JOURNAL OF VISUAL COMMUNICATION AND IMAGE REPRESENTATION
Vol. 5, No. 1, March, pp. 107–115, 1994

SHORT COMMUNICATION

Well-Shaped, Stable, and Reversible Skeletons from the (3,4)-Distance Transform

GABRIELLA SANNITI DI BAJA

Istituto di Cibernetica, CNR, Via Toiano 6, I-80072, Arco Felice, Naples, Italy

Received August 17, 1992; accepted October 15, 1992

The skeleton of a digital pattern is extracted from the distance transform of the pattern, computed according to a quasi-Euclidean distance function. The pattern can be quite faithfully reconstructed by applying the reverse distance transformation to the skeleton, since this includes almost all the centers of the maximal discs of the pattern. The shape characterizing the discs is rounded, due the use of a quasi-Euclidean distance function, so that skeleton stability under pattern rotation is greatly favored. A pruning step, which makes it possible to simplify the structure of the skeleton without losing significant information, and a beautifying step, which is aimed at reducing the jaggedness possibly affecting some skeleton branches, are added to the process to improve the well-shapedness of the resulting skeleton. © 1994 Academic Press, Inc.

1. INTRODUCTION

Thinning is a transformation which, applied to a binary picture consisting of black and white pixels, changes from black to white a number of pixels, without altering picture topology, until the set of the black pixels B is transformed into a subset T, unit wide and centered within B.

Elongated patterns, characterized by nearly constant width, can be thinned by repeatedly applying to the contour of the pattern a topological peeling process, aimed at removing the black pixels not necessary to preserve picture topology. When thinning patterns with nonconstant thickness, one should also preserve from deletion the end points, i.e., the pixels necessary to map in T the tips of the pattern protrusions. In this case, thinning is preferably named skeletonization and the set T is referred to as the skeleton. End point detection is crucial to prevent excessive shortening, or even complete vanishing, of the skeleton branches expected to map pattern protrusions whose width w is significantly smaller than the maximal pattern width W. In fact, the length of the branches mapped in T in correspondence with these protrusions would be about $(W - w)/2$ pixels shorter than expected, if the end points were not detected and protected from removal.

End point detection should rely upon some geometric criterion whose validity is influenced by neither the way in which thinning is performed, nor the orientation of the pattern. Contour curvature can be used to identify the end points as the pixels perceptually dominating the contour arcs which delimit convex regions of the pattern and whose curvature is sufficiently high. Alternatively, if the pattern is interpreted as a union of maximal discs, the end points can be identified as the centers of the maximal discs, which are placed in correspondence with the tips of the pattern protrusions. This criterion is used in the following.

The geometrical structure of T may be not enough to characterize the pattern. An illustrative example is shown in Fig. 1. Indeed, to account for pattern width variations, the pixels of the skeleton should be labeled *according to their distance from the complement of the pattern.*

The distance transform of the pattern with respect to the complement is used in this paper to drive skeletonization. Since the label pertaining to any pixel p of the distance transform accounts for the radius of the largest disc that can be centered on p without exceeding the pattern, a comparison between the label of p and those of its neighbors allows one to establish whether the disc centered on p is maximal. Thus, the centers of the maximal discs can easily be identified and ascribed to the skeleton, which becomes reversible. The end points are directly identified since, among the centers, those pertaining to the maximal discs placed in correspondence with pattern protrusions are also found. Skeleton connectedness is achieved by growing paths in the direction of the steepest gradient on the distance transform. To favor skeleton stability under pattern rotation, the distance transform is computed according to a weighted distance function reasonably approximating the Euclidean distance function [1]. Finally, to improve skeleton well-shapedness and to favor the use of the skeleton for shape analysis, a pruning step and a beautifying step are added. Pruning is guided by distance information and allows one to simplify the

1047-3203/94 $6.00

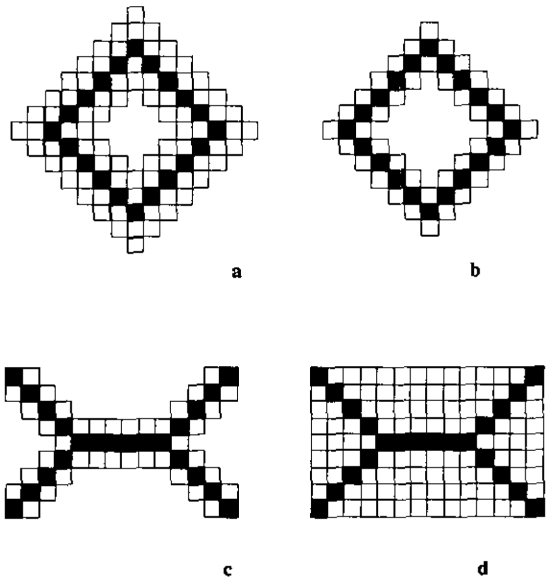


FIG. 1. Different patterns thinned down into equally structured sets (black squares).

n_2	n_3	n_4
n_1	p	n_5
n_8	n_7	n_6

FIG. 2. The eight neighbors of a pixel p.

skeleton structure without losing significant details on the shape and the size of the pattern.

2. PREMISES

Let $\overline{B} = \{1\}$ and $\overline{B} = \{0\}$ be the sets of the black pixels and of the white pixels in a binary picture, respectively. The 8-metric is selected for B, and the 4-metric for $\overline{B}$. We assume that B is an 8-connected set and that each pixel of B is sufficiently far apart from the frame of the picture so that we can safely apply 3×3 operations to any black pixel.

The eight neighbors of a pixel p are denoted by n_i ($i = 1, 8$) as shown in Fig. 2, and their set is referred to as the neighborhood $N(p)$. Neighbors n_i are also referred to as odd-neighbors and even-neighbors, depending on whether "i" is odd or even.

Any neighbor of p can be reached with a unit move from p. A path between two pixels p and q is a sequence of unit moves from p to q. The shortest path is the one requiring the minimum number of unit moves. The shortest path is not necessarily unique.

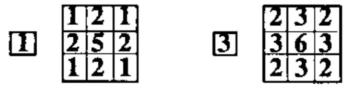
The discrete distance between p and q is the length of the shortest path from p to q, computed by using two weights (w_{odd} and w_{even}) to weight the horizontal/vertical and the diagonal unit moves, respectively. The weighted distance function is denoted by (w_{odd}, w_{even})-distance.

The (w_{odd}, w_{even})-distance transform of B with respect to $\overline{B}$ is a multivalued replica of B, where each pixel p is labeled with its (w_{odd}, w_{even})-distance from $\overline{B}$. In the following, the distance transform is briefly referred to as (w_{odd}, w_{even})-*DT*, and when no ambiguity arises, p indicates both the pixel and its associated label. We do not dwell on the (w_{odd}, w_{even})-*DT* computation, since a number of algorithms are available in the literature for this purpose [1–3].

On the (w_{odd}, w_{even})-*DT*, each pixel p can be interpreted as the center of a disc, D_p, which includes all the black pixels whose (w_{odd}, w_{even})-distance from p is less than p. A disc D_p not completely included in the disc D_q centered on any neighbor q of p is called a maximal disc. The center of a maximal disc is referred to as a maximal center and denoted by *mc*. The union of all the maximal discs has the same size and shape as B. It can be obtained by applying the (w_{odd}, w_{even})-reverse-distance transformation to the set of the *mc*'s [4].

Pattern recovery starting from the set of the maximal centers is guaranteed, whichever distance function is adopted to build the distance transform. However, if the chessboard ($w_{odd} = w_{even} = 1$) or the city-block ($w_{odd} = 1$, $w_{even} = \infty$) distance transform is considered, number, relative position, and label of the *mc*'s are likely to change drastically under pattern rotation. Stability under rotation is greatly favored if the shape of the disc is quasi-circular, i.e., if the adopted distance provides a better approximation to the Euclidean distance. In the following, $w_{odd} = 3$ and $w_{even} = 4$ are adopted. The corresponding disc is octagon-shaped, except for the case of small discs, whose shape is conditioned by the discrete nature of the digital plane.

Since the size of the disc increases with the label of its center, to establish whether the disc D_p is included in the discs D_q, centered on the neighbor q of p, one could compare p with q. Accordingly, D_p should be not completely included by D_q if $q < p + w_k$ (where w_k is either 3 or 4, depending on the relative position of p and q). If the same occurs in correspondence with all the neighbors of p, the pixel p should be the center of a maximal disc. Indeed, in [4] it has been shown that the previous criterion fails if labels 6 and 3 are not preliminary changed to 5 and 1, respectively. The reason this label updating is compulsory is summarized herein for clarity. The discs D_3 and D_6, associated with the pixels labeled 3 and 6 respectively, have the same size and shape as the discs D_1 and D_5, associated with 1 and 5. See Fig. 3. D_1 and D_5 are, then, equivalent to D_3 and D_6 as concerns any comparison involving shape and/or size among discs. Using the values 1 and 5 in place of 3 and 6 when searching for



a b

FIG. 3. Discs obtained by applying the (3,4)-reverse-distance transformation to 1 and 5 (a) and to 3 and 6 (b).

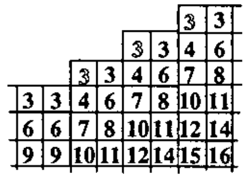


FIG. 4. The outlined 3's would be erroneously detected as centers of maximal discs, if label updating ($3 \to 1$, $6 \to 5$) were not taken into account.

the *mc*'s in the (3,4)-*DT* allows one to avoid detecting a number of false *mc*'s, whose associated discs are not maximal. See Fig. 4. During skeletonization, the false *mc*'s would become the origins of false branches, whose presence excessively burdens the structure of the resulting *T*. Label updating can be safely performed, since the values 1 and 5 are never found in the (3,4)-*DT*, where any label is a linear combination of the two adopted weights.[1]

A pixel p on the (3,4)-*DT* is the center of a maximal disc if, after the labels 6 and 3 have been changed into 5 and 1 respectively, it is

$$n_i < p + 3 \quad \text{for every odd-neighbor } n_i$$

and

$$n_i < p + 4 \quad \text{for every even-neighbor } n_i.$$

On the (3,4)-*DT*, an increasing path from a pixel p to a pixel q, $q > p$, is a path along which the labels increase monotonically. The path is termed a regularly increasing path if, for any two successive pixels in the path, say r_1 and r_2 ($r_2 > r_1$), it is

$$r_1 + w_m \leq r_2,$$

where $w_m = 3$ or $w_m = 4$, depending on whether r_2 is an odd-neighbor or an even-neighbor of r_1, and labels 6 and 3 have been changed into 5 and 1, respectively.

A maximal center cannot be an intermediate pixel of a regularly increasing path.

The layers of the (3,4)-*DT* are subsets of pixels, whose labels differ from each other by less than 3 [5]. A pixel p belongs to the kth layer if

$$(k - 1) \times 3 < p \leq k \times 3.$$

Basically, the layers can be understood as the successive contours which would characterize a pattern if a peeling process is iteratively accomplished on it. The layers are curves, either 8-connected or 4-connected.

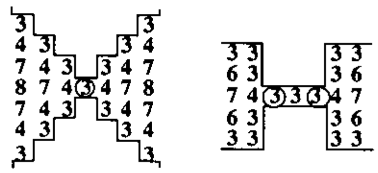
A pixel p in the (3,4)-*DT* is termed a saddle pixel, if in $N(p)$ there exist two 4-connected components of pixels with smaller labels, and one or two 8-connected components of pixels with larger labels. Having one or two components of pixels with label higher than p depends on whether p is centered in a hourglass configuration (Fig. 5a) or delimits an elongated saddle configuration (Fig. 5b). In the latter case, p is neighbor of an equally labeled pixel, which is a maximal center. In the presence of pattern subsets having even thickness, the definition of saddle pixel can be similarly given, using a 4×4 neighborhood.

The 8-connected components and the 4-connected components of black pixels in the neighborhood of p are counted by the connectivity number $C_8(p)$ and by the crossing number $X_4(p)$, respectively. $C_8(p)$ and $X_4(p)$ are computed as follows, where $n_9 = n_1$, and $\bar{n}_k = 1 - n_k$ [6, 7]:

$$C_8(p) = \sum_{k=1}^{4} (\bar{n}_{2k-1} - \bar{n}_{2k-1} \times \bar{n}_{2k} \times \bar{n}_{2k+1})$$

$$X_4(p) = \frac{1}{2} \times \sum_{k=1}^{8} |n_{k+1} - n_k|.$$

When $C_8(p)$ and $X_4(p)$ are computed for pixels belonging to the (3,4)-*DT*, a suitable binary version of $N(p)$ is taken into account.



a b

FIG. 5. The encircled pixels are centered in 3×3 saddle configurations.

[1] Updating would regard a remarkably larger number of labels, if w_{odd} and w_{even} were assigned values different from 3 and 4, to better approximate the Euclidean distance. For instance, if $w_{odd} = 8$ and $w_{even} = 11$ were used, all the following changes would be necessary: $8 \to 1$, $11 \to 9$, $16 \to 12$, $19 \to 17$, $22 \to 20$, $24 \to 23$, $27 \to 25$, $30 \to 28$, $32 \to 31$, $35 \to 34$, $38 \to 36$, $40 \to 39$, $43 \to 42$, $46 \to 45$, $48 \to 47$, $51 \to 50$, $54 \to 53$, $59 \to 58$, $62 \to 61$, and $70 \to 69$.

3. THE SKELETON

The skeleton T is an 8-connected set, which is a subset of an 8-connected pattern B; has the same connectivity order as B, and each of its loops surrounds a hole of B; is centered within B; is unit wide; has pixels labeled with their (3,4)-distance from $\bar{B}$; and includes all the centers of maximal discs of the (3,4)-DT, except for those whose removal is indispensible to allow T to be a unit wide set.

On the skeleton, the pixels can be classified into end points, normal points, and branch points, by taking into account the number of components of neighbors not belonging to the skeleton, and/or the number of neighbors belonging to the skeleton.

We define as end points the pixels having unique (4-connected) components of neighbors not in the skeleton. This definition is only apparently equivalent to the commonly used definition, according to which the end points are the pixels having just one neighbor in the skeleton. In fact, only the first definition allows one to identify the starting points of some short, generally undesired, skeleton branches (see Fig. 6) and, accordingly, is more adequate to drive a pruning step. Note that the previous definitions can both be used to identify the end points after the skeleton has been obtained, but are generally not adequate to identify them within skeletonization. In fact, they both rely upon an event whose occurrence strongly depends on the way skeletonization is performed. On the contrary, the detection criterion should account for a perceptual meaning: the end points are extremes of peripheral skeleton branches, and should be placed in correspondence with tips of protrusions of B. This is the case, when end point detection is accomplished through center of maximal disc detection.

A branch point is any pixel in the skeleton having more than two neighbors in the skeleton and a number of components of neighbors not in the skeleton different from one (still, refer to Fig. 6). The branch points identify crossings of skeleton branches, and should be located in correspondence to superposition of regions of B. Since branch points are likely to be extremes of peripheral skeleton branches, their identification is important to prevent skeleton disconnection during pruning. A normal point is a pixel of the skeleton which is neither an end point nor a branch point.

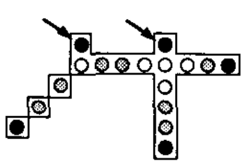


FIG. 6. Black, white, and gray dots are end points, branch points, and normal points, respectively. Pixels pointed to by the arrows would not be selected as end points, if the definition based on the number of neighbors in the skeleton were followed.

4. SKELETONIZING THE (3,4)-DISTANCE TRANSFORM

Skeletonization has been often performed by repeatedly applying topology preserving removal operations to the black pixels, which at any given iteration represent the pattern contour (e.g., [8]); removable pixels are changed to white provided that they are not identified as end points. Selection of the contour pixels as the only candidates for removal is done to favor an isotropic compression of the pattern towards its innermost part. The end point detection criterion is the weak point of many algorithms, and often compromises the isotropy of the transformation.

Our skeletonization method does not require the iterated application of topology preserving removal operations, and does not need checking a condition specifically tailored to end point detection, since end points are automatically identified when the maximal centers are found. No black pixel is actually changed to white on the (3,4)-DT (i.e., removed from the pattern), but the pixels to be ascribed to the skeleton (skeletal pixels) are directly identified and marked. The maximal centers and the saddle pixels are parallelwise detected. Since they do not generally group into a connected set, further skeletal pixels are found by growing connecting paths. Reduction to unit width is obtained by erasing the marker from suitable skeletal pixels. Finally, pruning and beautifying steps contribute to improve the shape of the resulting skeleton, which is a set of marked pixels on the (3,4)-DT.

Although the general scheme of a distance transform based algorithm is almost the same whichever distance function is used [9–11], different tasks have to be faced and solved, which are peculiar to the adopted distance transform. The algorithm proposed in this paper consists of a number of steps, some of which are individually discussed in the following. Steps accomplished by ordinary algorithms already available in the literature are not described here. This is the case for the cleaning step, whose inclusion in a skeletonization process as a preliminary step is always advisable, at least as concerns filling of noisy holes in the set of the black pixels. Noisy holes would otherwise be mapped into skeleton loops. While removal of noisy branches can be postponed to the pruning step, the presence in the skeleton of the loops originated in correspondence to noisy holes of B would irreparably modify the skeleton structure as concerns the number of arcs and curves expected to constitute the skeleton.

4.1. Extracting the Skeletal Pixels

The skeletal pixels found on the distance transform can be classified as "parallelwise detectable" and "sequentially detectable" skeletal pixels. Parallelwise detectable skeletal pixels can be directly identified on the distance transform, due to the structure of their neighborhoods.

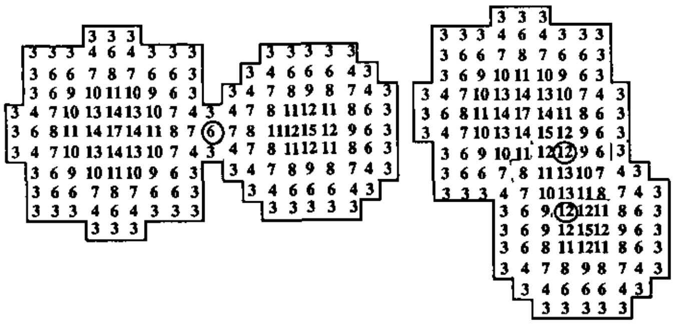


FIG. 7. The encircled pixels are saddle pixels as well as maximal centers.

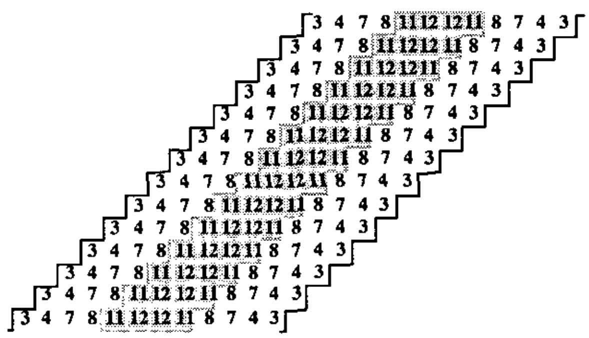


FIG. 8. All pixels in the gray area are maximal centers.

This is the case of the centers of the maximal discs and of the saddle pixels. Sequentially detectable skeletal pixels can be found only after some of their neighbors with smaller labels have been identified and marked as skeletal pixels. In the absence of marked pixels, the neighborhood configurations of the sequentially detectable skeletal pixels are indistinguishable from those characterizing the pixels which certainly are not skeletal pixels. Sequentially detectable skeletal pixels are necessary to link to each other the components of parallelwise detectable skeletal pixels.

4.1.1. *Parallelwise Detectable Skeletal Pixels*

In the (3,4)-*DT* case, the set of the parallelwise detectable skeletal pixels is almost completely exhausted by the *mc*'s, since most of the saddle pixels are maximal centers. In fact, the neighbors of a saddle pixel p, which are included in the component of pixels with label larger than p, are likely to be labeled less than $p + w_k$ ($w_k = 3, 4$). See Fig. 7. The set of the *mc*'s is rid of 8-internal pixels. In fact, any *mc* labeled p necessarily has at least one neighbor labeled $p - w_k$ ($w_k = 3, 4$), from which it derives label information. This neighbor is not a maximal center, since through it distance information propagates to p. Thick components of *mc*'s are likely to be found in correspondence to diagonally oriented regions of the input pattern, if the most internal layer of the (3,4)-*DT* in that region is a 4-connected curve. See Fig. 8. If a selection of the *mc*'s is done, in such a way that only the most internal ones in any component of maximal centers are marked, the successive reduction to unit width of the set of the skeletal pixels is facilitated. To this purpose, any maximal center p having a triple of consecutive neighbors (odd-neighbor/even-neighbor/odd-neighbor) which are all labeled more than p is not marked. This can be safely done since, if the nonmarked *mc* is necessary for connection purposes, it will be identified as a sequentially detectable skeletal pixel. The recoverability property of the set of the maximal centers does not diminish, since the disc associated with a nonmarked maximal center p is completely recovered by the union of the discs associated with its odd-neighbors, which are maximal centers, belong to the same layer as p, and are labeled more than p.

Any pixel p, which is not an *mc*, is marked as a saddle pixel if any of the following conditions holds:

(i) In $N(p)$ there is more than one 8-connected component of pixels labeled more than p (Fig. 9a);

(ii) In $N(p)$ there is more than one 4-connected component of pixels labeled less than p (Fig. 9b).

The above components are counted by the connectivity number $C_8(p)$ and the crossing number $X_4(p)$ respectively, computed by using a suitable binary version of $N(p)$.

For any pixel p labeled 3, the following additional condition is also checked, to guarantee detection of all the saddle pixels:

(iii) In $N(p)$ there exists a triple of consecutive neighbors of p (odd-neighbor/even-neighbor/odd-neighbor) which are all labeled 3 (Fig. 9c)

Marking the parallelwise detectable skeletal pixels is done during one raster scan inspection. Every pixel found to be a maximal center according to the criterion of Section 2, is marked if its presence in the set of the *mc*'s does not cause excessive thickening. Pixels which are not *mc*'s are subjected to the previous checks (i)–(iii) to verify whether they have to be marked as saddle pixels.

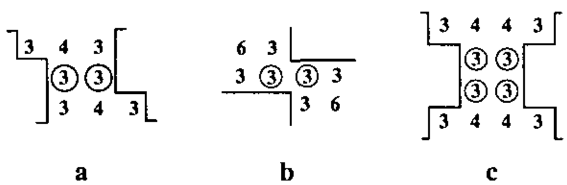


FIG. 9. Encircled labels satisfy one among three conditions: (a) Existence of two 8-connected components of pixels with higher label; (b) Existence of two 4-connected components of pixels with smaller label (white pixels); (c) Existence of an L-shaped triple of consecutive neighbors labeled 3.

4.1.2. *Sequentially Detectable Skeletal Pixels*

The sequentially detectable skeletal pixels are identified by growing increasing paths along the direction of the steepest gradient in the (3,4)-*DT*, starting from any already marked pixel. For this purpose, whenever a pixel is marked as a parallelwise detectable skeletal pixel, the raster scan inspection is temporarily interrupted to check whether a connecting path can be grown. The raster inspection is resumed after the path has been completely built.

The connecting path is grown by marking as skeletal pixels the pixels for which the maximal value of the gradient is positive. The first pixel in the path is the (unmarked) neighbor n_k of a marked pixel p, such that $n_k > p$ and the gradient of q with respect to p is maximum. The gradient is computed as

$$grad(n_k) = \frac{1}{w_m} \times (n_k - p)$$

where it is $w_m = 3$, for k odd; $w_m = 4$ otherwise.

The next pixel in the path is similarly found, if any, by analyzing a suitable subset of the neighbors of n_k, selected depending on the direction $p \rightarrow n_k$.

Two disjoint increasing paths may originate from the same pixel only when this is the saddle pixel of a hourglass saddle configuration. In fact, only in this case, two 8-connected components of neighbors with higher label exist in the neighborhood of the saddle pixel. The neighbor maximizing the gradient has to be found in both the components, and two paths along the direction of the steepest gradient have to be grown. On the (3,4)-*DT*, the hourglass configuration may be centered only on pixels labeled either 3 or 4. See Fig. 10. Thus for pixels labeled 3 and 4, the possibility of growing a second path is checked before the raster scan inspection is resumed.

A diagonally oriented connecting path may be created in a direction parallel to that of another path and very close to it. It may even happen that pixels in a path have neighbors in the other path. When this is the case, spurious holes are created in the set of the skeletal pixels, which have to be filled in to guarantee topology preservation. Spurious holes are one-pixel-sized. They are filled in during the next step of the skeletonization process, devoted to reduction to unit width of the set of the skeletal pixels.

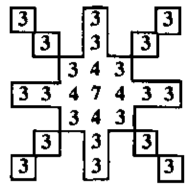


FIG. 11. All the pixels are marked as skeletal pixels.

Sparse 8-internal pixels may exist in the set of the skeletal pixels, due to the convergence of several paths towards common positions. An example is shown in Fig. 11. The existence of artificial lace-edged patterns (Fig. 12), in correspondence to which a unit wide skeleton cannot be obtained, is prevented by the cleaning step accomplished before the (3,4)-*DT* is computed.

In Fig. 13, the set of the skeletal pixels is superimposed over the input pattern. The pattern is completely recovered starting from this set.

4.2. Hole Filling and Final Thinning

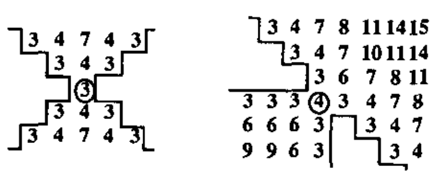
Reduction to unit width of the set of the skeletal pixels is performed within one raster scan inspection, by erasing the marker from any marked pixel p satisfying both the following conditions:

(1) At least one odd-neighbor of p is not marked;

(2) At least a triple of neighbors n_k, n_{k+2}, n_{k+5} exists (k odd, addition modulo 8) such that n_k and n_{k+2} are marked, while n_{k+5} is not marked.

Condition (1) prevents creation of holes in the set of the skeletal pixels, while condition (2) prevents both altering the connectedness, and shortening skeleton branches.

During the same inspection, any nonmarked pixel p whose odd-neighbors are all marked pixels is recognized as a spurious hole, and is accordingly marked. Marking p



a b

FIG. 10. Two paths along the direction of the steepest gradient originate from the same saddle pixels (encircled pixels).

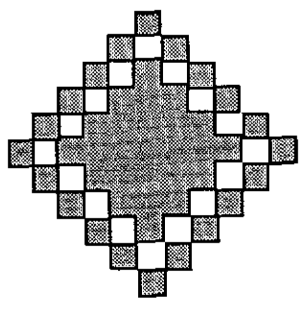


FIG. 12. A lace-edged pattern, whose one-pixel-sized holes are filled in during the cleaning step.

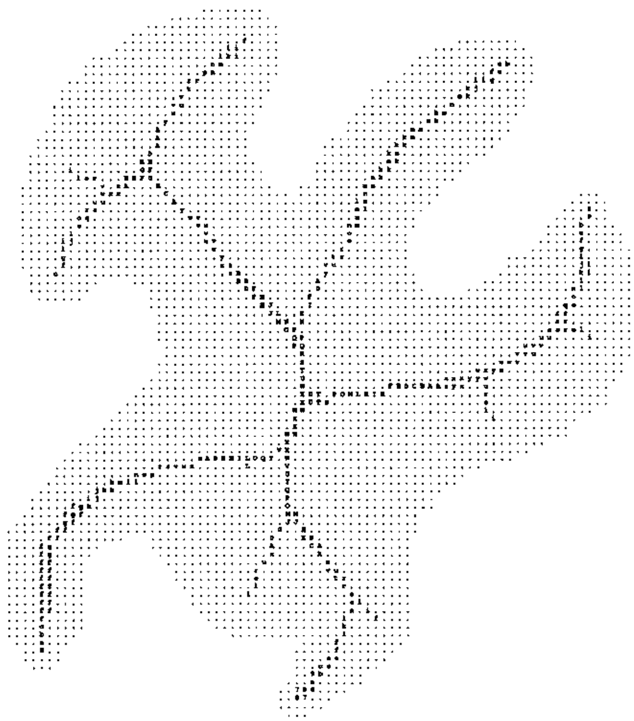


FIG. 13. The skeletal pixels found in the (3,4)-DT of a pattern, superimposed over the pattern.

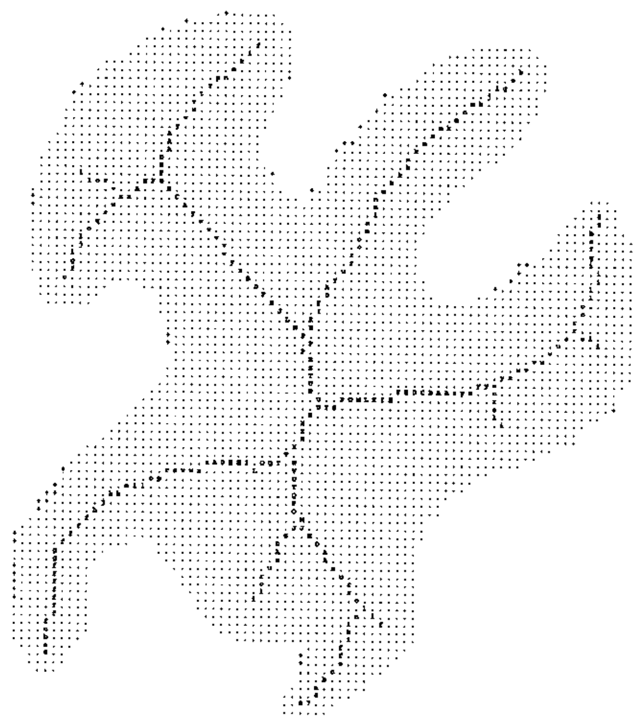


FIG. 15. The skeletal pixels remaining in the unit wide skeleton. Pixels which are not recovered starting from the unit wide skeleton are indicated by "+."

could make it possible to remove the marker from some of its already inspected odd-neighbors (see Fig. 14). These neighbors are, then, checked again before continuing the raster scan inspection.

For any skeletal pixel p which maintains the skeleton marker, the crossing number $X_4(p)$ is computed so as to count the components of pixels not belonging to the skeleton, placed in $N(p)$. When it is $X_4(p) = 1$, p is likely to be an end point in the resulting skeleton. Its coordinates are stored to directly access the pixel during the pruning step.

In Fig. 15, the set of the skeletal pixels is shown, as obtained after the reduction to unit width. Pixels denoted by "+" are input pixels which are not recovered, due to reduction to unit width of the set of the skeletal pixels.

4.3. Pruning and Beautifying

Pruning is done by erasing the marker from the pixels of a (subset of a) branch, which is not significant in the problem domain.

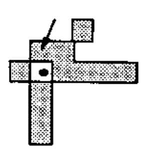


FIG. 14. After the spurious hole (black dot) has been filled, the pixel pointed to by the arrow becomes a superfluous pixel.

Let p be the end point delimiting a skeleton branch, and q any pixel along the branch, labeled more than p. If the disc D_q almost completely recovers D_p, the marker of all the pixels in the branch from p to q, q excluded, can be removed without noticeably diminishing the descriptive power of the skeleton. To judge about the degree of overlapping between D_q and D_p, we evaluate the number of rows and/or columns of the pattern protrusion mapped in the examined skeleton branch, which would not be recovered if the skeleton branch were pruned up to its pixel q. This is done by comparing the horizontally/vertically aligned radii of D_q and D_p, while taking into account the (3,4)-distance between p and q. If θ is 3 times the maximum number of peripheral rows and/or columns we are willing to miss, pruning can be accomplished if

$$(3)\quad p - q + d_{p,q} \leq \theta,$$

where $d_{p,q}$ is the (3,4)-distance between p and q.

To identify the most internal pixel q up to which pruning can be done, the skeleton branch is traced starting from the end point p. The pruning condition (3) is checked for any pixel q labeled more than p. Some tolerance on the label of q can be accepted to favor pruning. For instance, condition (3) can be checked also in correspondence to pixels labeled less than p, provided that they belong to the same layer as p. Satisfying condition (3) does not imply that pruning is immediately accom-

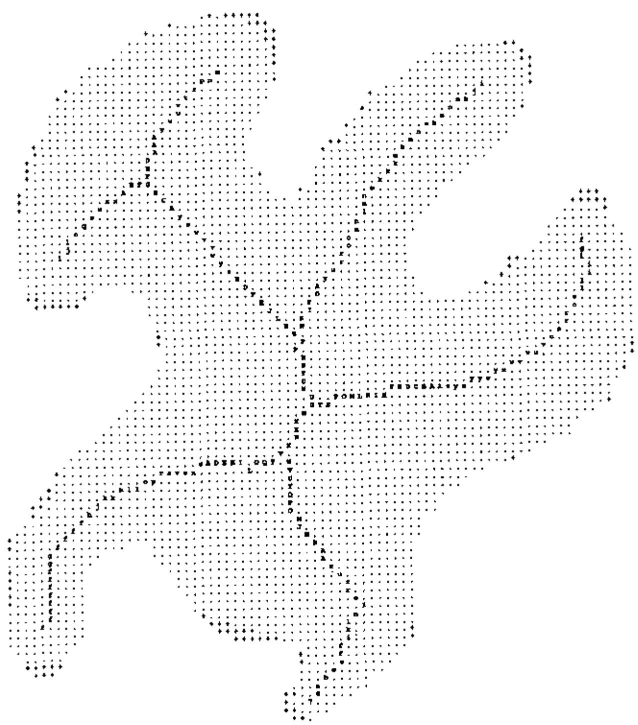


FIG. 16. The skeleton after pruning.

plished, as further pixels along the skeleton branch could as well satisfy the pruning condition. Accordingly, the coordinates of q are stored in a register R. Tracing is continued and the register R is updated as far as the pruning condition is satisfied.

Tracing is interrupted when, for the current pixel, condition (3) is not satisfied. The most internal pixel q up to which pruning can be safely accomplished is the pixel whose coordinates are stored in the register R. Tracing is also interrupted if the current pixel in correspondence to which the pruning condition is satisfied, is the second extreme of the peripheral branch. In this case, after the skeleton branch has been pruned, the connectivity number of the branch point is computed, to count the number of components of skeletal pixels in its neighborhood. If the connectivity number is equal to 1, the marker is removed, since the pixel has become superfluous for preserving the 8-connectedness. In this case, the same attempt to remove the marker is made on the neighbors of the branch point.

In Fig. 16, the pruned skeleton is shown superimposed over the input pattern. Pixels denoted by "+" are not recovered, starting from the pruned skeleton.

To reduce skeleton jaggedness, the marker is possibly shifted from a pixel p to one of its neighbors q. This is done if the following conditions are satisfied:

(4) p has exactly two neighbors, n_k and n_{k+2} (k even, addition modulo 8), in the skeleton;

(5) q is the odd-neighbor n_{k+1} of p.

Condition (5) can be more strict, to avoid marking a pixel q which is not entitled to be a skeletal pixel. To do this, one should keep track of all the pixels identified as skeletal pixels during the process, and allow shifting the skeleton-marker from p to q only provided that q is recorded in the list of the skeletal pixels.

In Fig. 17, the resulting skeleton is shown.

5. CONCLUSION

A distance driven skeletonization algorithm has been presented. Almost faithful pattern recovery is guaranteed, as the skeleton includes all the centers of the maximal discs, except those removed to gain unit width or removed during the pruning step and the beautifying step. The pixels which constitute the end points of the resulting skeleton have been ascribed to it, by implicitly using a reliable geometric criterion. In fact, they are the centers of the maximal discs located in correspondence with the tips of the pattern protrusions.

The selection of the maximal centers, the use of a quasi-Euclidean distance function, and the adopted pruning-and-beautifying step produce a well-shaped skeleton, only slightly sensitive to the orientation of the pattern. In Fig. 18, the results of applying the proposed algorithm to the same pattern in different orientations are shown. The same threshold value has been adopted during pruning.

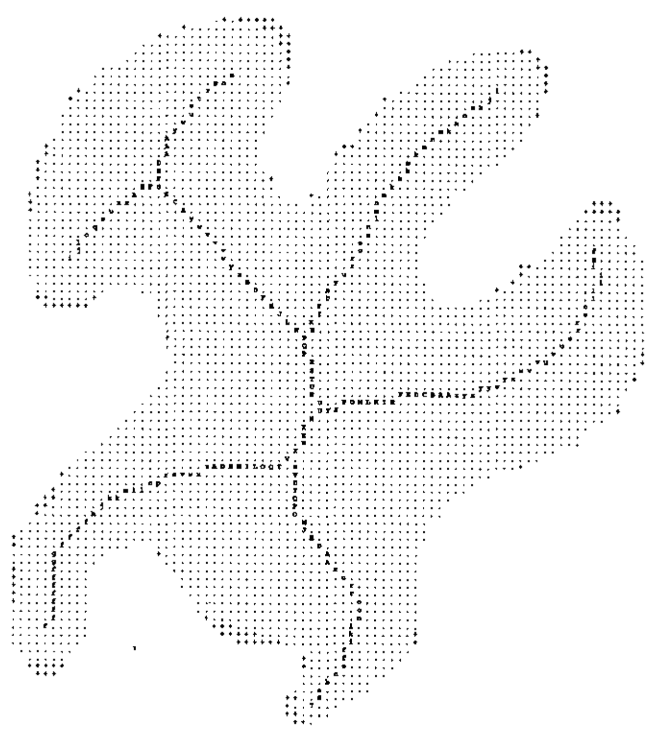


FIG. 17. The skeleton resulting after the beautifying step.

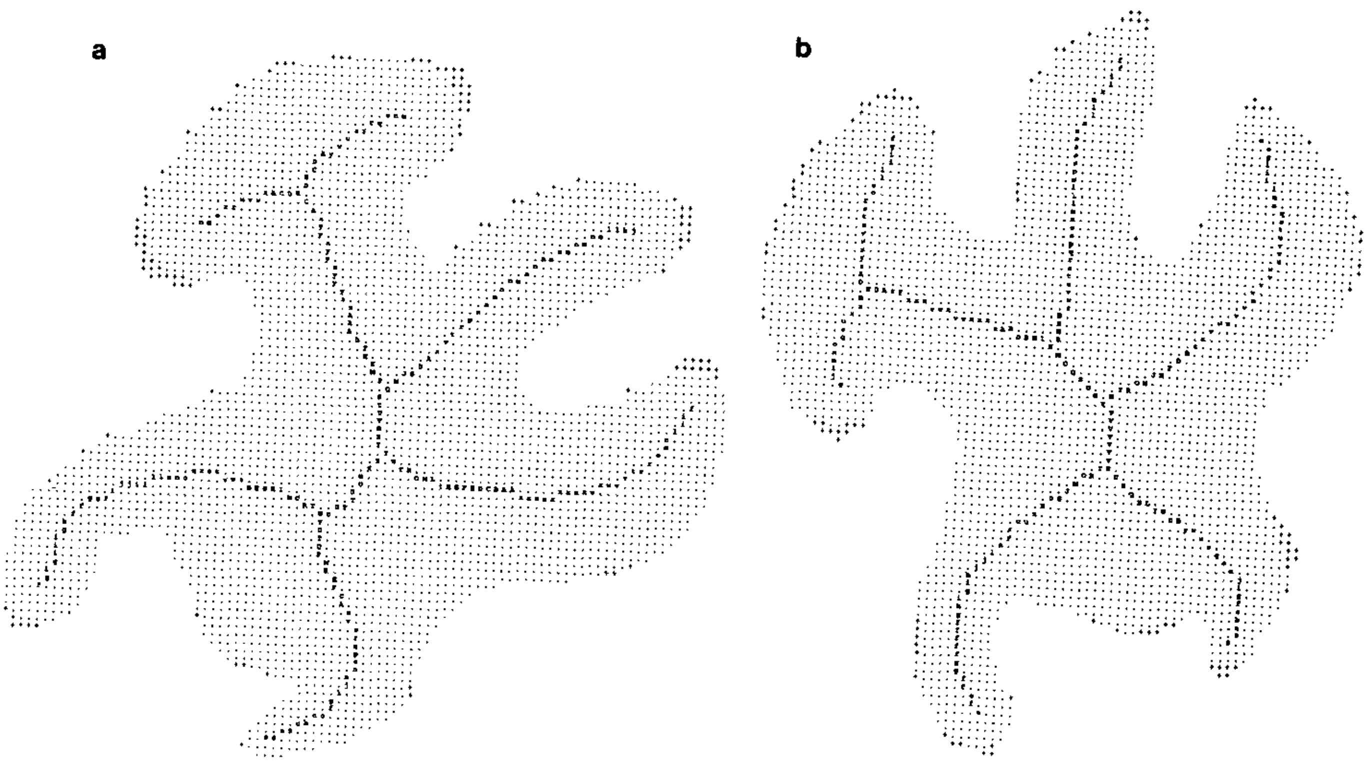


FIG. 18. The skeletons obtained after the pattern of Fig. 17 has been digitized in different orientations (a) and (b).

The computational cost of the process is rather modest and is independent of the thickness of the pattern to be skeletonized.

REFERENCES

1. G. Borgefors, Distance transformations in arbitrary dimensions, *Comput. Vision Graphics Image Process.* **27,** 1984, 321–345.
2. G. Borgefors, Distance Transformation in Digital Images, *Comput. Vision Graphics Image Process.* **34,** 1986, 344–371.
3. I. Ragnemalm, Contour processing distance transforms, in *Progress in Image Analysis and Processing* (V. Cantoni, L. P. Cordella, S. Levialdi, and G. Sanniti di Baja, Eds.), pp. 204–212, World Scientific, Singapore, 1990.
4. C. Arcelli and G. Sanniti di Baja, Finding local maxima in a pseudo Euclidean distance transform, *Comput. Vision Graphics Image Process.* **43,** 1988, 361–367.
5. C. Arcelli and G. Sanniti di Baja, Weighted distance transforms: A characterization, in *Image Analysis and Processing II* (V. Cantoni, V. Di Gesu', and S. Levialdi, Eds.), pp. 205–211, Plenum, New York, 1988.
6. D. Rutovitz, Pattern recognition, *J. Roy. Statist. Soc.* Ser. A, **129,** 1966, 504–530.
7. S. Yokoi, J. I. Toriwaki, and T. Fukumura, An analysis of topological properties of digitized binary pictures using local features, *Comput. Graph. Image Process.* **4,** 1975, 63–73.
8. C. J. Hilditch, Linear skeletons from square cupboards, in *Machine Intelligence IV* (B. Meltzer and D. Michie, Eds), pp. 403–420, 1969.
9. C. Arcelli and G. Sanniti di Baja, A width-independent fast thinning algorithm, *IEEE Trans. Pattern Anal. Mach. Intelligence* **7,** 1985, 463–474.
10. C. Arcelli and G. Sanniti di Baja, A one-pass two-operations process to detect the skeletal pixel on the 4-distance transform, *IEEE Trans. Pattern Anal. Mach. Intelligence* **11,** 1989, 411–414.
11. L. Dorst, Pseudo-Euclidean skeletons, in "Proc. 8th Int. Conf. on Pattern Recognition, Paris, 1986," pp. 286–288.

GABRIELLA SANNITI DI BAJA received the doctoral degree in physics from the University of Naples, Naples, Italy, in 1973. Since then, she has been working at the Institute of Cybernetics of the National Research Council of Italy, where she currently has the position of director of research. Her main research activities concern two dimensional shape representation and description. Since 1991, G. Sanniti di Baja has been chairman of the Education Committee of the International Association for Pattern Recognition.