On the Validity of the Two Raster Sequences Distance Transform Algorithm

Édouard Thiel

Aix Marseille Univ, Université de Toulon, CNRS, LIS, Marseille, France
Edouard.Thiel@univ-amu.fr
https://pageperso.lis-lab.fr/~edouard.thiel/

Abstract. This paper examines the validity of the two raster sequences distance transform algorithm, originally given by Rosenfeld and Pfaltz for the distance d_4, then extended to the weighted distances by Montanari and Borgefors. We show that the convergence in two passes does not hold for all chamfer masks, and we prove that the chamfer norm condition is a sufficient condition of validity for the algorithm.

Keywords: discrete geometry · distance transforms · weighted distances · chamfer norms.

1 Introduction

Given a binary image A composed of shape points and background points, a Distance Transform (DT) of A is a copy where each shape point is labelled to its distance from the nearest background point. Both the computation and the result properties depend on the considered distance function. The computation of a DT is generally a global operation, which can be quite expensive; however for some distance functions there are very efficient algorithms based on local operations, using sequential or parallel approaches.

DTs have been extensively studied and have played an important role in Discrete Geometry and Image Analysis since the late 1960s. In the founding paper [1], Rosenfeld and Pfaltz introduced the notion of DT, and presented a two raster sequences DT algorithm in 2D for the direct neighbourhood distance d_4. They also proved that for any given local transformation on an image, the sequential and parallel approaches are mathematically equivalent. Following that, the notion of weighted (or chamfer) distances has emerged in [2][3][4] together with a rather straightforward extension of the DT algorithm.

We recall some definitions and hypotheses from [6]. A *weighting* $(\vec{v}, w)$ is a displacement $\vec{v} \neq \vec{0}$ associated to a weight $w > 0$. A *chamfer mask* $\mathcal{M}$ is a non-empty set of weightings, such that the set of displacements contains at least a basis of the image points (reachability), and such that $\forall (\vec{v}, w) \in \mathcal{M}$, $(-\vec{v}, w) \in \mathcal{M}$ (central-symmetry). Two points P and Q are $\mathcal{M}$-adjacent if there exists $(\vec{v}, w) \in \mathcal{M}$ such that $\vec{PQ} = \vec{v}$. Two points P and Q are $\mathcal{M}$-connected if there exists a path of $\mathcal{M}$-adjacent points joining them, that is, a sequence of distinct points $P_0 = P, P_1, \ldots, P_k = Q$ with P_i a $\mathcal{M}$-neighbour of P_{i-1}, $1 \le i \le k$. The cost of the path is the sum of the weights of the displacements. The weighted distance $d_{\mathcal{M}}(P,Q)$ is the cost of a path having minimal cost:

$$d_{\mathcal{M}}(P,Q) = \min \Big\{ \sum \lambda_i w_i \ : \ \sum \lambda_i \vec{v}_i = \vec{PQ},\ (\vec{v}_i, w_i) \in \mathcal{M},\ \lambda_i \in \mathbb{Z}_+ \Big\} \ . \quad (1)$$

Equivalently, we can consider the weighted geometric graph (V, G), where the set of vertices V corresponds to the image points, and the set of edges G is defined as follows: each vertex $P \in V$ is connected to its $\mathcal{M}$-neighbours $P + \vec{v}$ by an edge having the weight w, $\forall (\vec{v}, w) \in \mathcal{M}$ s.t. $P + \vec{v} \in V$. The weighted distance $d_{\mathcal{M}}$ is then the intrinsic distance of this weighted graph, and always has the properties of a metric (positive definite, symmetric and triangular) since the graph is non oriented and the weights are strictly positive by hypothesis.

Let us go back to the origins. The first weighted distances d_4 and d_8 were presented in [1]; their mask correspond respectively to the 4- and 8-neighbourhood in $\mathbb{Z}^2$, each displacement having the weight 1; they coincide with the norms $\ell_1(\vec{x}) = |x_1| + \ldots + |x_n|$ and $\ell_\infty(\vec{x}) = \max(|x_1|, \ldots, |x_n|)$ in $\mathbb{Z}^n$.

In [2], Montanari introduced a family of weighted distances in $\mathbb{Z}^2$, where a mask $\mathcal{M}_k$ is the set of the displacements $\vec{v}(x,y)$ in the $(2k+1) \times (2k+1)$ neighbourhood (i.e. $-k \le x, y \le k$), such that (x,y) is visible (from the origin), i.e. $\gcd(x,y) = 1$. The weight of any displacement $\vec{v}(x,y)$ is its Euclidean length $\sqrt{x^2+y^2}$. The distance values obtained $d_{\mathcal{M}_k}$ are no longer integers, but can give a good approximation of the Euclidean distance d_E (depending on k). The two raster sequences DT algorithm is extended to the masks $\mathcal{M}_k$ and the convergence in two passes is shown.

The weighted distances using integer weights, or chamfer distances, have then been popularized for $\mathbb{Z}^2$ and $\mathbb{Z}^n$ by Borgefors in [3][4]. The merits of several masks and weights are discussed so as to approximate d_E in an efficient manner, and some conditions are given to choose the weights in order to establish direct distances formulas. The two raster sequences DT algorithm is presented in $\mathbb{Z}^n$. But the problem is that the convergence in two passes is not actually shown; and if we look closer, it cannot be deduced from the Rosenfeld and Pfaltz or Montanari proofs for all chamfer masks.

For these reasons, we propose to study the validity of the DT according to the mask, see some counter-examples, and give a sufficient condition of convergence.

The remainder of the paper is organized as follows: the section 2 first recalls the principle of the parallel and sequential DT algorithm for d_4 and d_8; we then examine in section 3 the original proof of [1], by completing it with a missing hypothesis; the section 4 presents an adaptation of the sequential DT algorithm for chamfer masks in $\mathbb{Z}^n$, in order to check the number of passes necessary for the convergence; in section 5, we study a counter-example which shows that the convergence does not always hold in two passes; after that in section 6 we show that the sequential DT algorithm always converges in two passes when using chamfer norms, and we conclude in section 7 .

2 Distance transformations in $\mathbb{Z}^2$ for d_4 and d_8

Let $A = (a_{i,j})$ be an input image, where $a_{i,j}$ denotes the value of the point at row i ($1 \leq i \leq m$) and column j ($1 \leq j \leq n$); the foreground points have value 1 and the background points 0. Given a chamfer mask $\mathcal{M}$, the goal is to compute the DT $D = (d_{i,j})$ where $d_{i,j}$ is the distance $d_{\mathcal{M}}$ to the set of 0's (supposed non-empty) of A. For any weighting $(\vec{v}, w) \in \mathcal{M}$ we denote by $(v_{\mathrm{i}}, v_{\mathrm{j}})$ the coordinates of $\vec{v}$.

Here is the naive parallel algorithm to compute DT. At step 0, let B^0 be a copy of A, where the 1's are set to ∞, or a sufficient large value. We compute for each step $k > 0$ the image $B^k = (b^k_{i,j})$, where

$$b^k_{i,j} = \min \left\{ b^{k-1}_{i+v_{\mathrm{i}},j+v_{\mathrm{j}}} + w \;:\; (\vec{v}, w) \in \mathcal{M},\; \begin{array}{c} 1 \leq i + v_{\mathrm{i}} \leq m, \\ 1 \leq j + v_{\mathrm{j}} \leq n \end{array} \right\}. \qquad (2)$$

The process is repeated until no point value changes. The number of iterations is bounded by the maximal number of displacements in a minimal $\mathcal{M}$-path, and can be quite large.

The same method can be processed in an iterative manner on a single image B. The order in which we compute the $b_{i,j}$ is arbitrary, and the convergence rate can be greatly increased by a clever choice of the order. The sequential DT algorithm of Rosenfeld and Pfaltz takes advantage of this idea, an converges in only two raster sequences on the image. Here is their original algorithm, presented in [1] for the distance d_4.

The forward scan processes the image row by row in the raster sequence $a_{1,1}, \ldots, a_{1,n},\; a_{2,1}, \ldots, a_{2,n},\; \ldots,\; a_{m,1}, \ldots, a_{m,n}$; the backward scan processes the points in the reverse order. During the forward scan the function f_1 is applied on A to obtain the image B, then during the backward scan the function f_2 is applied on B to get the image C. These functions are defined by:

$$\begin{array}{lll} f_1 : \; b_{i,j} & = 0 & \text{if } a_{i,j} = 0\,, \\ & = \min\left(b_{i-1,j} + 1\,, b_{i,j-1} + 1\right) & \text{if } a_{i,j} = 1 \text{ and } (i,j) \neq (1,1)\,, \\ & = \mu & \text{if } a_{i,j} = 1 \text{ and } (i,j) = (1,1)\,; \\ f_2 : \; c_{i,j} & = \min\left(b_{i,j},\, c_{i+1,j} + 1\,, c_{i,j+1} + 1\right). & \end{array}$$

The value μ is chosen to be an unattainable distance value in the image, e.g. $m + n$ (in the paper) or $+\infty$, and is set as an initialization for the top left point $(1,1)$. The min's are only evaluated on the neighbours inside the image; an alternative option is to consider the value μ for neighbours who are outside the image.

The algorithm can be easily adapted to d_8 by adding the indirect neighbours $(i-1, j-1)$ and $(i-1, j+1)$ in the min for f_1, and $(i+1, j-1)$, $(i+1, j+1)$ in the min for f_2.

Figure 1 shows an example with d_4 and Figure 2 with d_8. For simplicity, we have considered in the min's that $\mu + x = \mu$, $\forall x \geq 0$.

$$\begin{array}{|c|c|c|c|c|}\hline 1&1&1&1&1\\\hline 1&1&0&1&1\\\hline 1&1&1&1&1\\\hline\end{array} \xrightarrow{f_1} \begin{array}{|c|c|c|c|c|}\hline \mu&\mu&\mu&\mu&\mu\\\hline \mu&\mu&0&1&2\\\hline \mu&\mu&1&2&3\\\hline\end{array} \xrightarrow{f_2} \begin{array}{|c|c|c|c|c|}\hline 3&2&1&2&3\\\hline 2&1&0&1&2\\\hline 3&2&1&2&3\\\hline\end{array}$$

Fig. 1. Two raster sequences DT algorithm for d_4 on a 3×5 image.

$$\begin{array}{|c|c|c|c|c|}\hline 1&1&1&1&1\\\hline 1&1&0&1&1\\\hline 1&1&1&1&1\\\hline\end{array} \xrightarrow{f_1} \begin{array}{|c|c|c|c|c|}\hline \mu&\mu&\mu&\mu&\mu\\\hline \mu&\mu&0&1&2\\\hline \mu&1&1&1&2\\\hline\end{array} \xrightarrow{f_2} \begin{array}{|c|c|c|c|c|}\hline 2&1&1&1&2\\\hline 2&1&0&1&2\\\hline 2&1&1&1&2\\\hline\end{array}$$

Fig. 2. Two raster sequences DT algorithm for d_8 on a 3×5 image.

3 Original proof for the two raster sequences DT

The original proof in [1, §4.2] is rather compact; we will develop it and show that there was a missing hypothesis. The proof is constructed by induction for d_4 in $\mathbb{Z}^2$; the goal is to show that after applying f_1 and f_2, the obtained image C satisfies $C = D$ (using the notations of section 2).

On the base case it is noted that if $a_{i,j} = 1$ and a direct neighbour inside the image is 0, evidently $c_{i,j} = 1$, and conversely.

The original induction hypothesis is: suppose for a given $k > 1$ that

$$c_{i,j} = d_{i,j} \quad \forall i, j \ \text{ s.t. } \ d_{i,j} < k\,. \tag{3}$$

Hence $\forall i, j$ we have

$$d_{i,j} < k \quad \Rightarrow \quad c_{i,j} = d_{i,j}\ ; \tag{4}$$

but this does not exclude the existence of cases such as

$$d_{i,j} \geq k \quad \text{and} \quad c_{i,j} < k\,. \tag{5}$$

In fact, for the rest of the proof, we will have to exclude these cases in two places. The (extended) induction hypothesis has thus to be: suppose for a given $k > 1$ that

$$c_{i,j} = d_{i,j} \quad \forall i, j \ \text{ s.t. } \ d_{i,j} < k \ \text{ or } \ c_{i,j} < k\,. \tag{6}$$

We therefore further assumed that

$$c_{i,j} < k \quad \Rightarrow \quad c_{i,j} = d_{i,j}\,. \tag{7}$$

Remark. By (4) we have $d_{i,j} < k \ \Rightarrow \ c_{i,j} < k$, thus

$$c_{i,j} \geq k \quad \Rightarrow \quad d_{i,j} \geq k\,; \tag{8}$$

moreover, by (7) we have $c_{i,j} < k \ \Rightarrow\ d_{i,j} < k$, so $d_{i,j} \geq k \ \Rightarrow\ c_{i,j} \geq k$; hence

$$c_{i,j} \geq k \quad \Leftrightarrow \quad d_{i,j} \geq k \,. \tag{9}$$

We continue the induction by studying the case where $c_{i,j} = k$. By (8) we have $d_{i,j} \geq k$. If $d_{i,j} = k$ then $c_{i,j} = d_{i,j}$ and the proof is done. Let us suppose that $c_{i,j} = k$ and $d_{i,j} > k$. By definition of d_4, since $d_{i,j} > k$, the four direct neighbours are $\geq k$:

$$\begin{array}{c|c|c} & d_{i-1,j} \geq k & \\ \hline d_{i,j-1} \geq k & d_{i,j} > k & d_{i,j+1} \geq k \\ \hline & d_{i+1,j} \geq k & \end{array} \,. \tag{10}$$

Thanks to the extended hypothesis, we have by (9)

$$\begin{array}{lcl} d_{i,j+1} \geq k & \Rightarrow & c_{i,j+1} \geq k \,, \\ d_{i+1,j} \geq k & \Rightarrow & c_{i+1,j} \geq k \,, \end{array} \tag{11}$$

hence during the computation of $c_{i,j}$ by f_2 in backward sequence we have

$$c_{i,j} = \min \begin{cases} b_{i,j} \\ c_{i,j+1} + 1 \quad (\geq k+1) \\ c_{i+1,j} + 1 \quad (\geq k+1) \end{cases} , \tag{12}$$

thus $c_{i,j} = k \ \Rightarrow\ b_{i,j} = k$. However, when calculating $b_{i,j}$ by f_1 in forward sequence we have applied

$$b_{i,j} = \min \begin{cases} b_{i-1,j} + 1 \\ b_{i,j-1} + 1 \end{cases} , \tag{13}$$

thus $b_{i,j} = k \ \Rightarrow\ b_{i-1,j} = k-1$ or $b_{i,j-1} = k-1$. Suppose that the former holds, that is $b_{i-1,j} = k-1$. During the calculation of $c_{i-1,j}$ by f_2 we have

$$c_{i-1,j} = \min \begin{cases} b_{i-1,j} \qquad\qquad (= k-1) \\ c_{i-1,j+1} + 1 \\ c_{i,j} + 1 \end{cases} \tag{14}$$

therefore $c_{i-1,j} \leq k-1$; but $d_{i-1,j} \geq k$ by (10) so $d_{i-1,j} \neq c_{i-1,j}$, in contradiction with the extended hypothesis since $c_{i-1,j} < k$. □

This proof can be easily extended for d_8 by adding the four indirect neighbours in the min's. More generally, the algorithm and the proof can be extended in $\mathbb{Z}^n$ for the distances d_1 and d_∞ induced by the ℓ_1 and ℓ_∞ norms.

It should be noted that the algorithm can also be adapted to chamfer masks in $\mathbb{Z}^n$ (see [4]), but we will show further with a counter-example that the convergence in two scans is not always guaranteed for any chamfer mask. At the proof level, we can see that this proof cannot be extended either, because the inequations are performed on neighbours of (i,j) only, and they use the fact that the distance values are consecutive integers.

4 Sequential DTs for chamfer masks in $\mathbb{Z}^n$

We present an adaptation of the sequential DT in $\mathbb{Z}^n$ which is a bit hardened to handle counter-examples masks.

The masks need to be split in two parts for the forward and backward scans. Using coordinates $p = (x_1, \dots, x_n) \in \mathbb{Z}^n$, let us consider the forward raster sequence in ascending order by nested loops `for` x_n, `for` x_{n-1}, …, `for` x_1 .

The half-space visited by the loops after the origin in the raster sequence is $\mathcal{H}^n = \cup_{1 \le k \le n} \{ p : x_n = 0, \dots, x_{k+1} = 0, x_k > 0 \}$. For instance, the half-space $\mathcal{H}^3$ is $\{ p : x_3 = 0, x_2 = 0, x_1 > 0 \} \cup \{ p : x_3 = 0, x_2 > 0 \} \cup \{ p : x_3 > 0 \}$. Given a chamfer mask $\mathcal{M} = \{ (\vec{v}, w) : \vec{v} \in \mathbb{Z}^n \}$, we define the half-mask $\mathcal{M}^h = \{ (\vec{v}, w) \in \mathcal{M} : \vec{v} \in \mathcal{H}^n \}$. During the sequential DT, the forward raster sequence will then use the half-mask $\mathcal{M} \setminus \mathcal{M}^h$, whereas the backward one will use $\mathcal{M}^h$.

The computation of one sequential DT scan is presented in Figure 3 , for convenience in Python language in $\mathbb{Z}^2$. The source code and examples are online in [9]. To extend the function in higher dimension it is sufficient to add coordinates and loops for the additional dimensions.

```
def compute_one_DT_scan (img, half_mask, scan_num) :
    forward = scan_num % 2 == 1
    if forward :
        i_start = 0 ; i_end = img.m                    # 0 to m-1
        j_start = 0 ; j_end = img.n ; step = 1         # 0 to n-1
    else :
        i_start = img.m-1 ; i_end = -1                # m-1 to 0
        j_start = img.n-1 ; j_end = -1 ; step = -1     # n-1 to 0
    changed = False
    for i in range (i_start, i_end, step) :
        for j in range (j_start, j_end, step) :
            if img.mat[i][j] == 0 : continue
            min_w = -1 if scan_num == 1 else img.mat[i][j]
            for p_i, p_j, p_w in half_mask.weightings :
                q_i = i - p_i*step ; q_j = j - p_j*step
                if not img.is_inside (q_i, q_j) : continue
                if img.mat[q_i][q_j] == -1 : continue
                q_w = img.mat[q_i][q_j] + p_w
                if min_w == -1 or q_w < min_w : min_w = q_w
            if img.mat[i][j] != min_w : changed = True
            img.mat[i][j] = min_w                      # can be -1
    return changed
```

Fig. 3. Computation of one sequential DT scan in $\mathbb{Z}^2$ with $\mu = -1$.

The input and output image is `img`. The coordinates are $0 \le$ `i` $<$ `img.m` for x_2 (or y) and $0 \le$ `j` $<$ `img.n` for x_1 (or x); the point values are accessed by `img.mat[i][j]`. The method `img.is_inside(i,j)` returns `True` if the coordinates are inside the image. The parameter `half_mask` stores the $\mathcal{M}^h$ weightings as a list of tuples. The direction of the scan (forward or backward) is deduced from the scan number `scan_num` line 2. The loop `step` value is also used line 15 to compute the displacements of the half mask for the current scan direction.

The function is written with the special value $\mu = -1$. It indicates, as a forbidden distance value, the non currently propagated distance values in the image, and needs a test to handle the min's. Using signed pixel values, we find it more handy than choosing an arbitrary large integer to simulate $+\infty$.

The computation of the DT in two raster sequences is done by calling twice the function `compute_one_DT_scan` with the scan number, see the function `compute_sequential_DT_in_two_scans` in Figure 4 .

```
def compute_sequential_DT_in_two_scans (img, half_mask) :
    compute_one_DT_scan (img, half_mask, 1)
    compute_one_DT_scan (img, half_mask, 2)

def compute_sequential_DT_multi_scans (img, half_mask) :
    scan_num = 1
    while True :
        if compute_one_DT_scan (img, half_mask, scan_num) :
            scan_num += 1
        else : break
    return scan_num
```

Fig. 4. Sequential DT algorithms in $\mathbb{Z}^2$.

As for the parallel DT computation, the sequential DT can be performed scan by scan until no point value changes (all paths are propagated and convergence is reached). For this purpose, the function `compute_one_DT_scan` returns a boolean value `changed`, which is used to stop the loop in the function `compute_sequential_DT_multi_scans` in Figure 4 .

5 Counter-example for the two raster sequences DT

We present now a simple counter-example, which shows that the convergence of the DT in only two raster sequences does not hold for all chamfer masks.

One can imagine any kind of mask, see for instance [6, p. 42] for a gallery. In the literature, the most common category of studied masks are grid-symmetrical (8-symmetrical in $\mathbb{Z}^2$, 48- in $\mathbb{Z}^3$, $(2^n n!)$- in $\mathbb{Z}^n$). The weightings are chosen in the first octant (also called generator) $0 \leq x_n \leq \ldots \leq x_1$, then the grid symmetries are performed to populate the mask. For efficiency, the displacements are usually chosen among the visible points, because for a weighting $(\vec{v}, w)$, each period $O + \lambda\vec{v}$ is expected to get the distance value λw from O, if the mask has the good properties (see further), so adding $(\lambda\vec{v}, \lambda w)$ in the mask is useless.

In $\mathbb{Z}^2$, the first visible points in the first octant are denoted by $\mathbf{a} = (0, 1)$ (still using coordinates in the order (x_2, x_1)), $\mathbf{b} = (1, 1)$, $\mathbf{c} = (1, 2)$, $\mathbf{d} = (1, 3)$,

$\mathbf{e} = (2, 3)$, etc. A grid-symmetrical mask constituted by a set of weightings $(\mathbf{v}, w)$ where $\mathbf{v}$ is a visible point is denoted by $\langle(\mathbf{v}, w), \ldots\rangle$. For instance, the mask for d_4 is denoted by $\langle(\mathbf{a}, 1)\rangle$, the mask for d_8 is $\langle(\mathbf{a}, 1), (\mathbf{b}, 1)\rangle$, the mask for the chamfer distance 5,7,11 [4] is $\langle(\mathbf{a}, 5), (\mathbf{b}, 7), (\mathbf{c}, 11)\rangle$, and so on.

To find counter-examples it is sufficient to choose some displacements, loop on several weights, and compute the DTs on images of several sizes, where all points have value 1, except one point which has value 0 in the centre of the image. For each trial we can compare the results for the parallel algorithm and those of `compute_sequential_DT_in_two_scans`, or run the function `compute_sequential_DT_multi_scans` and check if it returns a number of scans > 3. See the program `checkWDT.py` in [9].

We have found a very simple counter-example for any image size larger then 3×3: this is the mask $\langle(\mathbf{c}, 1)\rangle$, also known as the Knight distance [5]. The Figure 5 shows the full mask and the two half masks.

(a)

·	1	·	1	·
1	·	·	·	1
·	·	O	·	·
1	·	·	·	1
·	1	·	1	·

(b)

·	1	·	1	·
1	·	·	·	1
·	·	O		

(c)

		O	·	·
1	·	·	·	1
·	1	·	1	·

Fig. 5. Mask $\langle(\mathbf{c}, 1)\rangle$ around the origin O: (a) full, (b) forward, (c) backward mask.

The mask $\langle(\mathbf{c}, 1)\rangle$ is a chamfer mask because the basis vector $(0, 1)$ can be obtained using the symmetrical displacements of $\mathbf{c}$, by $(-1, -2) + (-1, 2) + (2, 1) = (0, 1)$, and the same by symmetry for $(1, 0)$.

The Figure 6 shows the parallel passes for a 3×4 image; 6 passes are necessary to reach the correct DT values. See the program `showWDT.py` in [9].

(a)

1	1	1	1
1	1	0	1
1	1	1	1

(b)

μ	μ	μ	μ
μ	μ	0	μ
μ	μ	μ	μ

(c)

1	μ	μ	μ
μ	μ	0	μ
1	μ	μ	μ

(d)

1	2	μ	μ
μ	μ	0	μ
1	2	μ	μ

(e)

1	2	3	μ
μ	μ	0	3
1	2	3	μ

(f)

1	2	3	4
4	μ	0	3
1	2	3	4

(g)

1	2	3	4
4	5	0	3
1	2	3	4

(h)

1	2	3	4
4	5	0	3
1	2	3	4

Fig. 6. Parallel DT for $\langle(\mathbf{c}, 1)\rangle$: (a) original image, (b) initialization, (c–h) passes 1–6.

On figure 7 we can see that the raster sequences DT algorithm also needs 6 passes: 5 to converge and the sixth to detect no changes and stop.

(a)

1	1	1	1
1	1	0	1
1	1	1	1

(b)

μ	μ	μ	μ
μ	μ	0	μ
1	μ	μ	μ

(c)

1	2	μ	μ
μ	μ	0	μ
1	μ	μ	μ

(d)

1	2	μ	μ
μ	μ	0	3
1	2	3	μ

(e)

1	2	3	4
4	μ	0	3
1	2	3	μ

(f)

1	2	3	4
4	5	0	3
1	2	3	4

(g)

1	2	3	4
4	5	0	3
1	2	3	4

Fig. 7. Sequential DT for $\langle(\mathbf{c}, 1)\rangle$: (a) original image, (b–g) passes 1–6, (b,d,f) forward passes, (c,e,g) backward passes.

The following remarks can be made for the mask $\langle(\mathbf{c}, 1)\rangle$ taken as a counter-example for the convergence in two passes:

- The two raster sequences DT presented as the Knight Transform (KT) in [5] therefore does not converge in the general case.
- The necessary sequential passes number depends on the image size, and may decrease a little bit when the size grows.
- The passes number does not depend on the chosen $\mathbf{c}$ weight.
- We can replace $\mathbf{c}$ by any visible point $(1, 2k)$, $k \geq 1$ and still get a chamfer mask, since $k(-1, -2k) + k(-1, 2k) + (2k, 1) = (-2k, 0) + (2k, 1) = (0, 1)$.

6 Validity holds for chamfer norms

A metric d in $\mathbb{Z}^n$ induces a discrete norm g defined by $g(q - p) = d(p, q)$ if g satisfies the property of homogeneity over $\mathbb{Z}$ [6, §2.2.3]:

$$\forall \vec{x} \in \mathbb{Z}^n,\ \forall \lambda \in \mathbb{Z},\ g(\lambda \vec{x}) = |\lambda|\, g(\vec{x})\,. \tag{15}$$

A chamfer norm is a discrete norm induced by a chamfer mask.

For instance, the masks $\langle(\mathbf{a}, 1)\rangle$ for d_4, $\langle(\mathbf{a}, 1), (\mathbf{b}, 1)\rangle$ for d_8, $\langle(\mathbf{a}, 3), (\mathbf{b}, 4)\rangle$ and $\langle(\mathbf{a}, 5), (\mathbf{b}, 7), (\mathbf{c}, 11)\rangle$ all induce chamfer norms, but $\langle(\mathbf{c}, 1)\rangle$ clearly not (no homogeneity: let $P = (0, 1)$, then $d(O, P) = 3$ and $d(O, 2.P) = 2 \neq 2.d(O, P)$).

The chamfer norms have remarkable properties: they allow to completely characterize the geometry of the distance balls, to give direct distance formulas, and to determine the structure of minimal paths. Given a chamfer mask $\mathcal{M}$, we call rational ball of $\mathcal{M}$ the set

$$\mathcal{B}^{\mathbb{Q}}_{\mathcal{M}} = \operatorname{conv}\left(\frac{\vec{v}}{w} \ :\ (\vec{v}, w) \in \mathcal{M} \right)\,. \tag{16}$$

The rational ball is a convex polyhedron, whose geometry is the same as the distance balls up to a scale factor.

The conditions for being a chamfer norm in $\mathbb{Z}^n$ are established in [6, §4.3.4] and [7, §4.3.2]: a chamfer mask $\mathcal{M}$ induces a discrete norm if and only if it exists a triangulation of $\mathcal{B}^{\mathbb{Q}}_{\mathcal{M}}$ in unimodular cones of apex O. Now suppose that $\mathcal{M}$ induces a discrete norm and let $\mathcal{C}$ be such a cone, then $\mathcal{C}$ is bounded by a subset of n weightings of $\mathcal{M}$, denoted by $\mathcal{M}|_{\mathcal{C}} = \{ (\vec{v}_i\,', w_i'),\ 1 \leq i \leq n \}$; moreover, for each point P in $\mathcal{C}$, there is a minimal path from O to P which is a linear combination $\lambda_1 \vec{v}_1\,' + \ldots + \lambda_n \vec{v}_n\,'$, $\lambda_i \in \mathbb{Z}_+$ of displacements from $\mathcal{M}|_{\mathcal{C}}$, and whose intermediate points are all included in $\mathcal{C}$.

Proposition 1. *Let $\mathcal{M}$ be a chamfer norm mask, then the two raster sequences DT algorithm provides the correct DT values for $d_{\mathcal{M}}$.*

Proof. Let P be a feature point currently evaluated during a raster sequence, and Q a closest background point. Consider the unimodular cone $\mathcal{C}$ of apex P which contains a minimal $\mathcal{M}$-path $\mathcal{P}$ from P to Q, and the set $\mathcal{M}|_{\mathcal{C}}$ of weightings which are bounding $\mathcal{C}$. Then $\mathcal{P}$ is a sequence of distinct points $P_0 = P$, P_1, …, $P_k = Q$ with P_i a $\mathcal{M}|_{\mathcal{C}}$-neighbour of P_{i-1}, $1 \leq i \leq k$.

The cone $\mathcal{C}$ is either (a) contained in the half-space $P - \mathcal{H}^n = \{ P - \overrightarrow{OX} : X \in \mathcal{H}^n \}$ (the points before P in the forward scan), see Figure 8 ; (b) in the half-space $P + \mathcal{H}^n$ (the points before P in the backward scan); or (c) intersects both half-spaces.

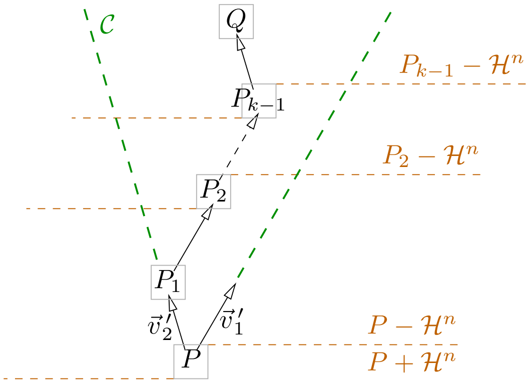

Fig. 8. Case (a) for the proof of proposition 1 , here in $\mathbb{Z}^2$.

In the case (a) each P_i is contained in the half-space $P_{i-1} - \mathcal{H}^n$, $1 \leq i \leq k$, so during the forward scan, each P_i is evaluated before P_{i-1}. As P_{k-1} is an $\mathcal{M}|_{\mathcal{C}}$-neighbour of $P_k = Q$, the min computation will give the correct associated weight value in the DT for P_{k-1}, an so on from P_{k-1} to P_0.

In the case (b), the same reasoning can be made using $P_{i-1}+\mathcal{H}^n$ during the backward scan.

In case (c), if $Q \in P-\mathcal{H}^n$, then a minimal path can be chosen such that all the path points are included in $\mathcal{C} \cap (P-\mathcal{H}^n)$, so we can revert to case (a); the same for $Q \in P+\mathcal{H}^n$ and case (b). □

7 Conclusion and future work

In this paper, we have improved the proof of [1] for d_1 and d_∞, and proposed a hardened raster sequence DT algorithm for the chamfer masks. We have shown with a counter-example that the convergence does not always hold in two passes for all chamfer masks, and we have proved in proposition 1 that the two raster sequences DT algorithm provides the correct distance values for any chamfer norm.

It can be pointed out that the norm condition is sufficient but non necessary. For instance, the algorithm holds for the following non-norm chamfer masks: $\langle(\mathbf{a},1),\ (\mathbf{b},1),\ (\mathbf{c},1)\rangle$, $\langle(\mathbf{a},1),\ (\mathbf{b},3),\ (\mathbf{c},2)\rangle$, $\langle(\mathbf{a},2),\ (\mathbf{b},3),\ (\mathbf{c},4)\rangle$, $\langle(\mathbf{a},1),\ (\mathbf{c},1)\rangle$, $\langle(\mathbf{a},2),\ (\mathbf{c},3)\rangle$.

In future works, it would be interesting to investigate if necessary conditions could be established on non-norm chamfer masks, to predict the number of passes for their convergence, and also to study the convergence for the reverse distance transform. This work on weighted distances might be extended on semi-regular grids, or other families of weighted geometric graphs. One could finally relate this work to ns-weighted distances, of which weighted distances are a special case [7][8].

References

1. Rosenfeld, A., Pfaltz, J.: Sequential operations in digital picture processing. Journal of ACM **13**(4), 471–494 (1966)
2. Montanari, U.: A method for obtaining skeletons using a quasi-euclidean distance. Journal of ACM **15**, 600–624 (1968)
3. Borgefors, G.: Distance transformations in arbitrary dimensions. Computer Vision, Graphics and Image Processing **27**, 321–345 (1984)
4. Borgefors, G.: Distance transformations in digital images. Computer Vision, Graphics and Image Processing **34**, 344–371 (1986)
5. Das, P.P., Chatterji, B.N.: Knight's distance in digital geometry. Pattern Recognition Letters **7**, 215–226 (1988)
6. Thiel, E.: Géométrie des distances de chanfrein. Habilitation à Diriger des Recherches, Université de la Méditerranée, Aix-Marseille 2 (Déc 2001), https://pageperso.lis-lab.fr/~edouard.thiel/hdr/
7. Normand, N.: Projections et distances discrètes. Habilitation à Diriger des Recherches, Université de Nantes (Nov 2012)
8. Strand, R, Normand, N.: Distance transform computation for digital distance functions. Theoretical Computer Science **448**, 80–93 (2012)
9. Annex with source code: https://pageperso.lis-lab.fr/~edouard.thiel/DGMM2022/