$
- 2 push  $(p,q)$  in  $\ell$ , the list of active edges
- 3 **while**  $L$  is not empty **do**
- 4   pop  $(p,q)$  from  $\ell$
- 5   take  $r$  in  $S \setminus \{p,q\}$
- 6    $S' \leftarrow S \setminus \{p,q,r\}$
- 7   **for**  $s$  in  $S'$  **do**
- 8     **if**  $s$  is above  $(p,q,r)$  **then**
- 9        $r \leftarrow s$
- 10    **end**
- 11   **end**
- 12 add each newly created edge  $(r,p)$  and  $(q,r)$  in  $\ell$  or remove it from  $\ell$  if already contained
- 13 **end**

Starting with a first edge, the gift wrapping algorithm 1 iteratively extends the convex hull by appending a new triangle and its edges. Since the convex hull is computed in  $\mathcal{G}$ , one vertex of the starting edge can be chosen as the point in the first axis of  $\mathcal{G}$  that is closest to the origin:  $p = (x_1, 0, 0)$  (with maximal  $x_1$  if several points exist on this axis) and the second vertex is the point  $q$  that maximizes the angle  $(O,p,q)$  in a facet of  $\mathcal{G}$  containing  $p$ . The algorithm proceeds until exhaustion of active edges.

4.3. Unimodular triangulation

The gift wrapping algorithm produces a set of  $nD$  simplicial cones  $C_{o,U}$ , each of which is unimodular if and only if  $\det(U) = \pm 1$ . When  $|\det(U)| > 1$ , a new point is introduced as an input of the convex hull computation and the detected non-unimodular cone is discarded. The unimodular triangulation is represented by a cone lattice  $\mathcal{L}$  used later as a partition of  $\mathcal{G}$ .

The intersection of the rational ball for mask 7,8,11,14 and  $\mathcal{G}$  is pictured in Fig. 3(b). A unimodular triangulation without any new point of the face  $(\vec{b} = (1 \ 1 \ 0), \vec{d} = (2 \ 1 \ 0), \vec{e} = (2 \ 1 \ 1), \vec{o} = (4 \ 1 \ 1))$  cannot be made because  $\det(\vec{d}, \vec{e}, \vec{o}) = \det(\vec{b}, \vec{e}, \vec{o}) = 2$ . This is

solved by introducing the vector  $\vec{h}$ . The cone lattice corresponding to this unimodular triangulation is pictured in Fig. 4.

4.4. Homogeneity check

The unimodular triangulation of  $B_R$  produces a set of cones that cover the generator  $\mathcal{G}$  and their corresponding face normal vectors. Using Corollary 1, by comparing  $d(O, O + \vec{u})$ , obtained by propagating the chamfer weights, with  $\vec{\mathcal{F}}_i \cdot \vec{u}$ , where  $\vec{\mathcal{F}}_i$  is the normal vector of one of the faces that intersects  $[O, \vec{u}]$ , we determine if the distance is homogenous in  $\mathcal{G}$ , so by symmetry, in  $\mathbb{Z}^3$ .

5. LUT and test neighborhood computation for 3D chamfer norms

5.1.  $\widehat{\mathcal{H}}$ -representation of chamfer balls

The computation of the LUT is based on an  $\widehat{\mathcal{H}}$ -representation of the chamfer norm balls. All share the same matrix  $A_{\mathcal{M}}$  which depends only on the chamfer mask (21). The  $\widehat{\mathcal{H}}$ -coefficients of the balls are computed iteratively from the ball of radius 0,  $B_{\leq}(O,0) = \{x : A_{\mathcal{M}}x = 0\}$  using (14) and (11):

$$\widehat{\mathcal{H}}(B_{\leq}(O,r)) = \max\{\widehat{\mathcal{H}}(B_{\leq}(O,r-1)); \widehat{\mathcal{H}}(B_{\leq}(O,r-w_i)) + A_{\mathcal{M}} \vec{v}_i, i \in [1 \dots m]\}.$$

5.2. LUT values

LUT values are directly obtained from the covering function (22):

$$\begin{aligned} \text{Lut}_{\vec{v}_i}[r] &= 1 + \mathcal{C}_{B_{\leq}(O,r-1)}(O + \vec{v}_i) \\ &= 1 + \max\{\widehat{\mathcal{H}}(B_{\leq}(O,r-1)) + A_{\mathcal{M}} \cdot \vec{v}_i\}. \end{aligned}$$

5.3. Test neighborhood

Algorithm 2 begins with an empty test neighborhood  $\mathcal{T}$  and seek direct covering relations in balls of increasing radii like the method

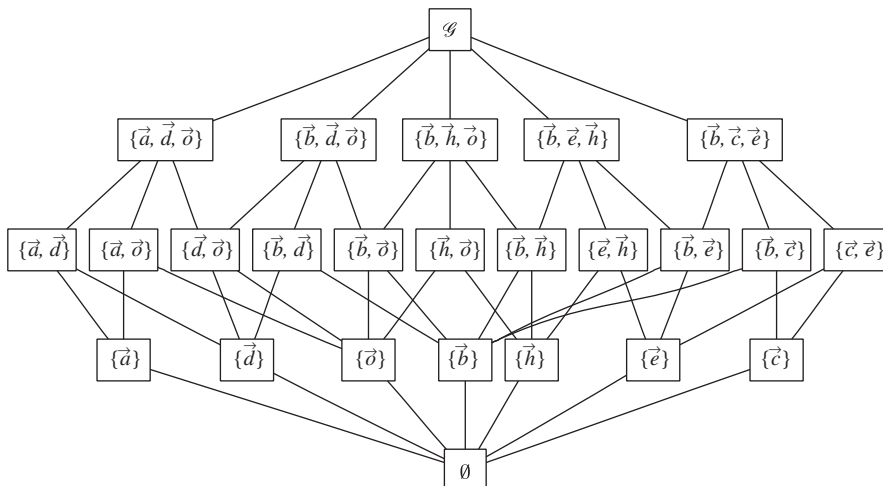


Fig. 4. Lattice of cone vector sets for the unimodular triangulation of  $B_R \cap \mathcal{G}$  (chamfer mask 7,8,11,14).

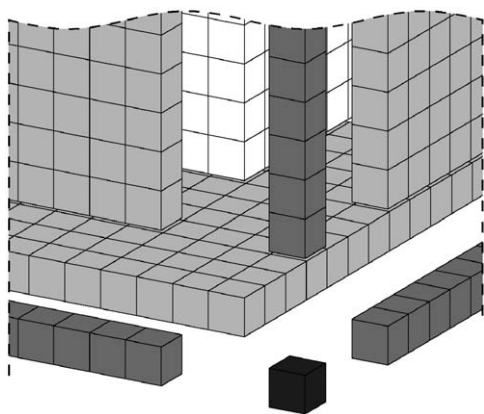


Fig. 5. The partition of a 3D unimodular cone in  $2^3$  subcones (from darkest to lightest): 1 point,  $C_1^3 = 3$  1D cones,  $C_2^3 = 3$  2D cones and  $C_3^3 = 1$  3D cone.

in [7]. However, in [7], the covering relations are seen from the perspective of the covering balls (in the distance map, requiring propagation of distances), whereas in our case, they are considered from the point of view of the covered balls, i.e. in the covering function that can be computed directly without propagation using the direct formula (22). The algorithm uses the lattice of cones  $\mathcal{L}$  (Fig. 4), resulting from the triangulation phase, as a partition of  $\mathcal{G}$ . Each cone that is not a covering cone is recursively partitioned by procedure `visitCone` as depicted in Fig. 5: a  $n$ D cone  $C_{o,U}$  is divided into  $2^n$  subcones corresponding to all the subsets of  $U$ . Direct covering relations at the vertices of all the visited cones are checked by procedure `visitPoint` using a direct computation of the covering function in all known neighbors in  $\mathcal{T}$ .

**Algorithm 2.** Computation of the test neighborhood.

```

Input:  $\mathcal{L}$  lattice of simplicial unimodular cones, maximal radius  $R$ 
1  $\mathcal{T} \leftarrow \emptyset$ 
2 for  $r \leftarrow 1$  to  $R$  do // inner radius
3   foreach  $C_{o,U} \in \mathcal{L}$  do // cones with increasing dimension
4      $o' \leftarrow o + \sum_{\vec{u} \in U} \vec{u}$ ;
5     visitCone( $\mathcal{T}, C_{o,U}, r$ );
6   end
7 end
    
```

**6. Results**

An implementation of these algorithms was developed in the C language. It produces output in the same format as the reference algorithm [7] so that outputs can be compared character-to-character. Tests were done on various chamfer masks and different maximal radii and helped discover a propagation issue in the reference algorithm. This problem corrected, the results are almost always identical except for insignificant cases close to the maximal radius for which covering radii exceed the maximum. These radii are handled differently by both algorithms but since they exceed the maximum radius, there is no impact on the medial axis computation.

**Procedure.** `visitCone`( $\mathcal{T}, C_{o,U}, r$ ).

```

Input:  $\mathcal{T}, C_{o,U}$ , inner radius  $r, y = \widehat{\mathcal{H}}(B_{\leq}(o, r))$ 
1 visitPoint( $o$ )
   // Visit subcones if  $C_{o,U}$  is not a covering cone
2 if  $\forall j, \forall \vec{u} \in U : y_j \neq \max_k \{y_k\}$  and  $A_{\mathcal{H}j} \cdot \vec{u} \neq \max_k \{A_{\mathcal{H}k} \cdot \vec{u}\}$  then
3   foreach subcone  $C_{o,U'}$  of  $C_{o,U}$  do
4      $o' \leftarrow o + \sum_{\vec{u} \in U'} \vec{u}$ ;
5     visitCone( $\mathcal{T}, C_{o',U'}, r$ );
6   end
7 end
    
```

**Procedure.** `visitPoint`( $\mathcal{T}, r, p$ ).

```

Input:  $\mathcal{T}$ , inner radius  $r$ , ball center  $p$  to test
1 if  $\forall \vec{v} \in \mathcal{T} : B_{\leq}(O + \vec{v}, \mathcal{C}_{B_{\leq}(O,r)}(O + \vec{v})) \not\subset B_{\leq}(p, \mathcal{C}_{B_{\leq}(O,r)}(p))$  then
2    $\mathcal{T} \leftarrow \mathcal{T} \cup \{\vec{0}\}$ 
3 end
    
```

The run times of both reference and proposed algorithms are given in Table 1 for various sizes and a few masks. These figures show that the overall mean computational complexity is linear with the maximal radius for the proposed method, whereas it is of order  $r^4$  in 3D for the reference algorithm.

**7. Conclusion**

In this paper, methods to compute both the chamfer LUT and the test neighborhood were presented. Speed gains from the reference algorithm [7] are attributable to the representation of chamfer balls as  $\mathcal{H}$ -polytopes. With this description, we avoid the use of weight

**Table 1**Run times (in seconds) of reference (ref.) and proposed ( $\mathcal{H}$ ) method implementations for four 3D masks and various volume sizes  $L \times L \times L$ .

$L$	3, 4, 5		3, 4, 5, 7		7, 8, 11, 14		4, 6, 7, 9, 10	
	Ref.	$\mathcal{H}$	Ref.	$\mathcal{H}$	Ref.	$\mathcal{H}$	Ref.	$\mathcal{H}$
10	0.0006	0.0003	0.0002	0.0004	0.0013	0.0010	0.0004	0.0008
20	0.0049	0.0002	0.0032	0.0004	0.0135	0.0011	0.0047	0.0010
50	0.1079	0.0003	0.1096	0.0005	0.2886	0.0027	0.1592	0.0015
100	1.6316	0.0005	1.6831	0.0008	4.4704	0.0054	2.4626	0.0023
200	30.391	0.0010	31.126	0.0012	81.627	0.0109	53.126	0.0038
500	3523	0.0022	3537	0.0024	9333	0.0272	6465	0.0088
1000		0.0042		0.0073		0.0549		0.0220

propagation in the image domain and obtain a constant time covering test by the direct computation of covering radii. Moreover the search space is greatly reduced using covering cones.

While applications always using the same mask can use precomputed test neighborhood  $\mathcal{T}$  and LUT, other applications that potentially use several masks, adaptive masks, variable input image size can benefit from these algorithms. A faster computation of  $\mathcal{T}$  is also highly interesting to explore chamfer mask properties. Beyond improved run times, the  $\mathcal{H}$ -polytope representation helped to prove new properties of chamfer masks. And a new formula of distance which does not need to find in which cone lies a point was given.

Examples and source codes for the 3D and 2D cases are available by simple email request to the authors or in the code section of the IAPR-TC18 web site (<http://www.cb.uu.se/~tc18/>).

#### Acknowledgment

The authors would like to thank the referees for their valuable suggestions and comments, which greatly improved the paper.

#### References

- [1] A. Rosenfeld, J.L. Pfaltz, Sequential operations in digital picture processing, *Journal of the ACM* 13 (4) (1966) 471–494.
- [2] U. Montanari, A method for obtaining skeletons using a quasi-Euclidean distance, *Journal of the ACM* 15 (4) (1968) 600–624.
- [3] G. Borgefors, Distance transformations in arbitrary dimensions, *Computer Vision, Graphics, and Image Processing* 27 (3) (1984) 321–345.
- [4] D. Coeurjolly, A. Montanvert, Optimal separable algorithms to compute the reverse Euclidean distance transformation and discrete medial axis in arbitrary dimension, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29 (3) (2007) 437–448.
- [5] R. Strand, Weighted distances based on neighborhood sequences, *Pattern Recognition Letters* 28 (15) (2007) 2029–2036.
- [6] G. Borgefors, Centres of maximal discs in the 5–7–11 distance transforms, in: *Proceedings of the 8th Scandinavian Conference on Image Analysis*, Tromsø, Norway, 1993, pp. 105–111.
- [7] É. Rémy, É. Thiel, Medial axis for chamfer distances: computing look-up tables and neighbourhoods in 2D or 3D, *Pattern Recognition Letters* 23 (6) (2002) 649–661.
- [8] N. Normand, P. Evenou, Medial axis LUT computation for chamfer norms using  $\mathcal{H}$ -polytopes, in: *Discrete Geometry for Computer Imagery, Lecture Notes in Computer Sciences*, vol. 4992, Springer, Berlin, Heidelberg, 2008, pp. 189–200. doi: 10.1007/978-3-540-79126-3\_18.
- [9] A. Rosenfeld, J. Pfaltz, Distances functions on digital pictures, *Pattern Recognition Letters* 1 (1) (1968) 33–61.
- [10] É. Thiel, Géométrie des distances de chanfrein, mémoire d'habilitation à diriger des recherches, Université de la Méditerranée, Aix-Marseille 2, December 2001 (<http://www.lif-sud.univ-mrs.fr/~thiel/hdr/>).
- [11] B.J.H. Verwer, Local distances for distance transformations in two and three dimensions, *Pattern Recognition Letters* 12 (11) (1991) 671–682.
- [12] É. Rémy, Normes de chanfrein et axe médian dans le volume discret, Thèse de doctorat, Université de la Méditerranée, December 2001.
- [13] C. Arcelli, G. Sanniti di Baja, Finding local maxima in a pseudo-Euclidian distance transform, *Computer Vision, Graphics, and Image Processing* 43 (3) (1988) 361–367.
- [14] S. Svensson, G. Borgefors, Digital distance transforms in 3D images using information from neighbourhoods up to  $5 \times 5 \times 5$ , *Computer Vision and Image Understanding* 88 (1) (2002) 24–53.
- [15] É. Thiel, Les distances de chanfrein en analyse d'images: fondements et applications, Thèse de doctorat, Université Joseph Fourier, Grenoble 1, September 1994 (<http://www.lif-sud.univ-mrs.fr/~thiel/these/>).
- [16] É. Rémy, É. Thiel, Computing 3D medial axis for chamfer distances, in: *Discrete Geometry for Computer Imagery*, Uppsala, Sweden, Lecture Notes in Computer Science, vol. 1953, Springer, Berlin, 2000, pp. 418–430. doi: 10.1007/3-540-44438-6\_34.
- [17] G.M. Ziegler, *Lectures on Polytopes* (Graduate Texts in Mathematics), Springer, Berlin, 2001.
- [18] G. Borgefors, Weighted digital distance transforms in four dimensions, *Discrete Applied Mathematics* 125 (1) (2003) 161–176.
- [19] C. Fouard, G. Malandain, 3-D chamfer distances and norms in anisotropic grids, *Image and Vision Computing* 23 (2) (2004) 143–158.
- [20] D.R. Chand, S.S. Kapur, An algorithm for convex polytopes, *Journal of the Association for Computing Machinery* 17 (1) (1970) 78–86.

**About the Author**—NICOLAS NORMAND was born in 1969. He received the M.S. degree in 1992 and his Ph.D. from the University of Nantes, France, in 1997. He has been an assistant professor in the Computer Science Department of École Polytechnique de l'Université de Nantes from 1997. His areas of interest include digital geometry and particularly discrete distances, data networks and the related Mojette transform.

**About the Author**—PIERRE EVENOU was born in 1973. He received the M.S. degree in 2003. He has been a research engineer in the Computer Science Department of École Polytechnique de l'Université de Nantes from 2005. His areas of interest include digital geometry and particularly discrete distances, data networks and the related Mojette transform.







# Multimedia Forward Error Correcting codes for wireless LAN

Benoît PARREIN\* — Nicolas NORMAND\* — Jeanpierre GUÉDON\*

## Abstract

*In this paper the forward error correction (FEC) codes useful for multimedia wireless transmissions are discussed. The class of usable codes are produced by multiple description (packet equivalence during transmission) after an initial hierarchical representation (joint source channel coding) stage. The Mojette transform is used to perform the implementation. A fair comparison with other optimal codes (systematic or not) is provided. JPEG2000 images transmission under a wireless link shows possible strategies.*

**Key words:**

---

## CODES CORRECTEUR D'ERREURS PAR ANTICIPATION POUR RÉSEAUX LOCAUX MULTIMÉDIAS SANS FIL

---

## Resumé

*Dans cet article sont discutés les codes correcteurs par anticipation (FEC) utilisables pour le transport du multimédia sur liens hertziens . La classe de codes utilisée est produite par une description multiple (équivalence des paquets sur le lien) après hiérarchisation des données initiales (codage conjoint source-canal). Pour ce faire, la transformation Mojette est utilisée. Une comparaison avec les autres types de codage systématiques ou non est présentée et l'application au transport d'images JPEG2000 montre les stratégies possibles.*

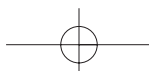
**Mots clés :**

## Contents

- |   |  |
|---|--|
| I. Introduction                               | IV. Application to images transmission |
| II. Multimedia Forward Error Correcting codes | V. Discussion                          |
| III. Mojette transform                        | VI. Conclusions                        |
|   | References (17 ref.)                   |

---

\* Laboratoire IRCCyN, Équipe Image & Vidéo Communications, CNRS UMR 6597, École, polytechnique de l'Université de Nantes , La Chantrerie Rue Christian Pauc, BP 50609, F-44306 Nantes cedex3, {prenom.nom}@irccyn.ec-nantes.fr and {prenom.nom}@polytech.univ-nantes.fr



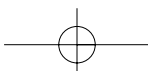


## I. INTRODUCTION

For the last ten years, the communications concept of Quality of Service (denoted QoS) has been understood at two distinguished levels: network and service by the corresponding communities. The first concerned applications were Internet both for the network (seamless while many different protocols coexist) and for the new services (convergence voice and IP traffic). Wireless communications are now involved in the QoS challenge and face more dramatic constraints for the physical layer. The most important paradigm to be solved for the communication QoS is the mixture between real-time and exact received information. While this trade-off was simply done for voice transmission only (errors at destination were ignored), complex error correcting codes have to be implemented for internet messages that consists of many layers of independent encoding (from the source encoding: transform, quantization, entropic coding; to the multiple network layers). Thus, forward error correcting codes (denoted FEC codes in the following) have been implemented at the application layer to protect against Internet packet losses. A more general mechanism called multiple description (denoted as MD) has been developed for Internet lately while studied in the early eighties for voice transmission (see [5]). For an information theory standpoint, it consists in generating multiple and redundant descriptions of the source and sending these independent descriptions along the network onto different channels. A reception of a limited number of descriptions allows for the recovery of the initial source. In this paper, this mechanism is adapted onto wireless links while considering each packet of information as emitted onto a specific channel. These considerations are detailed and exemplified in the following section. The implementation proposed here uses the Mojette transform to perform the multiple description. This transform is very simple as shown in section III, exhibit almost optimal properties from communication theory and a low order of complexity for its true implementation, and can be extended to multiple flows with different priorities in terms of signal recovery. This seems the clue for multimedia applications over wireless networks: the initial flows are prioritized (for instance the low frequencies of a still image or voice is more important than its high frequencies) but the wireless bitstream has to be simply implemented to be efficient. The term multimedia invoked here applies to voice, image, composite documents. The area of video transmission over wireless links is more complicated because of the time order relationships between the video encoding and the produced bitstream. However, some new attempts like the Moving JPEG2000 are concerned with the focus of this paper. The Mojette multiple description applied to JPEG2000 still images in section IV shows this compromise between blind discarded portions of bitstream and image recovery. In this way, it acts as a joint source-channel coding scheme. Related implementation of other optimal FEC codes are discussed in section 5.

## II. MULTIMEDIA FORWARD ERROR CORRECTING CODES

The rapid growth of multimedia real-time applications has led to different strategies for MFEC encoding when packets are used to convey the bit stream. From physical layer to application, this section reviews resilient mechanisms implemented at different stages, each corresponding to a specific source and channel definition.







## II.1. Physical layer

A first approach, that could be called the historical one, is to protect the stream (the source is here a series of Internet packets) at the physical (PHY) layer. Classical channel coding is employed in this case. While no Forward Error Correcting (FEC) code is implemented in the version b of 802.11, both the version a and its european counterpart HIPERLAN 2 (H/2) use convolutional coding at the PHY layer. For a fixed message length in bits,  $k = 7$ , the common FEC rate is  $1/2$ . By puncturing codes, other available rates are  $2/3$ ,  $3/4$  for the 802.11a and  $9/16$ ,  $3/4$  for H/2 [17]. Block codes like Reed-Solomon and also scrambling could be added on the Media Access Control (MAC) layer (802.11e).

A more sophisticated method should support FEC by adapting adequately modulation methods. BPSK, QPSK, 16QAM and 64 QAM are chosen according to the radio link quality or the application prioritisation. For example, the 802.11a standard consists of 8 PHY layer modes (from 6 Mbps to 54 Mbps). In [16], this modes are used to adequately protect both the base-ment layer and the enhanced layers. More exceptional is when the adaptive modulation takes into account both source and channel properties.

## II.2. At the application level (JPEG2000)

At the application level, the multimedia stream syntax improves resilience in a noisy environment. Audio encoders take profit of the packetization to send in a single packet both a classical piece of voice and a low frequency version of the previous packet [14]. More elaborated, the recent still image standard JPEG-2000 specifies several resilient tools [10] to preserve the source from multiple error types as corrupted packet body, packet header errors, missing bytes.

Basically, JPEG-2000 resilient tools work at two levels. First, an entropy coding box is implemented to protect code-block data. A special symbol sequence called segmentation symbols can be coded with a fixed context at the end of each bitplane. If a wrong sequence is decoded, an error occurs and the following bit planes are considered as corrupted. This method can be improved by a regular predictable algorithm which introduces useful correlations.

The segmentation in packets provides a second stage of protection. The short size of packets reduces losses within a small spatial image area. Furthermore, every packet header can be stamped by a Start Of Packet (denoted as SOP) marker with a sequence index. In the erasure case, the next unaffected packet is searched in the codestream and decoding proceeds from here. For a better source integrity, these packet headers can be moved to the main (or tile) headers and transmitted through a channel with much lower error rate. Despite all these resilient tools, very few missing bytes can lead to impossible decoding. Even in case of possible decoding, resulting pictures can be below service acceptability threshold. A decoding example is given in Figure 1.

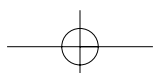




FIG. 1. – JPEG 2000 image reconstruction including resilient mechanisms after a burst error of thirty consecutive bytes starting at byte 0xD9C.

*Reconstruction d'une image JPEG-2000 incluant les outils de résistance aux pertes pour une rafale s'étalant sur trente octets consécutifs à partir de l'octet 0xD9C.*

### II.3. Priority encoding transmission from MDS codes

Notice that modern multimedia encoding will make use of different sub streams to transmit hierarchical information to the network. Image encoding using JPEG2000 or SPIHT encoders provide different quality streams ; each of them could therefore be encoded with respect to network priorities. Visio or video encoding can also gain from this kind of hierarchy for transmitting images differently according to the specific temporal encoding. When no hierarchy can be transmitted along the network protocol or when for different packets of the same hierarchy level can not all be transmitted in a given time range, a mechanism of priority encoding transmission (called PET) has to be set up. When this is the case, packet are all supposed to have the same probability to cross the network (and thus to be erased in a node or onto a physical link). This situation has been largely examined for different networks protocols during the last twenty years. For instance, the AAL1 layer superimposed on top of the ATM layer was designed for an ATM packet loss (conveying multimedia real-time contents) in



the beginning of the eighties. It was implementing a Reed Solomon code  $RS(128,124)$ . This means that for 124 packets of information four additional packets were generated and then the resulting 128 packets were sent to the network. When a set of any 124 packets arrives at destination the original information is retrieved. In this case, the code is optimal and belongs to the Minimum Distance Separable (MDS) family.

Chapter 11 of the Mac Williams and Sloane [11] describes MDS codes as the most fascinating part of the coding theory. Actually, it was shown that if  $n$  packets are computed from  $m$  message packets over a Galois field,  $GF[2^L]$ , any set of  $m$  packets among  $n$  is sufficient to decode the original message. The property is illustrated in Figure 2. Reed-Solomon represents the most important class belonging to the MDS family.

The construction of MDS codes proposed by [1] can be realized with a generator matrix if and only if each squared sub-matrices are invertible. Cauchy and Vandermonde matrices have both this property but the implementation with a Cauchy matrix [1] are much faster than Vandermonde implementation [15]. However, MDS codes imply a high computational cost.

Low cost, approximated MDS codes called  $(1+\epsilon)MDS$  are often preferred. These codes need  $(1+\epsilon)m$  packets at the decoding side. With this property, Tornado codes [9] provides a good trade-off between the optimal solution and encoding/decoding times. Redundant packets are computed by linear combinations of message packets with redundant packets. Matrices are very sparse contrary to Cauchy matrices. The average number of variables per equations is small enough to operate by simple substitutions. The reception of a packet number lightly greater than  $m$  leads to a tornado of substitutions. Below this number, few substitutions proceed. Tornado codes only use the XOR operation and avoid fields operations and matrix inversions. These codes are used in PET systems. By computing the correct stretch factor according to channel properties, it is possible to satisfy sub streams priorities. However, decoding is probabilistic in the sense that the message can be obtained from  $(1+\epsilon)m$  with a good probability. Moreover, the decoding inefficiency,  $\epsilon$ , is 5.4% for video delivery [2] and the granularity of protection *i.e.* allowed redundant rates can be ensured with a high complexity cost. All these reasons leads to propose an other MFEC.

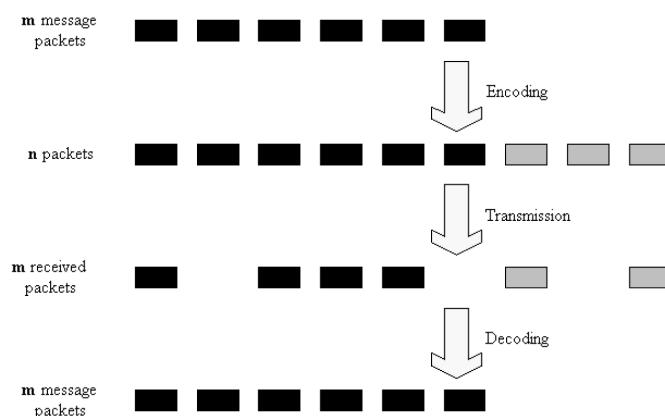


FIG. 2. – How to balance packet loss by FEC codes ? From a set of  $m$  message packets, a redundant set is computed into  $n$  packets ( $n \geq m$ ). The reception of at least  $m$  packets allows to decode the original message in the MDS case.

*Comment compenser la perte de paquets par les codes FEC ? A partir d'une série de  $m$  paquets de message, une série redondante est calculée au travers de  $n$  paquets ( $n \geq m$ ). La réception d'au moins  $m$  paquets permet de décoder le message original dans le cas MDS.*



### III. MOJETTE TRANSFORM

#### III.1. From a 2-D support to multiple 1-D supports

The Mojette transform is based on the Radon transform that allows to describe a 2-D signal by a projected 1-D signals set [12]. To realize the transform, additions are simply used in projection directions which is determined by a couple of integers  $(p, q)$  prime to each other. In the case of binary elements, the XOR operation is used. Each projection is composed with a set of sum elements called bins. The definition of a projection  $(p, q)$  is established from direct Mojette transform of an image  $f(k,l)$  and defined as follows :

$$(1) \quad M_{p,q}f(k,l) = \sum_k \sum_l f(k,l) \Delta(m + kq - lp),$$

with the Kronecker function,  $\Delta(m) = \{ 1 \text{ if } m = 0, 0 \text{ if } m \neq 0 \}$ .

Equation E.1 is the definition of the Mojette-Dirac transform : the summation in the plane corresponds to only take on each line crossed points of the 2-D grid as depicted Figure 2. Other interpolation kernels may be used but are not described in this paper (see [7]).

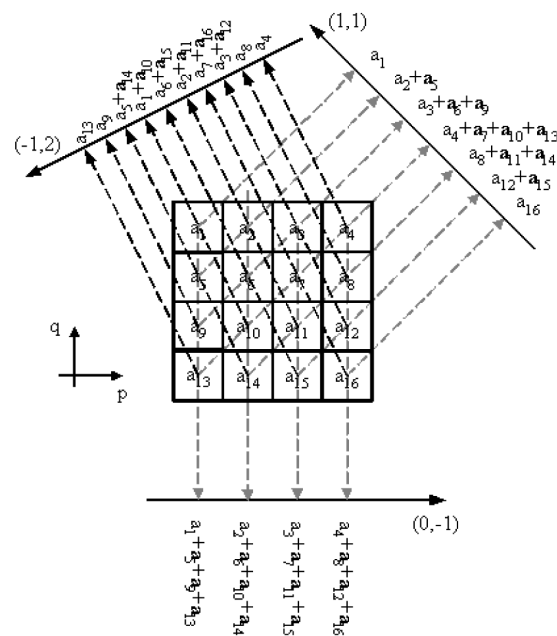


FIG. 3. –  $4 \times 4$  square Mojette transform with three projections.

*Transformation Mojette sur un carrée  $4 \times 4$  pour trois projections.*



By extension, the direct Mojette transform of an image  $f(k, l)$  of rectangular support  $P \times Q$  is defined by  $MF$  that represent a set of  $N$  projections  $M_{p,q}f(k, l)$  such as

$$(2) \quad Mf = \{\text{proj}_{p_i, q_i}, i = 1, 2, \dots, N\}.$$

The transform is illustrated Figure 3 for three classical projections. Since each pixel contributes to a bin, complexity is necessarily of  $O(PQ = I)$  for any projection. For a set of  $N$  projections, complexity is simply  $O(IN)$ , linear both in the pixels number  $I$  and in the projections number. The construction and the reconstruction order is not a priori fixed.

## III.2. Reconstruction conditions

### III.2.1. Rectangular support

Let  $A$ , a rectangular support composed of  $P \times Q$  pixels. Let  $B$  be the number of bins of the projection defined from its angle  $(p, q)$  :

$$(3) \quad B = (Q - 1) |p| + (P - 1) |q| + 1.$$

A lemma found in [8] and corresponding to this situation gives the following result. Let  $S$ , a set of  $N$  projections  $\{(p_i, q_i), i \in N\}$

$$(4) \quad S \text{ reconstructs } A \Leftrightarrow P \quad P_I = \sum_{i=1}^I |p_i| \quad \text{or} \quad Q \quad Q_I = \sum_{i=1}^I |q_i|.$$

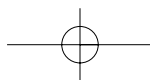
### III.2.2. Generalization by mathematical morphology

For a non rectangular support, equations E.3 and E.4 are not applicable. In order to compute the number of bins, sides of any polytope viewed from angle  $(p, q)$  can not be ever written so easily. This problem can be simplified by considering bins number as the difference between indices  $m_1$  and  $m_{\max}$  of first and last bin :

$$(5) \quad B = |m_{\max} - m_1| + 1$$

In order to generalize the reconstruction condition, a projection angle  $(p, q)$  is associated with 2 pixels structuring elements (2PSE) as a point couple  $\{O, (p, q)\}$ . It is demonstrated in [12] that a back projection corresponds in fact to a morphological opening (erosion and dilation) of the support with 2PSE defined by the direction.

It means that a image reconstruction is likely a set of morphological openings by 2PSE defined by projection direction. The reconstructibility criteria of Katz find here an another expression with just dilations. A convex image is not reconstructible if and only if the dilation result by 2PSE is not included in the image support (even if one pixel is concerned). This general result is expressed in the following theorem by the equivalence of two statements.



**Reconstructibility theorem [12]****Statement 1 : both propositions are equivalent**

- i)  $f(k,l)$  defined on the convex  $G$  is reconstructible by  $\{proj_{p_i,q_i}, i=1,2,\dots,N\}$  ;
- ii)  $R$  constructed by  $N$  dilations set  $\{O,(p_i,q_i),i=1,2,\dots,N\}$  is not included in  $G$ .

**Statement 2 : both propositions are equivalent**

- i)  $G$  is reconstructible by  $\{proj_{p_i,q_i},i=1,2,\dots,N\}$  ;
- ii) the erosion of  $G$  by  $R$  gives the empty set.

To compute the reconstruction, the reverse algorithm needs three kinds of data : the bin value, the sum of crossed pixels and the sum of pixels positions (the last two informations are computed at the decoding side without transmission and were added to speed-up the algorithm). The first step consists in finding explicit bins that can be directly back projected. Following the location of the pixels, all projections are then updated in corresponding bins. Next steps find other explicit bins up to the final reconstruction. Figure 4 exemplified the beginning of the reverse algorithm.

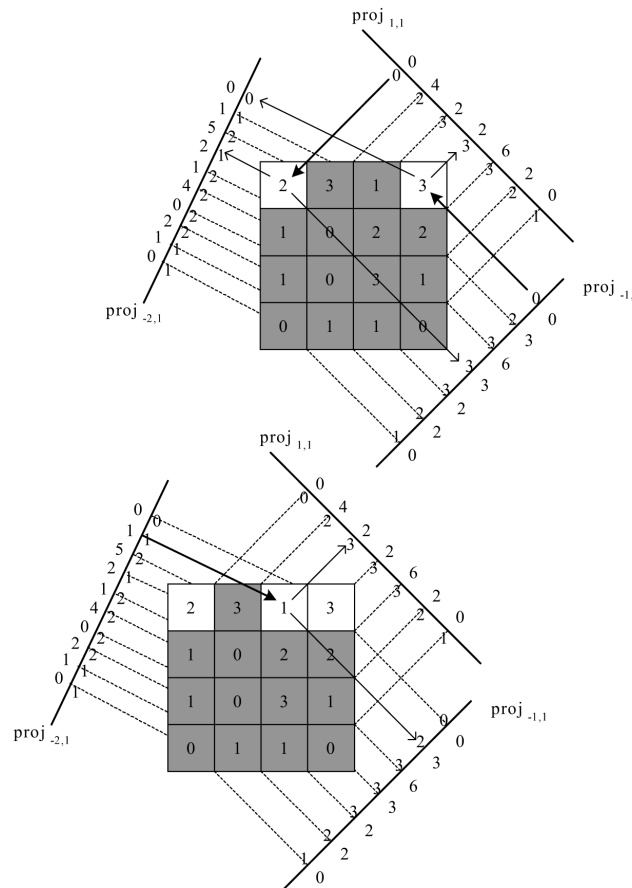


FIG. 4. – Beginning of the reverse Mojette transform. Three pixels are reconstructed. The top value on each line represents the summation value whereas the underline number gives for each bin the corresponding current number of pixels.

*Début de la transformation Mojette inverse. Trois pixels sont reconstruits. Sur les projections, les valeurs situées vers l'extérieur correspondent aux sommes des pixels alors que les valeurs situées à l'intérieur indiquent le nombre de pixels contribuant à chaque bin.*

#### IV. APPLICATION TO IMAGES TRANSMISSION



FIG. 5. – Lena test image encoded by JPEG-2000 with a rate of 1 bit per pixel (bpp). A  $64 \times 64$  Region of Interest (ROI) is specified and retained as the most important information of the source.

*Image test Léna codée par du JPEG-2000 pour un débit de 1 bpp. Une région d'intérêt est spécifiée et retenue comme étant l'information la plus importante de la source.*

This section illustrates the implementation of priority encoding transmission (PET) system using the Mojette transform onto scalable streams featuring JPEG-2000 [6]. In this application to images transmission, the source coding is spread into 32 equivalent descriptions which can be used in any order of reception. Without this MFEC layer, original JPEG 2000 provides non equivalent packets : erasures at the head of the stream lead to important degradations.

A  $512 \times 512$  Lena test image is encoded at 1 bit per pixel (1 bpp) including a  $64 \times 64$  Region Of Interest (ROI) coding as depicted in Figure 5. This ROI is considered as the most important information. The total initial information volume is 256 Kbits partitioned in four substreams. Its distribution is described in table I. Substreams are allocated in a support shown on Figure 6 that is called a geometrical buffer for its role in this transmission scheme. These kind of supports produces projections with a constant size. Moreover, the reconstruction is really scalable.

The 32 angles are taken such that all projections are equivalent for the reconstruction which only depends on the number of received projections. In [13], it is shown that this condition is checked for the set  $S = \{(p, q); p = -15, -14 \dots 15, 16 \text{ and } q = 1\}$ .

Protection levels (*i.e.* the numbers of required projections) were computed for an exponential loss profile with a mean value of 10 %. The computation consists in the maximization of a quality criteria as the Peak Signal to Noise Ratio (PSNR) [13]. Our system protects substream 1 by a possible reconstruction from 22 projections among 32 (erasure packets of .6875 % authorized) whereas substreams 2 and 3 can be reconstructed by 28 and 30 projections respectively among the 32 sent. The last substream requires all projections for decoding. Each projection contains 1047 bins which are byte encoded.

TABLE I.– Description of substreams including sizes, quality at the reconstruction and number of required projections.

*Description des sous-flux incluant la taille, la qualité obtenue après reconstruction et le nombre requis de projections.*

	<b>Stream 1</b>	<b>Stream 2</b>	<b>Stream 3</b>	<b>Stream 4</b>
Size (Kbits)	33	43	86	94
PSNR (dB)	22.12	32.64	37.21	39.56
#projections	22	28	30	32

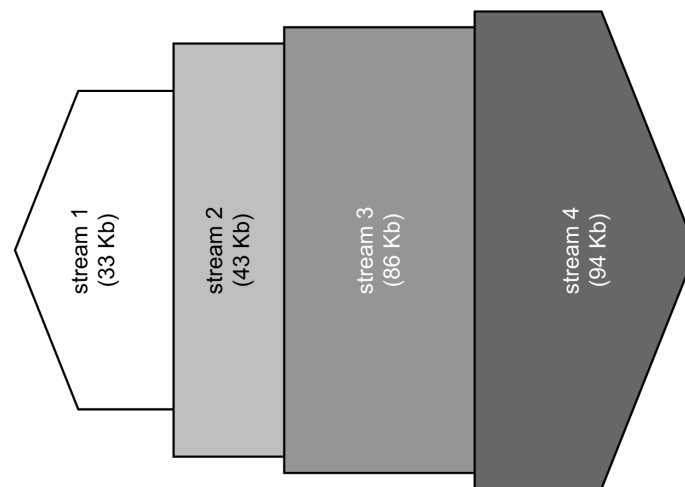


FIG. 6. – Geometrical buffer used for the substreams multiplexing.

*Mémoire tampon géométrique utilisée pour le multiplexage des sous-flux.*

The image is progressively reconstructed with the successive incoming projections. Figure 7 shows the quality criteria *i.e.* PSNR for the image and for the ROI related to the received projections number. Corresponding images are given on Figure 8. An acceptable level (35.08 dB) is quickly obtained for the ROI with just 22 projections. The whole image contains many degradations at this reconstruction step. Six following projections bring more details in





order to obtain a suitable version (32.64 dB). Refinements are given with the decoding of substreams 3 and 4 respectively with 30 and 32 projections. Both ROI and image qualities are improved with a significant gain for the whole image.

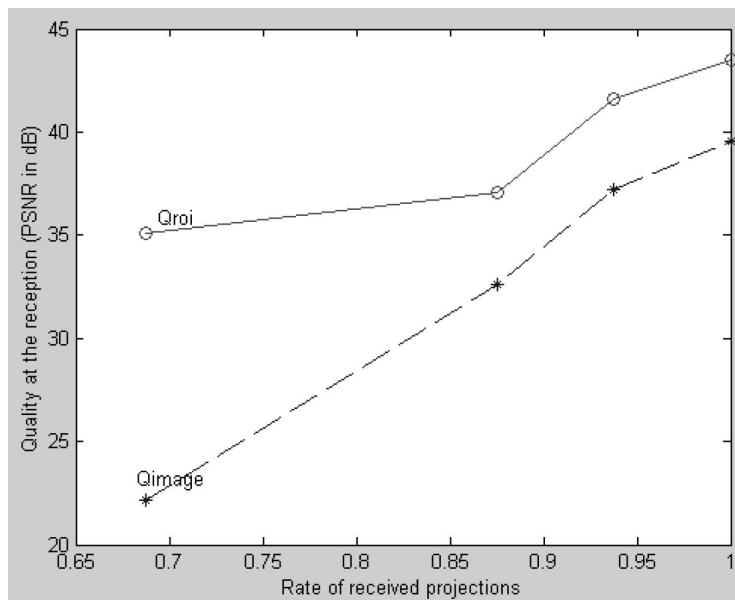


FIG. 7. – Quality levels obtained after the reception of 22, 28, 30 and 32 projections. A good quality for the ROI (in filled line) is quickly obtained as soon as the first 22 projections are available.

*Niveaux de qualité obtenus après réception de 22, 28, 30 et 32 projections. Pour la ROI, un bon niveau de qualité (ligne pleine) est obtenu rapidement dès que les 22 premières projections sont disponibles.*

## V. DISCUSSION

### V.1. Systematic or not systematic codes ?

#### V.1.2. Systematic codes

Implementation of a systematic FEC code for multimedia transmission turns out to be very efficient in terms of real-time characteristics while cumbersome and complicated to manage. The codes belonging to this category are the MDS Reed-Solomon and the  $(1+\epsilon)$ MDS Tornado codes. When the source produces the multimedia streaming a first version can be packetized while its copy can be used to compute the redundancy. This point of view allows for a true real-time gain (no lag generated at the encoder) only if the algorithmic complexity is low enough compared to the time range for sending the original information. Of course, this temporal range depends on the possible bitrate on the channel and will be considered as small on

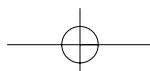




FIG. 8.– Successive refinements of Lena image. From upper left to lower right, Lena is reconstructed with 22, 28, 30 and 32 projections.

*Raffinements successifs de l'image Léna. De l'image en haut à gauche à celle d'en bas à droite, Léna est respectivement reconstruite par 22, 28, 30 et 32 projections.*

wireless channels. Moreover, the encoder complexity is high and a lot of spatial memory is needed.

At the decoder side, the lag is only present when erasures occur. Thus when erasures probability is high as in the wireless case, the decoding delay can not justify systematic codes implementation. However, it has been very often referred to the use of the systematic codes for multimedia transport as the right way to do because all the resilient errors algorithms are based on this principle (the data is not entirely arrived but the available portion can be used to predict the erased content). Multiple Description paradigm follows the reverse way : a single description must contain as much as possible of the global information which can only be represented by transforming the original information.



### V.2.3. Non systematic codes

When a non systematic code is used, each packetized portion of information conveys an information that can be added to the (already) decoded information to enhance the result as in the multiple description scheme. In this respect, RS codes give binary results : either there is not enough at the decoder and nothing is restituted or the whole initial information can be computed. By comparison, the Mojette transform gives progressive decoding according to the number of descriptions present at the decoder site. Another advantage of non systematic codes is the ability to use the inherent redundancy to procure an additional error detector/corrector scheme. However, this supplementary stage can only be implemented when the lower network layer is able to feed the higher layer with noisy data. In such a case, a modified version of the Mojette transform denoted as spline Mojette transform will add a convolution step for each projection with a specific discrete kernel according to the angle of projection. The goal is to correlate the data by spreading it onto the projection when using a known kernel [7].

TABLE II.– Systematic and non-systematic code comparison.

*Comparaison des codes systématiques et non-systématiques.*

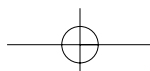
	+	–
Systematic Codes	1. Null complexity for zero packet loss case 2. Easy data pipeline	1. Reduced protection granularity 2. Complexity increasing for high packet losses
Non-systematic Codes	1. Constant complexity whatever the reception scenario 2. Fitted granularity to source and channel constraints	1. Almost MDS 2. Constant complexity

## V.2. Robustness of JPEG-2000

Let try to highlight the resilience lack of a JPEG-2000 stream. The weakness is principally due to the embedded structure of the source. All bytes are not equivalent. An erasure is much more destructive when located at the beginning of the flow.

In the following, an erasure is simulated and included in the JPEG-2000 stream for several coding methods. The length of the error is constant and concerns thirty consecutive bytes. Considering  $512 \times 512$  Lena coded at 1 bpp, the bit error rate (BER) is  $9 \cdot 10^{-4}$ . This rate correspond to a basic erasure over a short distance wireless link [3]. All missing bytes are set to zero because an entire missing packet does not allow the decoding. The main header is supposed error free as recommended in [4].

First, the describing error is moved regularly from the first JPEG-2000 packet (excluding main header) until it has no more effects. All erasures are tested with error resilient (ER) tools and over a simple source. As depicted in Figure 9, the ER source which includes segmentation symbols can be reconstructed more easily than the basic stream. Without resilient



mechanism, an erasure at the beginning (near first third) does not allow the decoding. The segmentation symbols improve significantly the qualities reconstruction. Very good versions are obtained when the erasure location is located between 3500<sup>th</sup> and 10000<sup>th</sup> byte. However, this reconstruction is random when the burst occurs before. This weak zone concerns at least 10 % of the total volume of information elements.

When the decoding can be done, the reconstructed images are composed of several visual defects which can be unacceptable for certain services. Despite a better resilience than the previous source (better synchronisation of coding words, no bypass of the AC), the image shown in Figure 1 illustrates a burst occurring at the byte 0xD9C i.e. 3484. This erasure implies the loss of just one JPEG-2000 packet.

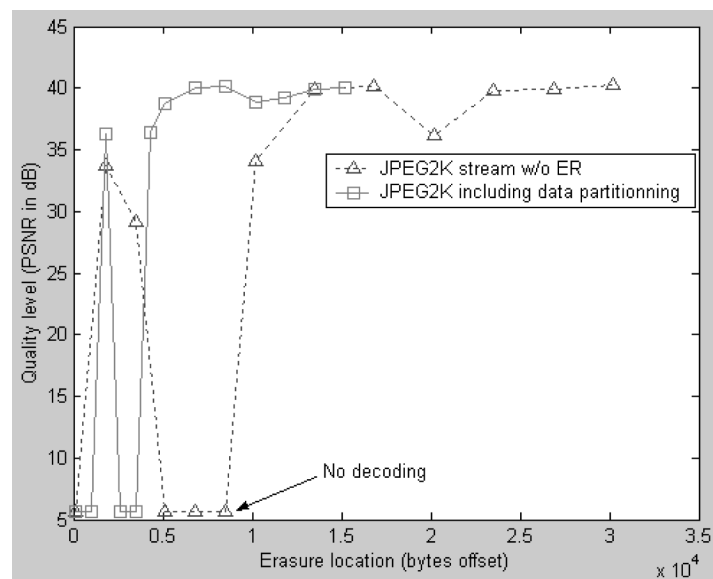


FIG. 9.– Reconstruction qualities (PSNR) in function of the erasure location for an error resilient (filled line) and a basic JPEG 2000 source.

*Qualités de reconstruction (PSNR) en fonction de la position de l'effacement pour une source JPEG-2000 résistante aux erreurs (ligne pleine) et une source JPEG-2000 classique (ligne pointillée).*

## VI. CONCLUSIONS

Network transmissions will always have to deal with packet losses, especially in the case of wireless applications. Fortunately, when integrity is mandatory and delay is not an issue, a full data reception can be ensured at the transport protocol level by retransmitting lost packets. However, multimedia streams do not fit in this scheme. If lost packets of an interactive multimedia stream were retransmitted, they would arrive too late for playback. A better strategy is to protect the emitted data such that the receiver can use the redundancy in order to retrieve partly or totally the lost information.



This protection can take place at several stages of the transmission, from the physical layer to the application. When resilient tools are used at the application level, we can still observe dramatic quality drops when some specific parts of the data is lost. This relative sensibility has been shown for JPEG 2000 in this paper. Thus, some extra protection is needed but this will have a cost in terms of network bandwidth. The seamless network issue leads also to put this protection at high level in order to be used both for wireless and for IP networks.

To achieve an optimal use of the bandwidth, both network and source characteristics must be taken into account. Priority encoding systems allocate redundancy according to the weight of each information part. Using progressive encoding, they reach the multiple description motto: each new partial description introduces a refinement in the decoding.

Classical PET implementations use either Reed-Solomon codes, which are optimal in terms of coding efficiency because of their MDS properties but have high computational costs, or Tornado codes which are easier to compute but require an extra unusable redundancy  $(1+\epsilon)$ MDS and have only probabilistic properties.

As an alternative to both RS and Tornado codes the Mojette transform was proposed. It is a  $(1+\epsilon)$ MDS code with deterministic properties and very low computational needs. Furthermore, the decoding cost is constant (only depending on the amount of decoded data). Thus, the Mojette transform constitutes an excellent code for PET systems, especially when many packets are lost as it is the case for wireless communications.

*Manuscrit reçu le 8 octobre 2002*

*Accepté le 17 février 2003*

## Acknowledgements

The authors would like to thank Dr David Guyard from Bouygues Telecom who gave a useful contribution for this work. This study was supported by the state-region Pays de la Loire grant named “multimedia mobile communications” (C2M).

---

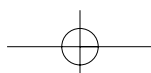
## REFERENCES

- [1] BLÖMER (J.), KALFANE (M.), KARP (R.), KARPINSKI (M.), LUBY (M.), ZUCKERMAN (D.), An XOR-Based Erasure-Resilient Coding Scheme, ICSI Tech. Rep. TR-95-048, August 1995.
- [2] BYERS (J.W.), LUBY (M.), MITZENMACHER (M.), REGE (A.), A Digital Fountain Approach to Reliable Distribution of Bulk Data, ICSI Tech.Report, TR-98-013, 1998.
- [3] COSTAMAGNA (E.), FAVALLI (L.), GAMBA (P.), “Multipath Channel Modeling With Chaotic Attractors”, Proc. of the IEEE, **90**, n° 5, 2002, p. 842-859.
- [4] EDWARDS (E.), FUTEMA (S.), ITAKURA (E.), TOMITA (N.), LEUNG (A.), FUKUHARA (T.), RTP Payload Format for JPEG 2000 video streams, Internet Draft, IETF, june 2002.
- [5] GOYAL (V.), “Multiple description coding : compression meets the network”, IEEE Signal Processing Mag., **18**, no. 5, pp. 74-93, Sept. 2001.
- [6] GROSBOIS (R.), SANTA-CRUZ (D.), EBRAHIMI (T.), “New Approach to JPEG-2000 compliant Region of Interest Coding”, SPIE 46<sup>th</sup> Applications of Digital Image Processing, San Diego, 10p., 2001.





- [7] GUÉDON (J.), NORMAND (N.), “Spline Mojette Transform : Applications in Tomography and Communications”, *Eusipco* 2002, **2**, pp. 407-410, 2002.
- [8] KATZ (M.), Questions of uniqueness and resolution in reconstruction from projections, Lectures notes in Biomathematics, **26**, *Springer Verlag*, 1979.
- [9] LUBY (M.), MITZENMACHER (M.), SHOKROLLAHI (A.), SPIELMAN (D.), STEMANN (V.), “Practical Loss-Resilient Codes”, 29<sup>th</sup> ACM Symposium on Theory of Computing, 150-159, 1997.
- [10] MOCCAGATTA (I.), SOUDAGAR (S.), LIANG ( J.), CHEN (H.), “ Error resilient coding in JPEG 2000 and MPEG 4”, *IEEE JSAC*, **18**, n° 6, june 2000, pp. 899-914.
- [11] MAC WILLIAMS (F.), SLOANE (N.), *The Theory of Error Correcting Codes*, *North-Holland*, 1977.
- [12] NORMAND (N.), GUÉDON ( J.), “La transformée Mojette : une représentation redondante pour l’image”, *Comptes rendus de l’Académie des Sciences de Paris, Section informatique théorique*, pp. 123-126, jan 1998.
- [13] PARREIN (B.), “Description Multiple de l’Information par transformation Mojette”, PhD thesis (in french), Univ. Nantes (2001), 121-125. available at <http://www.irccyn.ec-nantes.fr/~parrein>.
- [14] PERKINS (C.), KOUVELAS (I.), HODSON (O.), HARDMAN (V.), HANDLEY (M.), BOLOT (J.C.), VEGA-GARCIA (A.), FOSSE-PARISIS (S.), RTP Payload for Redundant Audio Data, RFC 2198, IETF, Sept. 1997.
- [15] RIZZO (L.), VICISANO (L.), “A Reliable Multicast Data Distribution Protocol based on Software FEC technics”, *Proc. of the fourth IEEE HPCS’97 workshop*, 10 p.
- [16] VAN DER SHAAR (M.), MEEHAN (J.), “ Robust Fine-Granularity-Scalability for Wireless Video”, *pv 2002 workshop*, 10 p., April 2002.
- [17] WALKE (B.), ESSELING (N.), HABETHA (J.), HETTICH (A.), KADELKA (A.), MANGOLD (S.), PEETZ (J.), VORNEFELD (U.), “IP over Wireless Mobile ATM - Guaranteed Wireless QoS by HIPERLAN/2”, *Proc. of the IEEE*, **89**, n° 1, p. 21-40, 2001.



# Index

## A

additivité, 98, 104  
axe médian, 77, 78, 80, 90, 92, 97  
  translaté, 97

## B

boule rationnelle, 77, 78, 81, 82, 84, 86, 87, 89

## C

caractéristique d'EULER-POINCARÉ, 98, 102–104, 106, 107, 111  
cellule, 101  
chemin minimal, 33  
cône, 81  
  de couverture, 85  
  générateur, 35  
  simplicial, 81, 82  
  translaté, 81  
  unimodulaire, 82

## D

dilatation, 18, 54  
discrétisation de GAUSS, 69  
disque, 32  
  fermé, 32  
  maximal, 78, 97  
  ouvert, 32

## distance

  à séquence de voisinages, 35, 65, 97, 109  
  à séquence de voisinages asymétrique, 61  
  de chanfrein, 36, 52, 97  
  discrète, 32  
  euclidienne, 97  
  locale, 35  
  octogonale, 31, 34, 61, 96, 97, 110  
  octogonale translatée, 97  
  simple, 97

## E

élément structurant  
  à deux pixels, 18

ensemble convexe, 56

érosion, 18, 51, 54, 55, 67, 75

## F

fermeture, 55  
  algébrique, 55

## fonction

  d'appui, 56, 57, 68, 75  
  distance, 32, 79, 80  
  d'ouverture, 94, 96, 97

formule de STEINER, 98, 99, 102

## G

granulométrie, 93, 94, 102, 111

## H

$\mathcal{H}$ -polytope, 77, 78, 80, 89, 91  
hyperplan d'appui, 56, 83, 86

## I

inverse de LAMBEK-MOSER, 37, 47, 63

## M

masque de chanfrein, 35, 36, 79, 80

## N

norme, 32

## O

ouverture, 54, 55  
  algébrique, 54

## P

parallélépipède  
  fondamental, 82  
polyèdre convexe, 56  
polytope, 56, 57  
  discret, 57  
pondération, 35, 81, 86  
portée, 99, 100  
  directionnelle, 111  
principe d'inclusion-exclusion, 98, 102, 104

## S

séquence

complémentaire, 36, 37  
 de BEATTY, 36, 61  
 de BEATTY rationnelle, 61  
 non décroissante, 37, 38  
**somme de MINKOWSKI, 19, 48–50, 54–56, 58, 60, 68, 74, 75, 77, 98, 102, 106, 109**

**spectre**  
 granulométrique, 94–97, 102  
**spectre granulométrique, 93**

**subdivision, 101**  
 mixte, 101, 102

**T**  
**théorème de PICK, 104**

**transformation**  
 de Serra, 54  
 en couverture, 84, 85, 89

en distance, 51, 55, 59, 63, 68, 75, 77, 78, 85, 89, 92, 93, 95–97, 102, 109–111  
 généralisée, 56, 67, 75  
 en distance généralisée, 59, 60  
 en distance translatée, 59, 65, 110  
 en ouverture, 89, 93, 102, 104, 108, 110, 111

**tube, 98**

**V**  
**valuation, 98, 102–104, 107**

**voisinage, 33**  
 1-, 33  
 2-, 33  
 de test, 77–80  
 $k$ -, 33  
 $k$ - strict, 33

**volume**  
 intrinsèque, 99  
 mixte, 101



# Bibliographie

- [1] Yves Amouriq, Pierre Evenou, Aurore Arlicot, **Nicolas Normand**, Pierre Layrolle, Pierre Weiss, Jeanpierre Guédon, « Evaluation of trabecular bone patterns on dental radiographic images : influence of cortical bone », dans *SPIE Medical Imaging 2010 : Biomedical Applications in Molecular, Structural, and Functional Imaging*, t. 7626, San Diego, CA, USA : SPIE, fév. 2010. DOI : 10.1117/12.843964.
- [2] Yves Amouriq, Jeanpierre Guédon, **Nicolas Normand**, Aurore Arlicot, Pierre Weiss, Yassine Benhdech, « Bone texture analysis on dental radiographic images : results with several angulated radiographs on the same region of interest », dans *SPIE Medical Imaging 2011 : Biomedical Applications in Molecular, Structural, and Functional Imaging*, t. 7965, Lake Buena Vista, Florida, USA : SPIE, fév. 2011, p. 796 525. DOI : 10.1117/12.878152.
- [3] Giovanni Anelli, Alberto Broggi, Giulio Destri, « Decomposition of arbitrarily shaped binary morphological structuring elements using genetic algorithms », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, t. 20, n° 2, p. 217–224, fév. 1998. DOI : 10.1109/34.659943.
- [4] Carlo Arcelli, Gabriella Sanniti di Baja, « Finding local maxima in a pseudo-Euclidian distance transform », *Computer Vision, Graphics, and Image Processing*, t. 43, n° 3, p. 361–367, sept. 1988. DOI : 10.1016/0734-189X(88)90089-8.
- [5] Aurore Arlicot, Yves Amouriq, Pierre Evenou, **Nicolas Normand**, Jeanpierre Guédon, « A single scan skeletonization algorithm : application to medical imaging of trabecular bone », dans *SPIE Medical Imaging 2010 : Image Processing*, t. 7623, San Diego, CA, USA : SPIE, fév. 2010. DOI : 10.1117/12.844250.
- [6] Aurore Arlicot, Pierre Evenou, **Nicolas Normand**, « Single-scan skeletonization driven by a neighborhood-sequence distance », dans *Advances in Image Analysis and Applications*, R. P. BARNEVA, V. E. BRIMKOV, K. KOROUTCHEV et E. KORUTCHEVA, édés., Madrid, mai 2011, p. 61–72, ISBN : 978-981-08-7923-5, ERA 43761 C.
- [7] Aurore Arlicot, **Nicolas Normand**, *Extraction of bone structure with a single-scan skeletonization driven by distance*, Nantes, mai 2011.
- [8] Marie Babel, Benoît Parrein, Olivier Déforges, **Nicolas Normand**, Jeanpierre Guédon, Véronique Coat, « Joint source-channel coding : secured and progressive transmission of compressed medical images on the Internet », *Computerized Medical Imaging and Graphics*, t. 32, n° 4, p. 258–269, juin 2008. DOI : 10.1016/j.compmedimag.2008.01.002, 5-year IF : 1.363 (ISI 2008), ERA 15962 B.

- [9] Marie Babel, Benoît Parrein, Olivier Déforges, **Nicolas Normand**, Jeanpierre Guédon, Joseph Ronsin, « Secured and progressive transmission of compressed images on the internet : application to telemedicine », dans *Color Imaging X : Processing, Hardcopy, and Applications*, G. G. ESCHBACH Reiner ; Marcu, éd., SPIE, t. 5670, San Jose, CA, jan. 2005, p. 126–136. DOI : 10.1117/12.589191.
- [10] H. G. Barrow, J. M. Tenenbaum, R. C. Bolles, H. C. Wolf, « Parametric correspondence and chamfer matching : two new techniques for image matching », dans *Proceedings of the 5<sup>th</sup> international joint conference on Artificial intelligence*, Morgan Kaufmann Publishers Inc., t. 2, Cambridge, Massachusetts, USA, 22–25 août 1977, p. 659–663.
- [11] Alexander I. Barvinok, « A polynomial time algorithm for counting integral points in polyhedra when the dimension is fixed », *Mathematics of Operations Research*, t. 19, n° 4, p. 769–779, nov. 1994. DOI : 10.1287/moor.19.4.769.
- [12] Samuel Beatty, « Problem 3173 », *The American Mathematical Monthly*, t. 33, n° 3, p. 159, mar. 1926. DOI : 10.2307/2300153.
- [13] Stéphane Beaumont, Tarraf Torfeh, Yves Barbotteau, Norbert Ailleres, Jeanpierre Guédon, **Nicolas Normand**, Sylvain Crespin, « Automatic image quality control of on-board imaging systems used for patient positioning in radiation therapy », dans *Dreiländertagung, Congress of the Swiss, Austrian and German societies of medical physics*, Bern, sept. 2007.
- [14] Stéphane Beaumont, Tarraf Torfeh, Jeanpierre Guédon, **Nicolas Normand**, *Automatisation du contrôle de qualité des systèmes d'imagerie embarquée en radiothérapie externe*, en français, Saint Malo, mai 2007.
- [15] Matthias Beck, Sinai Robins, *Computing the Continuous Discretely : Integer-Point Enumeration in Polyhedra*, sér. Undergraduate texts in mathematics. Springer, 2007, ISBN : 978-0-387-29139-0.
- [16] Yassine H. Ben Hdech, Stéphane Beaumont, Jeanpierre Guédon, Tarraf Torfeh, « New method to perform dosimetric quality control of treatment planning system using PENELOPE Monte Carlo and anatomical digital test objects », dans *SPIE Medical Imaging 2010 : Physics of Medical Imaging*, t. 7622, San Diego, CA, USA : SPIE, fév. 2010. DOI : 10.1117/12.844104.
- [17] Richard E. Blahut, *Algebraic codes for data transmission*. Cambridge University Press, 2003, ISBN : 9780521553742.
- [18] Pauline Bléry, Yves Amouriq, Jeanpierre Guédon, Paul Pilet, **Nicolas Normand**, Nicolas Durand, Florent Espitalier, Aurore Arlicot, Olivier Malard, Pierre Weiss, « Microarchitecture of irradiated bone : comparison with healthy bone », dans *SPIE Medical Imaging 2012 : Biomedical Applications in Molecular, Structural, and Functional Imaging*, t. 8317, San Diego, CA, USA : SPIE, fév. 2012, pages. DOI : 10.1117/12.911375.
- [19] Alexandra Bonnassie, Françoise Peyrin, Dominique Attali, « Shape description of three-dimensional images based on medial axis », dans *International Conference on Pattern Recognition*, IEEE, t. 3, oct. 2001, p. 931–934. DOI : 10.1109/ICIP.2001.958277.
- [20] Gunilla Borgefors, « Centres of maximal discs in the 5-7-11 distance transform », dans *Proc. 8th Scandinavian Conf. on Image Analysis*, Tromsø, Norway, mai 1993, p. 105–111.
- [21] —, « Distance transformations in arbitrary dimensions », *Computer Vision, Graphics, and Image Processing*, t. 27, n° 3, p. 321–345, sept. 1984. DOI : 10.1016/0734-189X(84)90035-5.

- [22] —, « Distance transformations in digital images », *Computer Vision, Graphics, and Image Processing*, t. 34, n° 3, p. 344–371, juin 1986, ISSN : 0734-189X. DOI : 10.1016/S0734-189X(86)80047-0.
- [23] —, « Weighted digital distance transforms in four dimensions », *Discrete Applied Mathematics*, t. 125, n° 1, p. 161–176, jan. 2003. DOI : 10.1016/S0166-218X(02)00229-9.
- [24] Georges Bouligand, « Problèmes connexes de la notion d’enveloppe de M. Georges Durand », en français, *Comptes rendus hebdomadaires des séances de l’Académie des sciences*, t. 189, p. 446–447, 29 sept. 1929.
- [25] Donald R. Chand, Sham S. Kapur, « An algorithm for convex polytopes », *Journal of the Association for Computing Machinery*, t. 17, n° 1, p. 78–86, jan. 1970. DOI : 10.1145/321556.321564.
- [26] Shekhar Chandra, Imants Dzintars Svalbe, Jeanpierre Guédon, Andrew Maurice Kingston, **Nicolas Normand**, « Recovering missing slices of the discrete fourier transform using ghosts », *IEEE Transactions on Image Processing*, t. 21, n° 10, p. 4431–4441, oct. 2012. DOI : 10.1109/TIP.2012.2206033, 5-year IF : 4.205 (ISI 2010), ERA 17873 A\*.
- [27] Su Chen, Robert M. Haralick, « Recursive erosion, dilation, opening, and closing transforms », *IEEE Transactions on Image Processing*, t. 4, n° 3, p. 335–345, mar. 1995. DOI : 10.1109/83.366481.
- [28] David Coeurjolly, Annick Montanvert, « Optimal separable algorithms to compute the reverse Euclidean distance transformation and discrete medial axis in arbitrary dimension », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, t. 29, n° 3, p. 437–448, mar. 2007. DOI : 10.1109/TPAMI.2007.54.
- [29] David Coeurjolly, **Nicolas Normand**, « Discrete geometry and projections », dans *The Mojette Transform : Theory and Applications*, J. GUÉDON, éd., Wiley-ISTE, 9 jan. 2009, chap. 1, p. 3–20, ISBN : 9781848210806.
- [30] Per-Erik Danielsson, « Euclidean distance mapping », *Computer Graphics, and Image Processing*, t. 14, n° 3, p. 227–248, nov. 1980. DOI : [http://dx.doi.org/10.1016/0146-664X\(80\)90054-4](http://dx.doi.org/10.1016/0146-664X(80)90054-4).
- [31] Partha Pratim Das, P. P. Chakrabarti, B. N. Chatterji, « Distance functions in digital geometry », *Information Sciences*, t. 42, n° 2, p. 113–136, juil. 1987.
- [32] Alain Daurat, **Nicolas Normand**, « Transformation et reconstruction par projections », en français, dans *Géométrie discrète et images numériques*, sér. traité IC2, D. COEURJOLLY, A. MONTANVERT et J.-M. CHASSERY, éd., Hermès, 2007, chap. 10, p. 239–251, ISBN : 978-2-7462-1643-3.
- [33] Olivier Déforges, Marie Babel, Joseph Ronsin, Laurent Bédard, **Nicolas Normand**, Benoît Parrein, Jeanpierre Guédon, « Le LAR aux Mojettes », en français, dans *CORESAs*, Lille, mai 2004.
- [34] Olivier Déforges, Virginie Fresse, **Nicolas Normand**, « Processeur temps-réel générique d’opérateurs morphologiques multi-niveaux de gris », en français, dans *Workshop sur l’Adéquation Algorithme-Architecture*, jan. 2000, p. 71–77.
- [35] Olivier Déforges, **Nicolas Normand**, « A generic systolic processor for real time grayscale morphology », dans *International Conference on Acoustic, Speech and Signal Processing*, t. 6, juin 2000, p. 3331–3334. DOI : 10.1109/ICASSP.2000.860113.

- [36] —, « A systolic preprocessor for online morphologic operations », dans *SPIE conf. on Parallel and Distributed Methods for Image Processing*, t. 3166, sept. 1997, p. 212–223. DOI : 10.1117/12.279619.
- [37] —, « Recursive morphological operators for gray image processing. application in granulometry analysis », dans *ICIP*, t. 2, Santa Barbara, CA, USA, oct. 1997, p. 672–675. DOI : 10.1109/ICIP.1997.638585.
- [38] Olivier Déforges, **Nicolas Normand**, Marie Babel, « Fast recursive grayscale morphology operators : from the algorithm to the pipeline architecture », *Journal of Real-Time Image Processing*, p. 1–10, juil. 2010. DOI : 10.1007/s11554-010-0171-8.
- [39] Éloïse Denis, Stéphane Beaumont, Jeanpierre Guédon, **Nicolas Normand**, « Method for generating digital test objects », US patent 8160327, 24 oct. 2008.
- [40] Éloïse Denis, Stéphane Beaumont, Jeanpierre Guédon, **Nicolas Normand**, Tarraf Torfeh, « Automatic quality control of digitally reconstructed radiograph computation and comparison with standard methods », dans *SPIE Medical Imaging 2007 : Physics of Medical Imaging*, t. 6510, San Diego, CA, USA, 17 fév. 2007. DOI : 10.1117/12.708193.
- [41] Éloïse Denis, Stéphane Beaumont, Jeanpierre Guédon, Tarraf Torfeh, **Nicolas Normand**, Norbert Ailleres, « New automatic quality control methods for geometrical treatment planning system tools in external conformal radiotherapy », dans *SPIE Medical Imaging 2008 : Physics of Medical Imaging*, t. 6913, San Diego, CA, USA : SPIE, fév. 2008, 12 pages. DOI : 10.1117/12.768670.
- [42] —, *Nouvelle méthode automatique de contrôle de qualité des systèmes de planification géométrique des traitements en radiothérapie externe conformationnelle*, en français, Marseille, juin 2008.
- [43] Éloïse Denis, Jeanpierre Guédon, Stéphane Beaumont, **Nicolas Normand**, « Discrete and continuous description of a three dimensional scene for quality control of radiotherapy treatment planning systems », dans *SPIE Medical Imaging 2006 : Physics of Medical Imaging*, t. 6142, San Diego, CA, USA, fév. 2006, 12 pages. DOI : 10.1117/12.650571.
- [44] Emeric Deutsch, *The On-Line Encyclopedia of Integer Sequences, Number of hex trees with n edges and having no nonroot nodes of outdegree 2*, N. J. A. SLOANE et S. PLOUFFE, éd., A126184, déc. 2006.
- [45] Elena Deza, Michel Marie Deza, Michel Marie Deza, Elena Deza, « Encyclopedia of distances », dans *Encyclopedia of Distances*, Springer Berlin Heidelberg, 2009, p. 1–583, ISBN : 978-3-642-00234-2. DOI : 10.1007/978-3-642-00234-2\_1.
- [46] William Donnelly, « Per-pixel displacement mapping with distance functions », dans *GPU Gems 2*, NVidia, 2005, p. 123–136.
- [47] Georges Durand, « Sur une généralisation des surfaces convexes », en français, thèse de doct., Faculté des sciences de Paris, 1931.
- [48] Dietmar Eggeman, Jeanpierre Guédon, **Nicolas Normand**, « Transferts de documents multimédias sur réseaux à qualité de service », dans *CORESA*, Lannion, 1998.
- [49] Pierre Évenou, « Serveurs Mojette distribués », Mémoire CNAM, 9 juil. 2003.
- [50] John Farey, « On a curious property of vulgar fractions », *The Philosophical Magazine and Journal*, t. 47, p. 385–386, mai 1816.
- [51] Bakir Farhi, « An explicit formula generating the non-Fibonacci numbers », *Arxiv preprint arXiv :1105.1127*, 2011.

- [52] H. Fayad, Jeanpierre Guédon, Imants Dzintars Svalbe, Yves J. Bizais, **Nicolas Normand**, « Applying Mojette discrete Radon transforms to classical tomographic data », dans *SPIE Medical Imaging 2008 : Physics of Medical Imaging*, t. 6913, SPIE, fév. 2008. DOI : 10.1117/12.770478.
- [53] Herbert Federer, « Curvature measures », *Transactions of the American Mathematical Society*, t. 93, n° 3, p. 418–491, oct. 1959. DOI : 10.1090/S0002-9947-1959-0110078-1.
- [54] Céline Fouard, Grégoire Malandain, « 3-D chamfer distances and norms in anisotropic grids », *Image and Vision Computing*, t. 23, n° 2, p. 143–158, fév. 2004. DOI : 10.1016/j.imavis.2004.06.009.
- [55] Herbert Freeman, « On the encoding of arbitrary geometric configurations », *IRE Transactions on Electronic Computers*, t. 10, n° 2, p. 260–268, juin 1961. DOI : 10.1109/TEC.1961.5219197.
- [56] Wei Gong, « On decomposition of structure element for mathematical morphology », dans *International Conference on Pattern Recognition*, nov. 1988, p. 836–838. DOI : 10.1109/ICPR.1988.28374.
- [57] Peter M. Gruber, *Convex and discrete geometry*, sér. Grundlehren der mathematischen Wissenschaften. Springer, 2007, t. 336, ISBN : 9783540711322. DOI : 10.1007/978-3-540-71133-9.
- [58] Branko Grünbaum, *Convex Polytopes*, 2<sup>e</sup> éd., G. M. ZIEGLER, éd., sér. Graduate Texts in Mathematics. Springer, 2003 (1967), t. 221, ISBN : 978-0-387-00424-2.
- [59] Ralph Guderlei, Simone Klenk, Johannes Mayer, Volker Schmidt, Evgueni Spodarev, « Algorithms for the computation of the Minkowski functionals of deterministic and random polyconvex sets », *Image and Vision Computing*, t. 25, n° 4, p. 464–474, avr. 2007, International Symposium on Mathematical Morphology 2005, ISSN : 0262-8856. DOI : 10.1016/j.imavis.2006.07.019.
- [60] J. GUÉDON, éd., *The Mojette Transform : Theory and Applications*. Wiley-ISTE, 9 jan. 2009, ISBN : 9781848210806.
- [61] Jeanpierre Guédon, « Un tomographe discret : l’opérateur Mojette. Applications en image et réseaux », Habilitation à Diriger des Recherches, École polytechnique de l’Université de Nantes, Nantes, 18 déc. 2001.
- [62] Jeanpierre Guédon, **Nicolas Normand**, « Direct Mojette transform », dans *The Mojette Transform : Theory and Applications*, J. GUÉDON, éd., Wiley-ISTE, 9 jan. 2009, chap. 3, p. 41–66, ISBN : 9781848210806.
- [63] —, « Mojette transform : applications for image analysis and coding », dans *Visual Communications and Image Processing*, t. 3024, fév. 1997, p. 873–884. DOI : 10.1117/12.263299.
- [64] —, « Reconstructability with the inverse Mojette transform », dans *The Mojette Transform : Theory and Applications*, J. GUÉDON, éd., Wiley-ISTE, 9 jan. 2009, chap. 4, p. 67–83, ISBN : 9781848210806.
- [65] —, « Spline Mojette transform. applications in tomography and communications », dans *EUSIPCO 2002*, t. II, Toulouse, sept. 2002, p. 407–410.
- [66] —, « The Mojette transform : the first ten years », dans *Discrete Geometry for Computer Imagery 2005*, É. ANDRES, G. DAMIAND et P. LIENHARDT, éd., sér. Lecture Notes in Computer Science, t. 3429, Poitiers, France : Springer Berlin / Heidelberg, avr. 2005, p. 79–91, ISBN : 978-3-540-25513-0. DOI : 10.1007/978-3-540-31965-8\_8, ERA 43320 C.

- [67] Jeanpierre Guédon, **Nicolas Normand**, Sandrine Lecoq, « Transformation Mojette en 3D : mise en œuvre et application en synthèse d'images », en français, dans *GRETSI*, sept. 1999, p. 427–430. DOI : 2042/13101.
- [68] Jeanpierre Guédon, **Nicolas Normand**, Benoît Parrein, Christophe Pouliquen, « Distributed image transmission and storage on internet system », dans *ACIDCA 2000*, Monastir, mar. 2000, p. 164–169.
- [69] Jeanpierre Guédon, **Nicolas Normand**, Pierre Verbert, Benoît Parrein, Florent Autrusseau, « Load-balancing and scalable multimedia distribution using the Mojette transform », dans *ITCOM*, SPIE, t. 4519, Denver, août 2001, p. 226–234. DOI : 10.1117/12.434272.
- [70] Jeanpierre Guédon, Benoît Parrein, **Nicolas Normand**, « Internet distributed image information system », *Integrated Computer-Aided Engineering*, t. 8, n° 3, p. 205–214, août 2001.
- [71] Jeanpierre Guédon, Myriam Servières, Stéphane Beaumont, **Nicolas Normand**, « Medical software control quality using the 3D Mojette projector », dans *ISBI*, avr. 2004, p. 836–839. DOI : 10.1109/ISBI.2004.1398668.
- [72] András Hajdu, Lajos Hajdu, « Approximating the Euclidean distance using non-periodic neighbourhood sequences », *Discrete Mathematics*, t. 283, n° 1-3, p. 101–111, juin 2004. DOI : 10.1016/j.disc.2003.12.016.
- [73] Lajos Hajdu, « Unique reconstruction of bounded sets in discrete tomography », dans *Proceedings of the Workshop on Discrete Tomography and its Applications*, G. T. HERMAN et A. KUBA, édés., sér. Electronic Notes in Discrete Mathematics, t. 20, New York, juin 2005, p. 15–25. DOI : 10.1016/j.endm.2005.04.002.
- [74] Lajos Hajdu, R. Tijdeman, « Algebraic aspects of discrete tomography », *Journal für die reine und angewandte Mathematik (Crelles Journal)*, t. 2001, n° 534, p. 119–128, avr. 2001. DOI : 10.1515/crll.2001.037.
- [75] Haros, « Tables pour évaluer une fraction ordinaire avec autant de décimales qu'on voudra ; et pour trouver la fraction ordinaire la plus simple, et qui approche sensiblement d'une fraction décimale », en français, *Journal de l'École Polytechnique*, t. 4, n° cahier XI, p. 364–368, Messidor an X.
- [76] John C. Hart, « Sphere tracing : a geometric method for the antialiased ray tracing of implicit surfaces », *The Visual Computer*, t. 12, n° 10, p. 527–545, déc. 1996, ISSN : 0178-2789. DOI : 10.1007/s003710050084.
- [77] Martin Henk, Jürgen Richter-Gebert, Günter M. Ziegler, « Basic properties of convex polytopes », dans *Handbook of discrete and computational geometry*, sér. Discrete Mathematics and its Applications, J. E. GOODMAN et J. O'ROURKE, édés., 2<sup>e</sup> éd., Boca Raton, FL, USA : CRC Press, Inc., 2004 (1997), chap. 16, ISBN : 0-8493-8524-5.
- [78] Marcel Herk, « A fast algorithm for local minimum and maximum filters on rectangular and octagonal kernels », *Pattern Recognition Letters*, t. 13, n° 7, p. 517–521, juil. 1992, ISSN : 0167-8655. DOI : 10.1016/0167-8655(92)90069-C.
- [79] Jérôme Hulin, « Axe médian discret : propriétés arithmétiques et algorithmes », en français, Thèse de doctorat, Université de la Méditerranée, 20 nov. 2009.
- [80] Olivier Jacquier, Jeanpierre Guédon, **Nicolas Normand**, Simon Colombié, Benoît Parrein, « Procédé sécurisé de fourniture de documents payants via un réseau de communication », Brevet FR2858497, juil. 2003.

- [81] R. A. Jarvis, « On the identification of the convex hull of a finite set of points in the plane », *Information Processing Letters*, t. 2, n° 1, p. 18–21, mar. 1973, ISSN : 0020-0190. DOI : 10.1016/0020-0190(73)90020-3.
- [82] Liang Ji, Jim Piper, Jing-Yan Tang, « Erosion and dilation of binary images by arbitrary structuring elements using interval coding », *Pattern Recognition Letters*, t. 9, n° 3, p. 201–209, avr. 1989. DOI : 10.1016/0167-8655(89)90055-X.
- [83] Myron Bernard Katz, *Questions of uniqueness and resolution in reconstruction from projections*, sér. Lecture Notes in Biomath. Berlin, RFA : Springer-Verlag, 1978, t. 26, ISBN : 3-540-09087-8.
- [84] Andrew Maurice Kingston, **Nicolas Normand**, Benoît Parrein, Imants Dzintars Svalbe, « An involutory finite Radon transform and its application as a maximum distance separable code », In preparation, 2010.
- [85] Simone Klenk, Volker Schmidt, Evgueni Spodarev, « A new algorithmic approach to the computation of Minkowski functionals of polyconvex sets », *Computational Geometry*, t. 34, n° 3, p. 127–148, juil. 2006, ISSN : 0925-7721. DOI : 10.1016/j.comgeo.2006.02.002.
- [86] Joachim Lambek, Leo Moser, « Inverse and complementary sequences of natural numbers », *The American Mathematical Monthly*, t. 61, n° 7, p. 454–458, août 1954, ISSN : 00029890. DOI : 10.2307/2308078.
- [87] Patrick Le Callet, Vincent Ricordel, **Nicolas Normand**, « Expertise Courant Multimédia », en français, Rapport d’expertise, 20 oct. 2004.
- [88] Chuanlin Liu, Jeanpierre Guédon, Imants Dzintars Svalbe, **Nicolas Normand**, Yves Amouriq, « Line Mojette ternary reconstructions and ghosts », dans *Advances in Image Analysis and Applications*, R. P. BARNEVA, V. E. BRIMKOV, K. KOROUTCHEV et E. KORUTCHEVA, édés., Madrid, mai 2011, 12 pages, ISBN : 978-981-08-7923-5, ERA 43761 C.
- [89] Franck Lusteau, Dominique Guillerme, Antoine Ciftsi, **Nicolas Normand**, Jeanpierre Guédon, « MUSHAN : multimedia ship area network », dans *Internet Multimedia Management Systems III*, SPIE, t. 4862, Boston, juil. 2002, p. 296–304. DOI : 10.1117/12.473045.
- [90] Georges Matheron, « Étude théorique des granulométries », en français, École des Mines de Paris, Note Géostatistique, déc. 1964.
- [91] —, « La formule de Steiner pour les érosions », en français, *Journal of Applied Probability*, t. 15, n° 1, p. 126–135, mar. 1978, ISSN : 00219002.
- [92] —, *Random Sets and Integral Geometry*. New York : Wiley, 1975.
- [93] František Matúš, Jan Flusser, « Image representation via a finite Radon transform », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, t. 15, n° 10, p. 996–1006, oct. 1993, ISSN : 0162-8828. DOI : 10.1109/34.254058.
- [94] A. Meijster, J. B. T. M. Roerdink, W. H. Hesselink, « A general algorithm for computing distance transforms in linear time », dans *Mathematical Morphology and its Applications to Image and Signal Processing*, sér. Computational Imaging and Vision, J. GOUTSIAS, L. VINCENT et D. S. BLOOMBERG, édés., t. 18, Springer US, 2002, p. 331–340, ISBN : 978-0-306-47025-7. DOI : 10.1007/0-306-47025-X\_36.
- [95] Ugo Montanari, « A method for obtaining skeletons using a quasi-Euclidean distance », *Journal of the ACM*, t. 15, n° 4, p. 600–624, oct. 1968. DOI : 10.1145/321479.321486.
- [96] Jean-Marie Morvan, *Generalized Curvatures*, sér. Geometry and Computing. 2008, t. 2. DOI : 10.1007/978-3-540-73792-6.

- [97] Benedek Nagy, « Distance with generalized neighbourhood sequences in  $nD$  and  $\infty D$  », *Discrete Applied Mathematics*, t. 156, n° 12, p. 2344–2351, juin 2008. DOI : 10.1016/j.dam.2007.10.017.
- [98] **Nicolas Normand**, *Algorithmes rapides pour la morphologie mathématique binaire*, en français, Réunion GDR ISIS Thème E, 27 mai 2003.
- [99] —, « Convex structuring element decomposition for single scan binary mathematical morphology », dans *Discrete Geometry for Computer Imagery 2003*, I. NYSTRÖM, G. SANNITI DI BAJA et S. SVENSSON, édés., sér. Lecture Notes in Computer Science, t. 2886, Naples, Italy : Springer Berlin / Heidelberg, nov. 2003, p. 154–163, ISBN : 3-540-20499-7. DOI : 10.1007/978-3-540-39966-7\_14, ERA 43320 C.
- [100] —, *Discrete distances, paths, disks and polytopes*, Séminaire de l'École de Physique, Monash University, 16 nov. 2009.
- [101] —, « Erasure coding with the Mojette and Finite Radon Transforms », Journées du groupe de travail géométrie discrète, Clermont-Ferrand, 14 oct. 2011.
- [102] **Nicolas Normand**, « Représentation d'images et distances discrètes basées sur les éléments structurants à deux pixels », en français, Thèse de doctorat, thèse de doct., Université de Nantes, jan. 1997.
- [103] **Nicolas Normand**, Florent Autrusseau, Jeanpierre Guédon, « Biométrie, reconnaissance de visages », en français, Rapport d'expertise, 10 oct. 2005.
- [104] **Nicolas Normand**, Olivier Déforges, Patrick Le Callet, Aurélien Chiron, « Algorithme de transformation en distance généralisée en un balayage, mises en œuvre séquentielle et vectorielle », dans *GRETSI*, sept. 2003. DOI : 2042/13753.
- [105] **Nicolas Normand**, Pierre Evenou, « Medial axis lookup table and test neighborhood computation for 3D chamfer norms », *Pattern Recognition*, t. 42, n° 10, p. 2288–2296, oct. 2009. DOI : 10.1016/j.patcog.2008.11.014, 5-year IF : 3.453 (ISI 2009), ERA 4503 A\*.
- [106] **Nicolas Normand**, Pierre Évenou, « Medial axis LUT computation for chamfer norms using H-polytopes », dans *Discrete Geometry for Computer Imagery 2008*, D. COEURJOLLY, I. SIVIGNON, L. TOUGNE et F. DUPONT, édés., sér. Lecture Notes in Computer Science, t. 4992, Lyon, France : Springer Berlin / Heidelberg, avr. 2008, p. 189–200. DOI : 10.1007/978-3-540-79126-3\_18, ERA 43320 C.
- [107] **Nicolas Normand**, Jeanpierre Guédon, « Analyse de la solution Trident Media Stop », Rapport d'expertise, juil. 2003.
- [108] —, « Applications de la transformation Mojette », en français, dans *Géométrie discrète et images numériques*, sér. traité IC2, D. COEURJOLLY, A. MONTANVERT et J.-M. CHASSERY, édés., Hermès, 2007, chap. 16, p. 337–347, ISBN : 978-2-7462-1643-3.
- [109] —, « La transformée Mojette : une représentation redondante pour l'image », en français, *Comptes Rendus de l'Académie des Sciences - Série I - Mathématiques*, t. 326, n° 1, p. 123–126, jan. 1998. DOI : 10.1016/S0764-4442(97)82724-3, ERA 107 B.
- [110] **Nicolas Normand**, Jeanpierre Guédon, Olivier Philippé, Dominique Barba, « Controlled redundancy for image coding and high-speed transmission », dans *Visual Communications and Image Processing*, t. 2727, Orlando, FL, mar. 1996, p. 1070–1081. DOI : 10.1117/12.233180.
- [111] **Nicolas Normand**, Jeanpierre Guédon, Francois Terrien, Dominique Barba, « Source-channel coding for medical imaging », dans *PCS*, Melbourne, mar. 1996, p. 329–334.



- [112] **Nicolas Normand**, Andrew Maurice Kingston, Pierre Évenou, « A geometry driven reconstruction algorithm for the Mojette transform », dans *Discrete Geometry for Computer Imagery 2006*, A. KUBA, L. G. NYÚL et K. PALÁGYI, édés., sér. Lecture Notes in Computer Science, t. 4245, Szeged, Hungary : Springer Berlin / Heidelberg, oct. 2006, p. 122–133. DOI : 10.1007/11907350\_11, ERA 43320 C.
- [113] **Nicolas Normand**, Benoît Parrein, « Description multiple et codage source-canal sur un canal à effacement avec la transformation mojette », en français, dans *GRETSI*, sept. 1999, p. 869–872. DOI : 2042/13102.
- [114] **Nicolas Normand**, Myriam Servières, Jeanpierre Guédon, « How to obtain a lattice basis from a discrete projected space », dans *Discrete Geometry for Computer Imagery 2005*, É. ANDRES, G. DAMIAND et P. LIENHARDT, édés., sér. Lecture Notes in Computer Science, t. 3429, Poitiers, France : Springer Berlin / Heidelberg, avr. 2005, p. 153–160, ISBN : 978-3-540-25513-0. DOI : 10.1007/978-3-540-31965-8\_15, ERA 43320 C.
- [115] **Nicolas Normand**, Robin Strand, Pierre Evenou, « Digital distances and integer sequences », dans *Discrete Geometry for Computer Imagery 2013*, sér. Lecture Notes in Computer Science, Sevilla, 20–22 mar. 2013, Accepté, ERA 43320 C.
- [116] **Nicolas Normand**, Robin Strand, Pierre Evenou, Aurore Arlicot, « A minimal delay neighborhood-sequence distance transform », *Image Processing On Line*, 2012, revision.
- [117] —, « Minimal-delay distance transform for neighborhood-sequence distances in 2D and 3D », *Computer Vision and Image Understanding*, 2012. DOI : 10.1016/j.cviu.2012.08.015, Accepté.
- [118] —, « Path-based distance with varying weights and neighborhood sequences », dans *Discrete Geometry for Computer Imagery 2011*, I. DEBLED-RENNESSON, É. DOMENJOUR, B. KERAUTRET et P. EVEN, édés., sér. Lecture Notes in Computer Science, t. 6607, Nancy, France : Springer Berlin / Heidelberg, avr. 2011, p. 199–210, ISBN : 978-3-642-19866-3. DOI : 10.1007/978-3-642-19867-0\_17, ERA 43320 C.
- [119] **Nicolas Normand**, Imants Dzintars Svalbe, Pierre Evenou, Andrew Maurice Kingston, « Inverse Mojette transform algorithms », dans *The Mojette Transform : Theory and Applications*, J. GUÉDON, éd., Wiley-ISTE, 9 jan. 2009, chap. 5, p. 85–113, ISBN : 9781848210806.
- [120] **Nicolas Normand**, Imants Dzintars Svalbe, Benoît Parrein, Andrew Maurice Kingston, « Erasure coding with the finite Radon transform », dans *IEEE Wireless Communications & Networking Conference*, Sydney, avr. 2010, p. 1–6. DOI : 10.1109/WCNC.2010.5506385, ERA 43061 B.
- [121] **Nicolas Normand**, Christian Viard-Gaudin, « A background based adaptive page segmentation algorithm », dans *ICDAR*, Montréal, août 1995, p. 138–141. DOI : 10.1109/ICDAR.1995.598961.
- [122] —, « A two-dimensional bar code reader », dans *International Conference on Pattern Recognition*, IAPR, Jerusalem, oct. 1994, p. 201–203. DOI : 10.1109/ICPR.1994.577158.
- [123] —, « Lecture de codes-barres par vision 2D », en français, *Technique et Science Informatiques*, t. 13, n° 4, p. 493–523, 1994.
- [124] **Nicolas Normand**, Yu Zhang, Imants Dzintars Svalbe, « Granulometry estimation from a neighborhood-sequence distance map », dans *International Symposium on Mathematical Morphology*, Uppsala, Sweden, mai 2013, En préparation.

- [125] S. Ohn, « Morphological decomposition of convex polytopes and its application in discrete image space », dans *ICIP*, t. 2, nov. 1994, p. 560–564. DOI : 10.1109/ICIP.1994.413633.
- [126] A. Ostrowski, J. Hyslop, A. C. Aitken, « Solutions to problem 3173 », *The American Mathematical Monthly*, t. 34, n° 3, p. 159–160, mar. 1927.
- [127] Benoît Parrein, Fadi Boulos, **Nicolas Normand**, Pierre Evenou, « Communication, networks and storage », dans *The Mojette Transform : Theory and Applications*, J. GUÉDON, éd., Wiley-ISTE, 9 jan. 2009, chap. 7, p. 151–182, ISBN : 9781848210806.
- [128] Benoît Parrein, Lionel Chatelier, **Nicolas Normand**, Jeanpierre Guédon, « FEC comparison for multimedia wireless distribution », dans *ASWN 2002*, IEEE, Paris, juil. 2002, p. 140–144.
- [129] Benoît Parrein, Jeanpierre Guédon, **Nicolas Normand**, « Multimedia packet transport : multiple layers or description ? », dans *Packet Video*, Nantes, avr. 2003.
- [130] Benoît Parrein, **Nicolas Normand**, « Multiplexage de flux multimédias sur IP par description multiple Mojette », en français, dans *CORESA*, Sophia Antipolis, juin 1999.
- [131] Benoît Parrein, **Nicolas Normand**, Dominique Barba, « Description multiple par transformation de Radon discrète exacte », en français, dans *GRETSI*, Toulouse, sept. 2001. DOI : 2042/13410.
- [132] Benoît Parrein, **Nicolas Normand**, Majd Ghareeb, Giulio D’Ippolito, Federica Battisti, « Finite Radon coding for content delivery over hybrid client-server and P2P architecture », dans *International Symposium on Communications, Control, and Signal Processing*, mai 2012. DOI : 10.1109/ISCCSP.2012.6217794.
- [133] Benoît Parrein, **Nicolas Normand**, Jeanpierre Guédon, « Multimedia forward error correcting codes for wireless LAN », *Annals of telecommunications*, t. 58, n° 3-4, p. 448–463, mar. 2003.
- [134] —, « Multiple description coding using exact discrete Radon transform », dans *Data Compression Conference*, IEEE, Snowbird, mar. 2001, p. 508. DOI : 10.1109/DCC.2001.10031.
- [135] —, « Système de transmission par codage prioritaire », en français, dans *CORESA*, Poitiers, oct. 2000.
- [136] Benoît Parrein, **Nicolas Normand**, Patrick Le Callet, « Transmission prioritaire JPEG2000 sur lien sans fil », en français, dans *GRETSI*, Louvain-la-Neuve, Belgique, sept. 2005.
- [137] Benoît Parrein, **Nicolas Normand**, Patrick Le Callet, Jeanpierre Guédon, *Transmission prioritaire d’images fixes pilotée par un critère de qualité visuelle*, en français, Réunion GDR ISIS Thème D, jan. 2006.
- [138] Benoît Parrein, Pierre Verbert, **Nicolas Normand**, Jeanpierre Guédon, « Scalable multiple descriptions on packets networks via the n-dimensional Mojette transform », dans *ITCOM*, SPIE, t. 4524, Denver, août 2001, p. 243–252. DOI : 10.1117/12.434360.
- [139] John L. Pfaltz, Azriel Rosenfeld, « Computer representation of planar regions by their skeletons », *Communications of the ACM*, t. 10, n° 2, p. 119–122, fév. 1967. DOI : 10.1145/363067.363120.
- [140] Georg Pick, « Geometrisches zur Zahlenlehre », en allemand, *Sitzungsberichte des Deutschen Naturwissenschaftlich-Medicinischen Vereines für Böhmen "Lotos" in Prag*, t. 47, p. 311–319, 1899.

- [141] Ioannis Pitas, Anastasios N. Venetsanopoulos, « Morphological shape decomposition », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, t. 12, n° 1, p. 38–45, jan. 1990. DOI : 10.1109/34.41382.
- [142] Chris Pudney, « Distance-ordered homotopic thinning : a skeletonization algorithm for 3D digital images », *Computer Vision and Image Understanding*, t. 72, n° 3, p. 404–413, déc. 1998. DOI : 10.1006/cviu.1998.0680.
- [143] Johann Radon, « On the determination of functions from their integral values along certain manifolds », trad. par P. C. PARKS, *IEEE Transactions on Medical Imaging*, t. 5, n° 4, p. 170–176, déc. 1986 (1917), ISSN : 0278-0062. DOI : 10.1109/TMI.1986.4307775.
- [144] Éric Rémy, « Normes de chanfrein et axe médian dans le volume discret », en français, Thèse de doctorat, Université de la Méditerranée, 17 déc. 2001. (visité le 27/05/2012).
- [145] Éric Rémy, Édouard Thiel, « Computing 3D medial axis for chamfer distances », dans *Discrete Geometry for Computer Imagery*, sér. Lecture Notes in Computer Science, t. 1953, Uppsala, Sweden, déc. 2000, p. 418–430. DOI : 10.1007/3-540-44438-6\_34.
- [146] —, « Medial axis for chamfer distances : computing look-up tables and neighbourhoods in 2D or 3D », *Pattern Recognition Letters*, t. 23, n° 6, p. 649–661, avr. 2002. DOI : 10.1016/S0167-8655(01)00141-6.
- [147] Azriel Rosenfeld, John L. Pfaltz, « Distances functions on digital pictures », *Pattern Recognition*, t. 1, n° 1, p. 33–61, juil. 1968. DOI : 10.1016/0031-3203(68)90013-7.
- [148] —, « Sequential operations in digital picture processing », *Journal of the ACM*, t. 13, n° 4, p. 471–494, oct. 1966, ISSN : 0004-5411. DOI : 10.1145/321356.321357.
- [149] Toyofumi Saito, Jun-Ichiro Toriwaki, « New algorithms for Euclidean distance transformation of an  $n$ -dimensional digitized picture with applications », *Pattern Recognition*, t. 27, n° 11, p. 1551–1565, nov. 1994. DOI : 10.1016/0031-3203(94)90133-3.
- [150] Rolf Schneider, « Parallelmengen mit vielfachheit und Steiner-formeln », en allemand, *Geometriae Dedicata*, t. 9, n° 1, p. 111–127, mar. 1980, ISSN : 0046-5755. DOI : 10.1007/BF00156479.
- [151] Jean Serra, *Image analysis and mathematical morphology*. Academic Press London, 1982.
- [152] Myriam Servières, Jeanpierre Guédon, **Nicolas Normand**, « A discrete tomography approach to PET reconstruction », dans *Fully 3D Reconstruction in Radiology and Nuclear Medicine*, Y. J. BIZAIS, éd., Saint Malo, juil. 2003.
- [153] Myriam Servières, Jeanpierre Guédon, **Nicolas Normand**, Yves J. Bizais, « Mojette discrete tomography », dans *The Mojette Transform : Theory and Applications*, J. GUÉDON, éd., Wiley-ISTE, 9 jan. 2009, chap. 8, p. 183–214, ISBN : 9781848210806.
- [154] Myriam Servières, Jérôme Idier, **Nicolas Normand**, Jeanpierre Guédon, « Conjugate gradient Mojette reconstruction », dans *SPIE Medical Imaging 2005*, SPIE, t. 5747, San Diego, CA, USA, fév. 2005, p. 2067–2074. DOI : 10.1117/12.593399.
- [155] Myriam Servières, **Nicolas Normand**, « Opérateur Mojette 3D », en français, IRCCyN, Internal report, 2003.
- [156] Myriam Servières, **Nicolas Normand**, Yves J. Bizais, Jeanpierre Guédon, « Noise behavior for spline Mojette FBP reconstruction », dans *SPIE Medical Imaging 2005*, SPIE, t. 5747, San Diego, CA, USA, fév. 2005, p. 2100–2109. DOI : 10.1117/12.593390.
- [157] Myriam Servières, **Nicolas Normand**, Jeanpierre Guédon, « A Mojette transform approach to discrete medical tomography », dans *First Malaysia-France Regional Workshop on Image Processing in Vision*, 2003.

- [158] —, « Interpolation method for the Mojette transform », dans *SPIE Medical Imaging 2006*, t. 6142, San Diego, CA, USA, fév. 2006. DOI : 10.1117/12.651009.
- [159] Myriam Servières, **Nicolas Normand**, Jeanpierre Guédon, Yves J. Bizais, « The Mojette transform : discrete angles for tomography », dans *Discrete Tomography and its Applications*, G. T. HERMAN et A. KUBA, édés., sér. Electronic Notes in Discrete Mathematics, t. 20, Elsevier, juil. 2005, p. 587–606. DOI : 10.1016/j.endm.2005.05.086.
- [160] Myriam Servières, **Nicolas Normand**, Peggy Subirats, Jeanpierre Guédon, « Some links between continuous and discrete Radon transform », dans *SPIE Medical Imaging 2004*, SPIE, t. 5370, fév. 2004, p. 1961–1971. DOI : 10.1117/12.533472.
- [161] Lauro Snidaro, Gian Luca Foresti, « Real-time thresholding with euler numbers », *Pattern Recognition Letters*, t. 24, n° 9-10, p. 1533–1544, juin 2003, ISSN : 0167-8655. DOI : 10.1016/S0167-8655(02)00392-6.
- [162] Pierre Soille, Edmond J. Breen, Ronald Jones, « Recursive implementation of erosions and dilations along discrete lines at arbitrary angles », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, t. 18, n° 5, p. 562–567, mai 1996, ISSN : 0162-8828. DOI : 10.1109/34.494646.
- [163] Reinhard Johannes Steffens, « Mixed volumes, mixed Ehrhart theory and applications to tropical geometry and linkage configurations », thèse de doct., Johann Wolfgang Goethe-Universität, Frankfurt, 2009.
- [164] Jakob Steiner, « Über parallele flächen », en allemand, *Monatsbericht der Akademie der Wissenschaften zu Berlin*, p. 114–118, 1840.
- [165] Robin Strand, Gunilla Borgefors, « Distance transforms for three-dimensional grids with non-cubic voxels », *Computer Vision and Image Understanding*, t. 100, n° 3, p. 294–311, déc. 2005. DOI : 10.1016/j.cviu.2005.04.006.
- [166] Robin Strand, Benedek Nagy, Céline Fouard, Gunilla Borgefors, « Generating distance maps with neighbourhood sequences », dans *Discrete Geometry for Computer Imagery 2006*, A. KUBA, L. G. NYÚL et K. PALÁGYI, édés., sér. Lecture Notes in Computer Science, t. 4245, Szeged, Hungary : Springer Berlin / Heidelberg, oct. 2006, p. 295–307. DOI : 10.1007/11907350\_25, ERA 43320 C.
- [167] Robin Strand, **Nicolas Normand**, « Distance transform computation for digital distance functions », *Theoretical Computer Science*, t. 448, p. 80–93, 24 août 2012. DOI : 10.1016/j.tcs.2012.05.010, 5-year IF : 1.087 (ISI 2010), ERA 18143 A.
- [168] Peggy Subirats, Myriam Servières, **Nicolas Normand**, Jeanpierre Guédon, « Angular assessment of the Mojette Filtered Back Projection », dans *SPIE Medical Imaging 2004*, A. A. AMINI et A. MANDUCA, édés., SPIE, t. 5370, fév. 2004, p. 1951–1960. DOI : 10.1117/12.533483.
- [169] Imants Dzintars Svalbe, Nikesh Nazareth, **Nicolas Normand**, Shekhar Chandra, « On constructing minimal ghosts », dans *Digital Image Computing : Techniques and Applications 2010*, Sydney, Australia, déc. 2010, p. 276–281. DOI : 10.1109/DICTA.2010.56, ERA 42717 B.
- [170] Imants Dzintars Svalbe, **Nicolas Normand**, « Properties of minimal ghosts », dans *Discrete Geometry for Computer Imagery 2011*, I. DEBLED-RENNESON, É. DOMENJOUR, B. KERAUTRET et P. EVEN, édés., sér. Lecture Notes in Computer Science, t. 6607, Nancy, France : Springer Berlin / Heidelberg, avr. 2011, p. 417–428, ISBN : 978-3-642-19866-3. DOI : 10.1007/978-3-642-19867-0\_35, ERA 43320 C.

- [171] Stina Svensson, Gunilla Borgefors, « Digital distance transforms in 3d images using information from neighbourhoods up to  $5 \times 5 \times 5$  », *Computer Vision and Image Understanding*, t. 88, n° 1, p. 24–53, oct. 2002. DOI : 10.1006/cviu.2002.0976.
- [172] Édouard Thiel, « Géométrie des distances de chanfrein », en français, Habilitation à Diriger des Recherches, Université de la Méditerranée, 10 déc. 2001. (visité le 27/05/2012).
- [173] —, « Les distances de chanfrein en analyse d’images : fondements et applications », en français, Thèse de doctorat, Université Joseph Fourier, Grenoble 1, 21 sept. 1994. (visité le 27/05/2012).
- [174] Édouard Thiel, Annick Montanvert, « Chamfer masks : discrete distance functions, geometrical properties and optimization », dans *International Conference on Pattern Recognition*, t. III, août 1992, p. 244–247. DOI : 10.1109/ICPR.1992.201971.
- [175] Tarraf Torfeh, Stéphane Beaumont, David Bonnet, Yves Barbotteau, Jeanpierre Guédon, **Nicolas Normand**, Éloïse Denis, Pascal Fenoglietto, Norbert Ailleres, « Digital phantoms for an evaluation of software used for an automatic analysis of the Winston-Lutz test in image guided radiation therapy », dans *SPIE Medical Imaging 2008 : Physics of Medical Imaging*, t. 6913, San Diego, CA, USA : SPIE, fév. 2008, 11 pages. DOI : 10.1117/12.768668.
- [176] Tarraf Torfeh, Stéphane Beaumont, Jeanpierre Guédon, Yassine H. Ben Hdech, « 3D numerical test objects for the evaluation of a software used for an automatic analysis of a linear accelerator mechanical distortion », dans *SPIE Medical Imaging 2010 : Physics of Medical Imaging*, t. 7622, San Diego, CA, USA : SPIE, fév. 2010. DOI : 10.1117/12.844157.
- [177] Tarraf Torfeh, Stéphane Beaumont, Jeanpierre Guédon, **Nicolas Normand**, Éloïse Denis, « Software tools dedicated for an automatic analysis of the CT scanner quality control images », dans *SPIE Medical Imaging 2007 : Physics of Medical Imaging*, t. 6510, San Diego, CA, USA, 17 fév. 2007, 13 pages. DOI : 10.1117/12.707343.
- [178] Marc Van Droogenbroeck, « Algorithms for openings of binary and label images with rectangular structuring elements », dans *Mathematical morphology*, H. TALBOT et R. BEARE, édés., Sydney, Australia : CSIRO Publishing, avr. 2002, p. 197–207.
- [179] Ben J. H. Verwer, « Improved metrics in image processing applied to the hilditchskeleton », dans *International Conference on Pattern Recognition*, t. 1, nov. 1988, p. 137–142. DOI : 10.1109/ICPR.1988.28189.
- [180] —, « Local distances for distance transformations in two and three dimensions », *Pattern Recognition Letters*, t. 12, n° 11, p. 671–682, nov. 1991. DOI : 10.1016/0167-8655(91)90004-6.
- [181] Ben J. H. Verwer, Piet W. Verbeek, Simon T. Dekker, « An efficient uniform cost algorithm applied to distance transforms », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, t. 11, n° 4, p. 425–429, avr. 1989. DOI : 10.1109/34.19041.
- [182] Christian Viard-Gaudin, **Nicolas Normand**, Dominique Barba, « A bar code location algorithm using a two-dimensional approach », dans *ICDAR*, oct. 1993, p. 45–48. DOI : 10.1109/ICDAR.1993.395785.
- [183] Benjamin Villain, Matthew Davis, Julien Ridoux, Darryl Veitch, **Nicolas Normand**, « Probing the latencies of software timestamping », dans *International IEEE Symposium on Precision Clock Synchronization for Measurement, Control and Communication*, San Francisco, CA, USA, sept. 2012, p. 1–6. DOI : 10.1109/ISPCS.2012.6336623.

- [184] Luc M. Vincent, « Fast opening functions and morphological granulometries », *Proc. SPIE Image Algebra and Morphological Image Processing V*, t. 2300, n° 1, p. 253–267, juil. 1994. DOI : 10.1117/12.179208.
- [185] Xiaoli Wang, Gilles Bertrand, « An algorithm for a generalized distance transformation based on Minkowski operations », dans *International Conference on Pattern Recognition*, t. 2, nov. 1988, p. 1164–1168. DOI : 10.1109/ICPR.1988.28465.
- [186] —, « Some sequential algorithms for a generalized distance transformation based on Minkowski operations », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, t. 14, n° 11, p. 1114–1121, nov. 1992, ISSN : 0162-8828. DOI : 10.1109/34.166628.
- [187] Jianning Xu, « Decomposition of convex polygonal morphological structuring elements into neighborhood subsets », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, t. 13, n° 2, p. 153–162, fév. 1991. DOI : 10.1109/34.67644.
- [188] Masafumi Yamashita, Namio Honda, « Distance functions defined by variable neighborhood sequences », *Pattern Recognition*, t. 17, n° 5, p. 509–513, 1984. DOI : 10.1016/0031-3203(84)90048-7.
- [189] Shigeki Yokoi, Jun-Ichiro Toriwaki, Teruo Fukumura, « On generalized distance transformation of digitized pictures », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, t. 3, n° 4, p. 424–443, juil. 1981. DOI : 10.1109/TPAMI.1981.4767128.
- [190] Li-Ping Yuan, « A fast algorithm for size analysis of irregular pore areas », dans *SPIE/SPSE Vol. 1451, Nonlinear Image Processing II*, San Jose, CA, fév. 1991, p. 125–136. DOI : 10.1117/12.44320.
- [191] Yu Zhang, **Nicolas Normand**, *Analysis of composite material image using varying-weight distances*, Nantes, mai 2011.
- [192] Günter M. Ziegler, *Lectures on Polytopes (Graduate Texts in Mathematics)*. Springer, juil. 2001, ISBN : 038794365X.
- [193] Karel J. Zuiderveld, Anton H. J. Koning, Max A. Viergever, « Acceleration of ray-casting using 3-D distance transforms », dans *Visualization in Biomedical Computing '92*, SPIE, t. 1808, Chapel Hill, NC, USA, oct. 1992, p. 324–335. DOI : 10.1117/12.131088.

# Table des matières

<b>Curriculum Vitæ</b>	<b>3</b>
1 Identification . . . . .	3
2 Activités pédagogiques . . . . .	3
2.1 Déroulement de carrière . . . . .	3
2.2 Enseignements . . . . .	3
2.3 Responsabilités organisationnelles . . . . .	4
3 Responsabilités scientifiques . . . . .	4
4 Activités en Recherche . . . . .	5
4.1 Encadrement doctoral . . . . .	5
4.2 Collaborations scientifiques . . . . .	6
4.3 Brevets . . . . .	7
4.4 Expertises . . . . .	7
4.5 Collaborations avec l’environnement socio-économique . . . . .	8
4.6 Publications et production scientifique . . . . .	8
<b>1 Introduction</b>	<b>17</b>
1.1 Positionnement de ce travail . . . . .	17
1.1.1 Géométrie discrète . . . . .	17
1.1.2 Tomographie discrète . . . . .	17
1.1.3 Distances discrètes . . . . .	18
1.2 Contributions personnelles . . . . .	18
1.2.1 Éléments structurants à deux pixels . . . . .	18
1.2.2 Transformée Mojette . . . . .	19
1.2.3 FRT . . . . .	24
1.2.4 Distances discrètes . . . . .	25
1.2.5 Séjour en Australie . . . . .	25
1.3 Choix de contenu et présentation . . . . .	28
<b>2 Distances à poids locaux variables</b>	<b>31</b>
2.1 Introduction . . . . .	31
2.2 Préliminaires . . . . .	32
2.2.1 Distances discrètes . . . . .	32
2.2.2 Séquences non décroissantes d’entiers . . . . .	36
2.3 Distances à poids variables . . . . .	38
2.3.1 Variabilité des coûts de déplacements dans les distances . . . . .	40
2.3.2 Définition des distances à poids variables . . . . .	41
2.3.3 Expression de $d$ . . . . .	44

2.4	Propriétés des disques . . . . .	45
2.4.1	Construction itérative des disques . . . . .	45
2.5	Conclusion . . . . .	50
<b>3</b>	<b>Transformation en distance</b>	<b>51</b>
3.1	Introduction . . . . .	51
3.2	Préliminaires . . . . .	51
3.2.1	Transformation en distance par propagation . . . . .	52
3.2.2	Morphologie mathématique ensembliste . . . . .	54
3.2.3	Convexité et polytopes . . . . .	56
3.3	Transformation en distance à latence minimale . . . . .	59
3.3.1	Transformation en distance généralisée . . . . .	59
3.3.2	Distances à séquence de voisinages . . . . .	60
3.3.3	Recentrage de la transformation en distance asymétrique . . . . .	62
3.3.4	Disques intermédiaires . . . . .	63
3.4	Décomposition d'éléments structurants convexes . . . . .	67
3.4.1	Algorithme . . . . .	67
3.4.2	Validité de l'algorithme . . . . .	73
3.5	Conclusion . . . . .	75
<b>4</b>	<b>Axe médian</b>	<b>77</b>
4.1	Introduction . . . . .	77
4.2	Préliminaires . . . . .	78
4.2.1	Axe Médian . . . . .	78
4.2.2	Axe Médian des Distances de Chanfrein . . . . .	78
4.3	Normes de chanfrein et $\mathcal{H}$ -polytopes . . . . .	80
4.3.1	Géométrie des Boules de Chanfrein . . . . .	80
4.3.2	Condition de norme . . . . .	82
4.3.3	$\mathcal{H}$ -représentation des boules de normes de chanfrein . . . . .	84
4.4	Triangulation des boules de chanfrein 3D . . . . .	86
4.4.1	Sommets de l'enveloppe convexe dans le cône générateur . . . . .	86
4.4.2	Enveloppe convexe et triangulation de la boule rationnelle . . . . .	86
4.4.3	Triangulation unimodulaire . . . . .	87
4.4.4	Test d'homogénéité . . . . .	89
4.5	Voisinage de test et tables d'inclusion pour les normes de chanfrein . . . . .	89
4.5.1	$\widehat{\mathcal{H}}$ -représentation des boules de chanfrein . . . . .	89
4.5.2	Valeurs des tables d'inclusion . . . . .	89
4.5.3	Voisinage de test . . . . .	89
4.6	Résultats . . . . .	90
4.7	Conclusion . . . . .	91
<b>5</b>	<b>Granulométrie</b>	<b>93</b>
5.1	Introduction . . . . .	93
5.2	Préliminaires . . . . .	93
5.2.1	Granulométrie . . . . .	93
5.2.2	Calcul du spectre granulométrique . . . . .	95
5.2.3	Théorie de BRUNN-MINKOWSKI . . . . .	98
5.3	Estimation du spectre granulométrique à partir de la transformation en distance	102
5.3.1	Modèle continu . . . . .	102



5.3.2	Extraction simultanée des mesures pour tous les rayons . . . . .	104
5.3.3	Extrapolation des mesures à un dilaté . . . . .	106
5.4	Résultats et discussion . . . . .	110
5.5	Conclusion . . . . .	111
<b>Perspectives de recherche</b>		<b>115</b>
1	Transformation Mojette et FRT . . . . .	115
2	Distances discrètes et morphologie mathématique . . . . .	116
3	Thèmes connexes . . . . .	116
<b>Sélection d'articles</b>		<b>119</b>
	Recovering Missing Slices of the Discrete Fourier Transform using Ghosts . . . . .	121
	Distance transform computation for digital distance functions . . . . .	133
	Fast recursive grayscale morphology operators : from the algorithm to the pipeline architecture . . . . .	147
	Medial axis lookup table and test neighborhood computation for 3D chamfer norms .	157
	Multimedia Forward Error Correcting codes for wireless LAN . . . . .	167
<b>Index</b>		<b>183</b>
<b>Bibliographie</b>		<b>185</b>
<b>Table des matières</b>		<b>199</b>