УДК 681.3

S.V.Ablameyko+, C.Arcelli*, G.Sanniti di Baja*

EXTRACTION AND REPRESENTATION OF LINE DRAWING OBJECTS

INTRODUCTION

The process for automatic input and conversion of 2D line-drawings into digital vector models can be divided into two main stages: raster-to-vector trasform of a binary image to obtain its structural representation and recognition of primitives, graphical elements, scenes, and 2D objects. Most of the existing systems use intermediate vector representation for the object recognition which is the interface between these two stages and allows to formalize high-level processing and increase its speed.

To represent line-drawing objects at intermediate level, it is necessary to extract a set of object components from line-drawings that will be useful for object recognition and store this information in a special vector model. The more suitable representation of line-drawing objects for their recogntion is a skeleton which is a linear subset of the object and is characterized by the same topological and geometrical structure. There are different approaches for vector image representation based on object skeletons. The approaches used for engineering drawings processing usually detect critical points at the first step, then extract arcs, conics, and the remaining parts are approximated by using polygonal approximation as it is made by Nagasamy and Langrana [1]. Suzuki [2], Shih and Kasturi [3], suggested to extract feature points (node and end points) and segments bounded by these points, and represent image objects in terms the segments. Object decomposition by contour curvature, object thickness and elongation have been proposed by Arcelli et al [4,5], Thiel and Montanvert [6].

A desirable skeleton decomposition method should produce meaningful results when applied to a wide set of classes of objects. It means that the object decomposition should be obtained in terms of meaningful components which will be clear from a human point of view. For exmaple, analysis of diagrams, electronic circuit images, several engineering drawings require that meaningful components like loops, and connected branches with their feature points should be extracted (Fahn et al [7], Shimotsuji et al [8]). Moreover, the decomposition method should be as much as possible stable under object rotation, translation and scaling. Finally, to facilitate object recognition, the components of the decomposition should be ranked in a hierarchical way.

In this paper, a method and special data structure is suggested which allows to describe compactly all basic characteristics of objects and their components, and is oriented to the structural recognition of objects. To obtain the data structure, we propose a decomposition method that at a certain extent satisfies the above requirements. To make the decomposition method more general, a number of features may be associated to the skeleton branches sharing a branch point. The selected features are used to form concatenations of more internal skeleton branches and to establish a hierarchy among the skeleton decomposiition components. The skeleton decomposition is obtained in terms of loops, branches and concatenations of branches.

COMPONENT EXTRACTION AND IMAGE VECTORISATION

Let B and W be respectively the two sets of the black pixels and of the white pixels, constituting a binary image digitized on the square grid. At the first step of our procedure, we calculate distance maps (DM) and identify entry points in correspondence with the holes of B. This last process requires hole labeling. DM computation and hole labeling are accomplished simultaneously and starting pixel from which to trace and label the white border of any hole of B is recorded in the list of the entry points LH. On the DM, skeletonization is accomplished by using the algorithm of Sanniti di Baja [9], which has a limited cost since most of the computation is accomplished within two raster scans. While computing the skeleton, two lists LE and LB are built, where the end points and the branch points are recorded. The branch points are also marked on the image where the skeleton is stored, so as to guarantee correct termination of skeleton branches during tracing.

To rank skeleton branches and loops, the notion of degree of interiority Id has been introduced [I0]. Peripheral skeleton branches are the less internal branches; their interiority degree is I. To identify branches with higher and higher interiority degree, we refer to the following pruning process, accomplished in parallel fashion and in such a way to preserve skeleton connectedness. First, every peripheral skeleton branch is identified by tracing it from its first extreme (the end point) up to its second extreme (the first encountered branch point). Then, peripheral branches are pruned by simultaneously removing all the traced pixels, except for the branch points. In the pruned skeleton, new peripheral branches are likely to be originated. In fact, some pixels identified as branch points before pruning, play the role of end points in the pruned skeleton. The interiority degree of the current peripheral branches is 2. In general, if k iterations of the parallel pruning process are necessary in order a branch point can play the role of end point, the interiority degree of the peripheral skeleton branch starting from it is k+I.

Let us suppose that parallel pruning has been applied until no more peripheral branches are left in the skeleton. Let the highest interiority degree of the so found skeleton branches be Id=h. What remains in the skeleton at this stage of the process consists of loops and, possibly, sets of branches connecting pairs of loops (linking branches).

To compute the interiority degree for the remaining skeleton components, all the loops are traced from their inside; loops for which only one cluster of branch points is met during tracing are the less internal ones and, as such, have interiority degree Id=h+I. All the pixels of the loops with Id=h+I are simultaneously removed (pruned), except for the possibly existing branch points, shared with more internal linking branches. Loop removal causes some of the linking branches to become peripheral branches. They have interiority degree Id=h+2. Starting from these peripheral branches, the already sketched parallel pruning process is applied, and more internal linking branches are assigned higher interiority degree. The loop-and-linking-branch process is iterated until only loops remain, which are all assigned the same interiority degree.

Every skeleton decomposition component is vectorized: for every pixel coordinates and label are recorded. Additional information is also stored in correspondence with every branch point to compute the relevance of the region associated with the vectorized skeleton component, as well as to record the adjacency of the skeleton component with other skeleton components.

The spatial relevance of a region is defined as the product of the length by the average width of the region. The spatial relevance is used to decide about the possible concatenation of a skeleton branch with an adjacent more internal skeleton branch. To measure the spatial relevance, we compute features derived from the skeleton branch representing the region and closely related to length and average width of the region. While tracing and vectorizing the skeleton branch, its pixels are simultaneously removed from the image except for the branch point, second extreme of the skeleton branch.

Once none of the pixels in the list LB plays the role of end point in the image, the list LH is used to start the vectorization of what is left in the skeleton. Starting from the pixel identified by the current entry point, the corresponding row of the image is backtraced until a pixel of the skeleton is met. Tracing of the loop is then accomplished from its inside, starting from this pixel, so as to count the number of encountered clusters of branch points. In fact, only loops for which just one cluster is met are peripheral loops and, as such, are the first ones to be vectorized.

After all the less internal loops have been vectorized, the possibly existing linking branches, that have been transformed into peripheral branches due to loop removal, are vectorized by accessing their extremes through the list LB. Alternate inspections of LB and LH are done until only loops directly linked to each other remain in the skeleton. At this stage of the process, none of the loops is a peripheral one, since for each of them at least two clusters of branch points are met. The remaining loops are all assigned to the next hierarchy level, without linking them to any of the previous levels. The loops are individually vectorized and constitute the highest level of the hierarchy.

VECTOR IMAGE REPRESENTATION

Consider what information is necessary to store for recognition of objects on 2D line-drawings. At first, information about object structure is necessary. It should include information about object components, their characteristics and relations between components. The component coordinates (metrics) is important when digital model of processed line-drawing is built. Information about object feature points is also needed especially when we recognise complex objects, scenes, disconnected objects, etc. When it is necessary to look over many components and objects, it is more faster to analyse component feature points than whole components. So, we can conclude that for line-drawing object recognition we need to store information about object components, their coordinates and characteristics, and object feature points. Let us consider how this information can be extracted from object and how it can be stored.

Important part of object description is information about object components. Generally it should include description of object component, relations with other components, and its parameters. We propose to store it in a special component description file and include the following information: I) the component identifier k; 2) the component decomposition level, i.e., the highest interiority degree of the decomposition component stored into Ak; 3) the identifiers of the components adjacent to Ak and belonging to the same level; 4) the identifiers of the components adjacent to Ak and belonging to upper levels; 5) the identifiers of the components adjacent to Ak and belonging to lower levels; 6) the spatial relevance of the skeleton decomposition component; 7) references to corresponding feature points; 8) references to component metric.

To fill the component description file, the files Ak are backtracked, with the exclusion of the possibly existing files including only the loops at the highest decomposition level. For a given Ak, each time a record with more than three fields Gi is passed through, the interiority degree along Ak assumes a smaller value. Backtracking is temporarily interrupted, so as to keep track of the adjacency relations of the decomposition component stored in Ak with other adjacent components. Track of the adjacency relation between the components stored in Ak and Ah is kept in the proper field among K3-K5. The proper field is identified by comparing the interiority degree of Ah, and the new interiority degree pertaining the less internal portion of Ak, still to be backtracked.

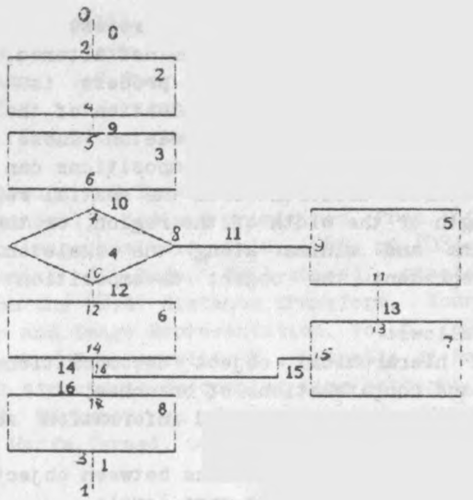
The second file contains description of feature points and reference to the first file to corresponding object components. The description of feature points includes I) the point identifier n; 2) the point type (branch or end point); 3) x,y point coordinates; 4)reference to the corresponding component in the component description file.

The third file contains metrics (component point coordinates) description. The coordinates of an approximated component line is obtained during the tracing files Ak. The connection between three files is performed by references. The example of data structure for an engineering drawing object is shown in Fig.I.

Further processing of document images shows that this reperesentation contains in compact form all needed information for the automatic object interpretation on 2D line-drawings.

CONCLUSION AND DISCUSSION

An object decomposition and description method has been presented, based on the notion of degree of interiority of a skeleton component (skeleton branch or loop). The method is



File I (component description)

KI	K2	K3	K4	K5	K6	K7	K8
0	I		2		87	0,2	N0
I	I		8		9I	I,3	NI
2	2	9		0	494	2,4	N2
3	2	9,I0			505	5,6	N3
4	2	I0,II,I2			7I5	7,8,I0	N4
5	2	II,I3			50I	9,II	N5
.	.	.			.	.	.

File 2 (description of feature points)

N	Type	X	Y	Ref to file I
0	I	x0	y0	0
I	I	xI	yI	I
2	2	x2	y2	0,2
3	2	x3	y3	I,8
.	.	.	.	.
.	.	.	.	.

File 3 (component coordinates

X	Y
x0I	y0I
xII	yII
x2I	y2I
x3I	y3I
x02	y02
xI2	yI2
.	.

Fig.2. Example of flow-chart and its description.

implemented by iterating a sequential process in such a way to guarantee that skeleton connectedness is not altered. The feature adopted to guide the concatenation process is the spatial relevance, i.e., an approximated evaluation of the area of the ribbon-like region associated with a skeleton subset. Depending on the problem domain, alternative decompositions can be obtained by choosing features different from the spatial relevance, for instance the length or the width of the region, or the number of curvature maxima and minima along the skeleton. The data structure to represent the object decomposition has been proposed.

Our approach allows:

- to obtain a hierarchical object decomposition in terms of loops, branches and concatenations of branches;
- to store in compact form all needed information about object components and feature points ;
- to have information about relations between object components from different levels and from the same level;
- to make an object recognition process more effective by using both semantical and structural information about object.

The proposed data structure can be applied to process various diagrams, electronic circuit images, flow-chart images, maps, and engineering drawings.

REFERENCES

I. V.Nagasamy, N.A.Langrana, Engineering drawing processing and vectorization system, Computer Vision, Graphics, and Image Processing, Vol.49, No.3, I990, pp.379-397.

2. S.Suzuki, Graph-based vectorization method for line patterns, Proc. IEEE Conf. on Computer Vision and Pattern Recognition, I988, pp.6I6-62I.

3. C.Shih, R.Kasturi, Extraction of graphic primitives from images of paper based line drawings, Machine Vision and Applications, No.2, I989, pp.I03-II3.

4.C. Arcelli, G. Sanniti di Baja, An approach to figure decomposition using width information, Computer Vision Graphics and Image Processing, Vol.26, I984, pp.6I-72.

5.C. Arcelli, R. Colucci and G. Sanniti di Baja, On the description of digital strips, in Proceedings, Int. Conference on Artificial Intelligence Applications and Neural Networks, Zurich, I990, pp.I93-I96.

6.E. Thiel and A. Montanvert, Shape splitting from medial lines using the 3-4 chamfer distance, in Visual Form Analysis and

Recognition, (C.Arcelli, L.P.Cordella and G. Sanniti di Baja, Eds.). Plenum, New York, 1992, pp.537-546.

7.C.-S.Fahn, J-F.Wang, J-Y.Lee, A topology-based component extractor for understanding electronic circuit diagrams, Comp.Vision, Graphics, and Image processing, Vol.44, 1988, pp.119-138.

8.S.Shimotsuji, S.Tamura, S.Tsunekawa, Model-based diagram analysis by generally defined primitives. Proc. 6th Scandinavian Conference on image analysis, Oulu, 1989, pp.1034-1041.

9.G. Sanniti di Baja, Well-shaped, stable and reversible skeletons from the (3,4)-distance transform, Journal of Visual Communication and Image Representation, Vol.5, 1994, pp.107-115.

10.Ablameyko S., Arcelli C., Sanniti di Baja, Finding and ranking basic structures on complex line patterns, Proceedings of International Workshop on Syntactic and Structural Pattern Recognition, Haifa,Israel, October 4-6, 1994.

+Institute of Engineering Cybernetics, Belarussian Academy of Sciences Surganova Str.6, 220012 Minsk, Belarus

*Istituto di Cibernetica, Consiglio Nazionale delle Ricerche Via Toiano 6, 80072 Arco Felice, Napoli, Italy