ELSEVIER

Pattern Recognition Letters 18 (1997) 465-472

Pattern Recognition Letters

Efficient shape representation by minimizing the set of centres of maximal discs/spheres

Gunilla Borgefors [a,1], Ingela Nyström [b,*]

[a] *Centre for Image Analysis, Swedish University of Agricultural Sciences, Sweden*

[b] *Centre for Image Analysis, Uppsala University, Lägerhyddvägen 17, S-752 37 Uppsala, Sweden*

Received 9 August 1996; revised 12 November 1996

Abstract

Efficient shape representations are important for many image processing applications. Distance transform based algorithms can be used to compute the set of centres of maximal discs/spheres, that represents a shape. This paper describes a method that reduces this set, under the constraint that the shape can be exactly reconstructed using the reverse distance transformation. The reduced set can be used in the same ways as the "standard" set, e.g. for efficient storage, segmentation into parts of different thickness, shape manipulation, and skeletonization, all in 2D and 3D. © 1997 Elsevier Science B.V.

Keywords: Shape representation; Maximal discs/spheres; Reverse distance transform; Skeleton

1. Introduction

In Fig. 1 a shape constructed by overlapping two digital discs (in the chessboard metric) is shown to the left. In the middle the same shape is shown with its two necessary and sufficient centres of maximal discs. If the centres are labelled with the respective radii, this is an efficient representation of the shape. This is, however, not the set of centres of maximal discs resulting from thinning, which is shown to the right. Here we present a reduction algorithm for the standard set of Centres of Maximal Discs/Spheres (CMD/CMS), which is a promising approach for quantification and manipulation of shape. It also promises to decrease the storage requirements significantly, which is important in 3D. The original shape can still be exactly reconstructed. In the case in Fig. 1, our method will result in the desired set of CMD in the middle image.

Our approach to obtain a reduced set of pixels/voxels is to compute the set of CMD/CMS from the distance transform of a binary image (Arcelli and Sanniti di Baja, 1988; Borgefors et al., 1991; Borgefors, 1993). While it is true that no maximal disc/sphere is completely covered by a single other maximal disc/sphere (this is in fact the definition of CMD/CMS), it is common that a maximal disc/sphere is covered by a *set* of other maximal discs/spheres. Thus, the standard set of CMD/CMS contains a lot of unnecessary data. Our algorithm removes (most of) these redundant CMD/CMS; hence, the shape representation is further reduced. In 2D, examples are shown for the city block (D^4) and chessboard (D^8) metrics, for the weighted 3-4 and 5-7-11 metrics, and for the Euclidean metric. In 3D, examples are shown for the D^6, D^{26} and weighted

* Corresponding author. E-mail: ingela@cb.uu.se.

[1] E-mail: gunilla@cb.uu.se.

PII S0167-8655(97)00027-5

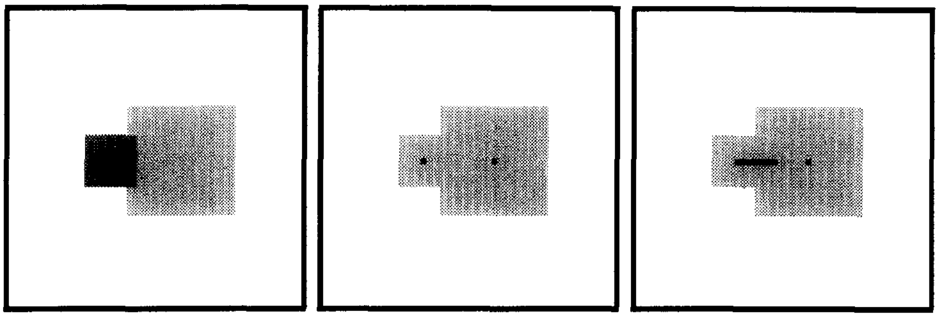

Fig. 1. Two fused digital discs (left). The desired set of centres of maximal discs in black (middle). The standard set of centres of maximal discs in black (right).

3-4-5 metrics. In a weighted metric the local distance between a pixel/voxel and its neighbours is set to different values for different neighbours, e.g. in the 3-4 metric the local distance between edge-neighbours is 3 and between point-neighbours 4. See (Borgefors, 1984) for details on different metrics.

2. Centres of maximal discs/spheres

To compute the CMD in 2D, the binary image containing the shape(s) is first converted to a distance transform. To detect the CMD in the distance transform, one raster scan is performed. The deletion or retention of each pixel depends on the configuration of the pixels in a local neighbourhood. How to compute the set of CMD has been described earlier in the literature for the city block, chessboard, 3-4 (Arcelli and Sanniti di Baja, 1988), 5-7-11 (Borgefors, 1993), and Euclidean (Borgefors et al., 1991) metrics. In the city block and chessboard distance transforms, the CMD are simply the local maxima.

The same principle can be used for volumetric (3D) shapes as for flat shapes, where the algorithm uses a $3 \times 3 \times 3$ neighbourhood. In the D^6 and D^{26} distance transforms the CMS are those voxels which are local maxima. The 3-4-5 metric is a reasonably good integer approximation of the Euclidean metric (Borgefors, 1996). The CMS are detected in the 3-4-5 distance transform in one raster scan. (The local distances are 3, 4 and 5 for face-, edge and point-neighbours, respectively.) As for the 3-4 metric, the distance value 3 first has to be substituted by the equivalence label 1, to avoid detection of false CMS (Arcelli and Sanniti di Baja, 1988). If the current voxel has a greater value than each neighbouring voxel minus the corresponding local distance to the current voxel, it is a CMS.

The shape is easily reconstructed from the set of CMD/CMS using the *reverse* distance transformation, see the next section.

3. Reverse distance transformation

The reverse distance transformation can be used to reconstruct a shape from its set of CMD/CMS. It is also used in the reduction algorithm described in the next section.

The input to the reverse distance transformation is a grey-level image with seed distance points in different positions, for example the set of CMD/CMS. The rest of the image is set to zero. Like the distance transformation, the reverse distance transformation requires raster scans (two, except for the Euclidean distance) of the image, during which a number of the neighbours of every pixel/voxel are taken into account (Borgefors, 1984; Ragnemalm, 1993). The algorithm performs a propagation with decreasing distances. Every pixel/voxel is assigned the maximum of the current pixel/voxel value and the difference between its already visited neighbouring pixels/voxels and the corresponding local distance to the current pixel/voxel.

The algorithm for 2D images can be found in (Arcelli and Sanniti di Baja, 1988; Borgefors et al., 1991), and for volume images in (Nyström and Borgefors, 1995).

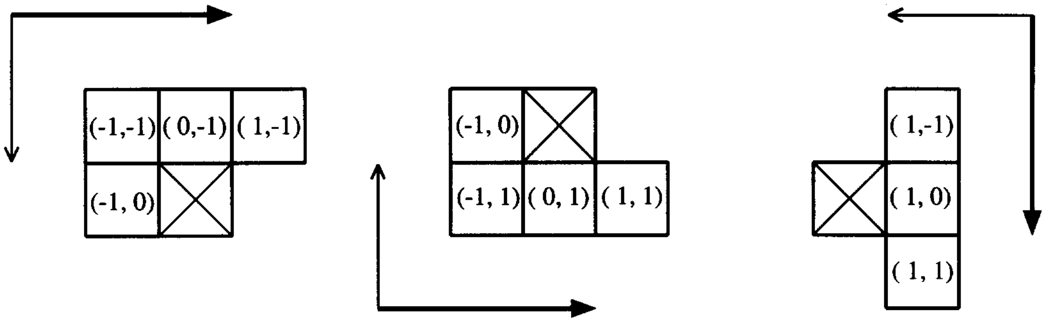

Fig. 2. The three masks used in the reverse Euclidean distance transformation. Row-wise scan from upper left to lower right (left), row-wise scan from lower left to upper right (middle), and column-wise scan from upper right to lower left (right).

3.1. Euclidean metric

The reverse distance transformation is straightforward to implement for all the metrics used, except for the Euclidean one (Borgefors et al., 1991). Difficulties occur, that can be overcome, but special care is required when designing the algorithm even in the 2D case. Therefore, we describe it here. To maintain precision we want to use integer arithmetic as much as possible. Hence, the seed distances should be the *squared* Euclidean distances. In three raster scans, values are propagated in three images. The first image is initially the set of squared CMD. Eventually its other pixels will contain the value of the closest CMD. The other two images contain the horizontal and vertical distances, respectively, to the closest CMD, and are initialised to zero (vector length zero). Together they form the vector pointing to the position of the closest CMD. At each pixel position, some of the eight neighbours of the current position, see the masks in Fig. 2, from (Ragnemalm, 1993), are candidates to propagate their three values. The sum of each neighbour vector and the corresponding mask vector form a candidate vector. Real arithmetic is necessarily introduced when the candidate distance values are computed. A candidate distance value for the current position is the difference between the square root of a neighbour and the length of its candidate vector. The highest candidate distance value gives values to the current position in the three images; a CMD value and the horizontal and vertical distances to the CMD. A simplification of (Borgefors et al., 1991) is that there is no need to distinguish between pixels that are and are not CMD; the corresponding vector lengths are zero for the CMD. The extra vector length computation requires less time than checking whether every pixel belongs to the set of CMD. Finally, the true distance value for each pixel is computed as the difference between the square root of the value in the first image (the closest CMD) and the length of the corresponding vector. If only the shape is needed, it is found in the first image, as the set of pixels with values greater than zero.

From the experiences with implementing the reverse Euclidean distance transformation for 2D images, together with the fact that at least four raster scans are necessary (Ragnemalm, 1993), we have decided not to extend it to volume images. It is quite possible though, if desired, but the simpler 3-4-5 distance transformation should be a fair approximation in many cases.

4. Reduction algorithm

When the set of CMD/CMS has been computed, a reduction of the set by removing pixels/voxels containing redundant information can be performed. A CMD/CMS is redundant when its disc/sphere is covered by the union of some other discs/spheres; the shape can be reconstructed from the other discs/spheres.

One approach to reducing this set is to keep a relation table with a column for every CMD/CMS and a row for every border pixel/voxel in the shape (Nilsson and Danielsson, 1996). A table entry indicates whether a CMD/CMS covers a border pixel/voxel.

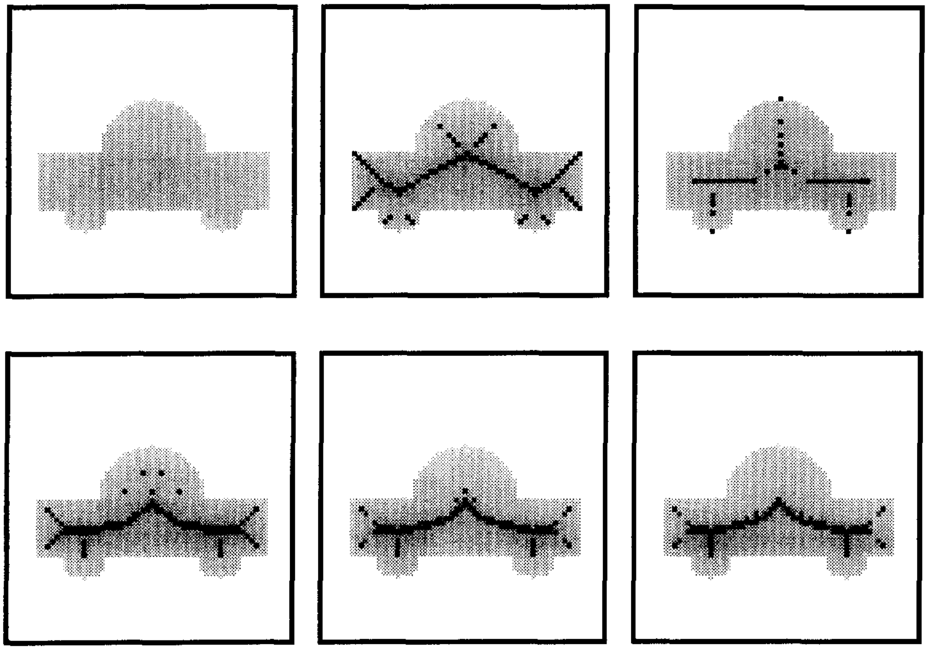

Fig. 3. The "car silhouette" example (top-left). The set of CMD is imposed in dark gray. The reduced set is imposed in black. The different metrics are city block (top-middle), chessboard (top-right), 3-4 (bottom-left), 5-7-11 (bottom-middle) and Euclidean (bottom-right).

The table is then reduced. This table will become a large data structure with increasing shape size (especially for volumetric shapes), which is a computational drawback. As only border pixels/voxels are considered, post-processing is necessary to remove holes/cavities, i.e. to cover interior parts not covered by the chosen set of CMD/CMS.

In our approach there are no complex data structures, and no need of post-processing, to produce a small set of CMD/CMS that is enough to reconstruct the shape. The 2D reduction algorithm has been described for the Euclidean distance transform in (Ragnemalm and Borgefors, 1993). The algorithm for volume images (Borgefors and Nyström, 1995) is a generalization of the 2D algorithm.

Step 1. A grey-level image containing the standard set of CMS is the input. The set of CMS is sorted, together with their positions, into a list according to increasing distance value.

Step 2. A temporary image is used for storing the number of spheres covering each voxel in the original shape. This image is obtained by traversing the list of CMS, generating the associated sphere for each distance value, and incrementing all the voxels in the temporary image corresponding to the spheres.

Step 3. The list is traversed once more. For each CMS, inspect the corresponding sphere in the temporary image. If all its voxels have values greater than one, it is covered by at least one other maximal sphere. The CMS is not necessary for reconstructing the shape and can be removed. All voxels of the corresponding sphere are then decremented.

In Step 3 it is necessary to process the list in increasing order to remove smaller spheres rather than larger spheres, as large spheres are perceived as more significant, and to increase the possibility of removing as many spheres as possible. For efficiency reasons, the sorted list is used in Step 2, as well. Generation of the associated spheres, by using the reverse distance transformation on the distance value, will then only be performed once per sphere size, rather than once per sphere. The spheres are, of course, generated with the same metric that was used for the distance transformation and the computation of the set of CMS.

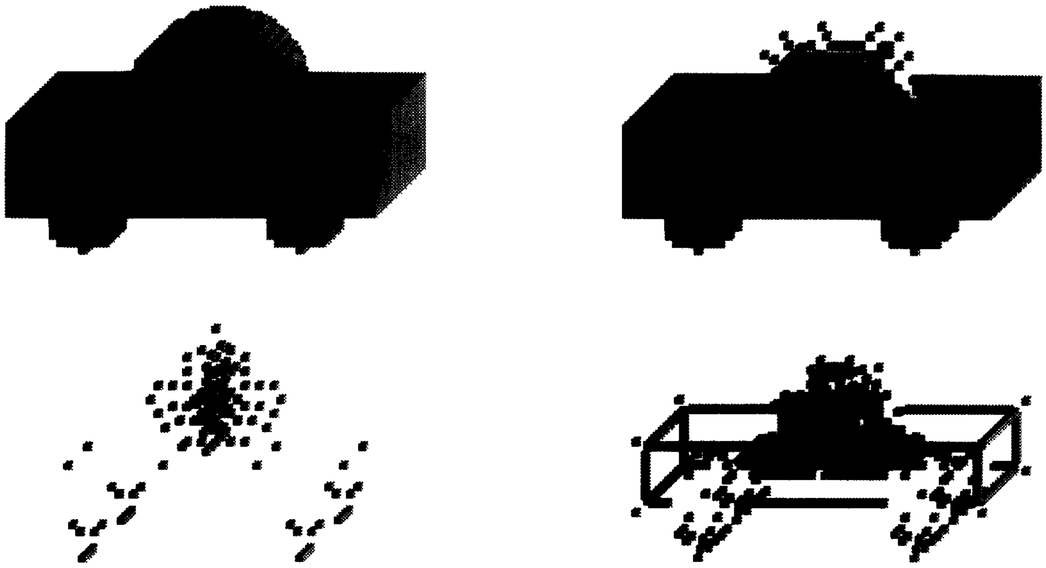

Fig. 4. Rendered 3D car in different representations. Original (top-left). The reduced sets of CMS for the D^6 metric (top-right), D^{26} metric (bottom-left), and 3-4-5 metric (bottom-right). (Thanks to Dr Pieter Jonker, Delft University, The Netherlands, for letting us use the volume visualization software.)

4.1. Implementation complexity

Our implementation is written in C. When run on a DEC Alpha (a standard UNIX workstation) the reduction time is, for example, in the order of a few minutes for shapes in 128 × 128 × 128 (2 Mbyte) images. The reconstruction from the reduced set takes approximately 10 seconds.

Our reduction algorithm is using a list for the set of CMD/CMS and their positions, and two temporary images (one to generate discs/spheres and one to store the number of discs/spheres covering each pixel/voxel in the original shape), so there are no complex data structures.

5. Examples

The example in Fig. 4 (top-left) is a synthetic "car" in a 64 × 64 × 64 image. It is constructed by combining a Euclidean sphere, and a rectangular block. Four short cylinders have been used as wheels. The silhouette of the car is used as an example in the 2D case. The car contains both curved and sharp borders, so none of (or all!) the different distance transforms are favoured. Fig. 3 shows the standard and reduced sets of CMD for different 2D metrics. Fig. 4 similarly shows different reduced sets of CMS for the 3D car.

A comparison can be made of the reduction rates of the reduced set compared to the standard set of CMD/CMS for different metrics. In this example, the standard set of CMD is almost the same as the reduced set for the city block metric, while for the Euclidean metric there is a major reduction, and the standard set of CMS coincides with the reduced set for the D^6 metric, while the 3-4-5 metric results in a reduction to less than half the set. See Table 1 for details.

A non-synthetic example is a hand in a 170 × 150 × 120 image. In Fig. 5 the result of a reduction of the hand can be seen. The original hand contains more than 400000 voxels. By representing it with its set of CMS, 8.4% of the voxels are needed. The reduced set contains 5.8% of the original voxels.

6. Conclusions

The standard set of centres of maximal discs/spheres is usually far from optimal, in the sense that many CMD/CMS are not necessary for shape recon-

Fig. 5. To the left a hand. To the right a reduction of the hand using the 3-4-5 metric, containing 5.8% of the original voxels. (The copyright holders of this hand are Professors Jun-ichiro Toriwaki and Kazuhiro Katada, Nagoya University, Japan.)

Table 1
The number of CMD/CMS for the "car" before and after reduction. The original shapes contain 941 pixels and 17919 voxels, respectively

Metric	Standard set		Reduced set	
	pixels	%	pixels	%
City block	79	8.4	73	7.8
Chessboard	49	5.2	20	2.1
3-4	99	10.5	21	2.2
5-7-11	93	9.9	15	1.6
Euclidean	94	10.0	14	1.5
	voxels	%	voxels	%
D^6	2223	12.4	2223	12.4
D^{26}	515	2.9	131	0.7
3-4-5	2351	13.1	931	5.2

struction. Our algorithm reduces this set under the constraint that the shape still can be exactly reconstructed. The reduction of data is of even greater importance for volume images than for 2D images. The reduction rate is naturally both shape dependent and metric dependent, see Table 1. Note that the remaining set of CMD/CMS is not necessarily optimal, as the standard set is inspected in a fixed order. However, the results should be close to optimal in most cases.

By discarding pixels/voxels having values less than a given threshold, the set may be further reduced, at the expense of losing fine details in the reconstructed shape. This can, in a more positive sense, be seen as noise reduction.

The set of CMD/CMS is an equivalent representation of the original shape, and can be used in itself, both for storing, manipulation and segmentation. It is, however, in all but the trivial cases, not connected and thus not topologically equivalent to the shape. A thin, topologically correct representation of a shape is the skeleton (Lam et al., 1992). Distance transform based algorithms have been used to generate skeletal representations of shape in 2D, e.g. (Sanniti di Baja and Thiel, 1996).

The standard method to compute a skeleton based on a distance transform is to first detect the set of CMD and the saddle points, and then grow linking paths through the ascending gradient in the distance transform between them. As the set of CMD is "thick" (see Fig. 3), the skeleton will need post-processing to reduce it to unit thickness. This will remove CMDs arbitrarily. If the set of CMD is first reduced as described here, the skeleton will become thinner and needs less post-processing. The final skeleton should contain more of the significant CMDs. The drawback is that many of the pixels that are removed will have to be added again, as they are necessary for connectedness.

To date, not much has been written on distance transform based skeletonization algorithms for volume images, but see (Saito and Toriwaki, 1995). The task is difficult due to the difficulty of identifying "saddle points" in the distance transform. Further work is necessary.

Acknowledgements

This work is a collaboration between the two universities at the Centre for Image Analysis. We thank Professor Ewert Bengtsson and Dr Bo Nordin for provided support. A special thanks also to Dr Ingemar Ragnemalm for doing the work in the 2D Euclidean case.

References

Arcelli, C. and G. Sanniti di Baja (1988). Finding local maxima in a pseudo-Euclidean distance transform. *Computer Vision, Graphics, and Image Processing* 43, 361–367.

Borgefors, G. (1984). Distance transformations in arbitrary dimensions. *Computer Vision, Graphics, and Image Processing* 27, 321–345.

Borgefors, G. (1993). Centres of maximal discs in the 5-7-11 distance transform. In: *Proc. 8th Scandinavian Conf. on Image Analysis*, Tromsø, Norway, pp. 105–111.

Borgefors, G. (1996). On digital distance transforms in three dimensions. *Computer Vision and Image Understanding* 64 (3), 368–376.

Borgefors, G. and I. Nyström (1995). Quantitative shape analysis of volume images – Reducing the set of centres of maximal spheres. In: *Proc. SSAB (Swedish Society for Automated Image Analysis) Symp. on Image Analysis*, Linköping, Sweden, pp. 5–8.

Borgefors, G., I. Ragnemalm and G. Sanniti di Baja (1991). The Euclidean distance transform: Finding the local maxima and reconstructing the shape. In: *Proc. 7th Scandinavian Conf. on Image Analysis*, Aalborg, Denmark, pp. 974–981.

Lam, L., S.-W. Lee and C.Y. Suen (1992). Thinning methodologies – A comprehensive survey. *IEEE Trans. Pattern Analysis Machine Intelligence* 14 (9), 869–885.

Nilsson, F. and P.-E. Danielsson (1996). Finding the minimal set of maximum disks for binary objects. In: *Proc. SSAB (Swedish Society for Automated Image Analysis) Symp. on Image Analysis*, Lund, Sweden, pp. 44–48.

Nyström, I. and G. Borgefors (1995). Synthesising objects and scenes using the reverse distance transformation in 2D and 3D. In: C. Braccini, L. DeFloriani and G. Vernazza, Eds., *Proc. ICIAP'95: Image Analysis and Processing*. Springer, Berlin, pp. 441–446.

Ragnemalm, I. (1993). The Euclidean distance transform in arbitrary dimensions. *Pattern Recognition Letters* 14 (11), 883–888.

Ragnemalm, I. and G. Borgefors (1993). Towards a minimal shape representation using maximal disks. In: I. Ragnemalm, *The Euclidean Distance Transform*, Chapter 9. Ph.D. Dissertation, No. 304, Linköping University, Linköping, Sweden.

Saito, T. and J.-I. Toriwaki (1995). A sequential thinning algorithm for three dimensional digital pictures using the Euclidean distance transformation. In: *Proc. 9th Scandinavian Conf. on Image Analysis*, Uppsala, Sweden, pp. 507–516.

Sanniti di Baja, G. and E. Thiel (1996). Skeletonization algorithm running on path-based distance maps. *Image and Vision Computing* 14, 47–57.