MORPHOLOGICAL CONTOUR CODING USING STRUCTURING FUNCTIONS OPTIMIZED BY GENETIC ALGORITHMS

P. Brigger and M. Kunt

Signal Processing Laboratory, Swiss Federal Institute of Technology
CH-1015 Lausanne, Switzerland
email: brigger@lts.de.epfl.ch

ABSTRACT

Shape representation is an important image analysis task which can be used for contour coding and feature extraction. The morphological skeleton is a geometrical shape description by means of maximal inscribed structuring elements. The form of the structuring element is usually chosen *a priori*, and we show how genetic algorithms can be used for an automatic optimization of an *arbitrary shaped* structuring element. It permits improved progressive contour transmission and the extraction of shape features.

1. INTRODUCTION

Modern telecommunication services have to respond to consumer needs that require a large number of services over a limited number of transmission channels. For video services, efficient compression techniques are necessary to assure an optimal occupation of a given bandwidth. Any such image coding system can be described by three abstraction levels: image partitioning, partition modeling and entropy coding of the model. Image partitioning by *blocks* of predefined size has proved to be efficient at high data rates (MPEG 2 [1]), whereas image partitioning by arbitrary shaped *regions* gives good results at very low data rates (second generation coding methods based on a contour-texture representation) [9]. In the present paper, we are situated in the second system, where the problem of contour coding is addressed.

Traditionally, the goal of a contour coding method is an accurate representation of the image partition. Freeman's chain code performs efficient and lossless coding [6]. Its main drawbacks are a high bit cost compared to the total bit–rate, and its inability to take into account the shape of the region to code. Indeed, it would be interesting to dispose of a contour coding tool that considers the *general shape* of the region rather than its individual contour points. It would allow a more flexible trade–off between accuracy and bit–cost. Furthermore, a shape description tool – considered as a feature – may also be useful in object description/analysis.

Such a description tool is the skeleton transformation [2, 10]. It is a geometrical representation by maximal disks that describes the interior of an object by using as primitive a disk. Two-dimensional regions are represented by using the disks which just fit inside. Thus, the description consists of two parts: the locus of the centers of the maximal disks and the associated radius of each disk. The transformation has found various applications in biological image analysis [2] and contour coding [3]. However, the *a priori* disk can not be regarded as a feature of the decomposed region, since it does not contain any information about the region shape.

Let us denote by $\delta_1(x)$ an arbitrary shaped structuring function for the skeleton decomposition, of which the disk is a special case (An example is shown in Fig. 1). Serra [11] has derived a skeleton formula that is valid for families of arbitrary shaped structuring functions that can be different from disks. Such a family may be constructed based on successive dilations of $\delta_1(x)$ by itself. Hence, an appropriately chosen structuring function $\delta_1(x)$ is representative for the region and can be regarded as a feature that gives information about edginess, circularity, length, width or orientation. Moreover, the optimal feature allows the compaction of the region contour points into a smaller number of skeleton points for an efficient contour representation. The topic of this paper shall be to propose a method for the definition of the shape of the optimal structuring function $\delta_1(x)$ that serves to define the whole family in [11].

The problem is one of minimization in a complex search space. Complex, because a structuring function $\delta_1(x)$ defined for example on a small 7×7 image sup-

---

This work was supported by the European RACE/MORPHECO project R2053

0-8186-7310-9/95 $4.00 © 1995 IEEE

Proceedings of the 1995 International Conference on Image Processing (ICIP '95)
0-8186-7310-9/95 $10.00 © 1995 IEEE

port gives rise to 2^{49} different feature shapes. Genetic algorithms (GAs) have been reported to be efficient, effective and robust for such kind of tasks [5] and we will show how they can be used.

The organization of the paper is as follows. In Section 2, the general scheme of a genetic algorithm is given. In Section 3, a shape–oriented crossover operator is presented that allows better convergence for our problem. Application results are given in Section 4 and finally, conclusions are drawn in Section 5.

2. GENETIC ALGORITHMS

A relatively new paradigm for search based on the principles of natural selection are genetic algorithms (GAs) [7]. They are probabilistic algorithms and provide a tool for investigating phenomena generated by complex adaptive systems. Therefore, the field of application of GAs perfectly suits our needs and may provide a solution to our optimization problem.

Three associated objects are characteristic for the optimization procedure: 1) the environment of the system undergoing adaptation, 2) the adaptive plan and 3) a measure of the performance of different structures.

The environment is given by the region shape and the performance measure is related to the number of skeleton points produced for a lossless reconstruction. The task of the GA is to control the mixture of operators that affect the system undergoing adaptation. The genetic algorithm carries out a simulated evolution dictated by the adaptive plan which should evolve towards a structure that has a maximal performance capability.

During the evolution process, the adaptive plan acquires knowledge about the structures that are most fit. For this purpose, a population is formed with a set of members that are chosen among the set of all possible structures. Each member is tested and evaluated within the environment and based on the individual member fitness, the adaptive plan draws a selection in order to reproduce a new population.

A member is called a chromosome and several chromosomes form a population. Like in biology, a chromosome's characteristic is determined by the *genes*. Each gene has several forms or alternatives which are called *alleles*, producing differences in the set of characteristics associated with that gene. A current population develops into a new population via the reproduction of chromosomes. The way the reproduction takes place is dictated by the adaptive plan and lies in the action of genetic operators, crossover and mutation, for example.

The link of the GA to the skeleton decomposition is made via a mechanism of encoding a structuring function $\delta_1(x)$ on a chromosome and the existence of an evaluation function. Encoding of the arbitrary shaped structuring function $\delta_1(x)$ is rather straightforward. Suppose $\delta_1(x)$ is defined on a $N \times M$ binary image. Because of its binary nature, two alleles are sufficient in the chromosome representation (0 = black and 1 = white). The chromosome can be a vector representation of the image when it is scanned in direct video scan (or any other scan). If a pixel element belongs to $\delta_1(x)$, its gene has a value of 1 and a value of 0 otherwise (Fig. 1). The evaluation function consists in counting the number of skeleton points in the skeletal decomposition. The smaller the number of skeleton points, the higher the worth of the structure.

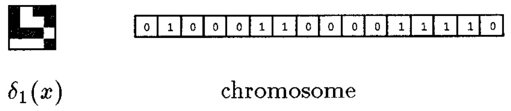

Figure 1: Encoding of a chromosome.

The adaptive plan creates a new population in two steps according to some reproductive plan: first, a certain number of chromosomes are selected within the initial population. Then the selection serves as templates (parents) on which genetic operators are applied, interchanging and modifying sets of alleles, to produce a new generation (offsprings). Our specific problem forces us to use rather small populations and to run the algorithm with only relatively few iterations, because the evaluation function - the skeleton decomposition - is computationally much more complex than the operations of the GA itself. A restriction in the size of the population and in the evolution time is necessary to assure a convergence to the optimal structuring function in a short time. There are two consequences to be drawn for the design of the reproductive plan: we must be sure to keep the best population member, because if it was lost it could take a long time to find it again, and we must produce as many new children as possible at each iteration to provide as many new (and better) structures as possible. A reproductive plan that fits these conditions is the one presented in reference [8] and which is called *simplex reproduction*.

The algorithm works as follows: the population has to be of odd size. Choose the best member of the old population and put it into the new population without changing it. Then perform a local competition two–by–two of the members selected at random in the old population and keep the member of each competition which has the better fitness value. Between those winners, the crossover operator (Section 3) is applied to give $\frac{N-1}{2}$ children. The children are inserted in the new population at all odd member sites, except at the

Proceedings of the 1995 International Conference on Image Processing (ICIP '95)
0-8186-7310-9/95 $10.00 © 1995 IEEE

first site which is already occupied by the best member of the old population. Therefore, it remains to fill all even sites, which is done by mutations (Section 3) of the best chromosome.

3. SHAPE ORIENTED GENETIC OPERATORS

The function of genetic operators is to cause chromosomes created during reproduction to differ form those of their parents. The operators must be able to create configurations of genes that were never existent before.

The most commonly used genetic operators are mutation and crossover.

Mutation acts by randomly changing the alleles of a gene. If a certain probability μ_M is passed for a gene, any of the possible alleles is chosen at random for that gene. It provides the only way to recover an allele that has been lost in a population.

The crossover operator allows the exchange of genetic material between two parent chromosomes, resulting in a combination of beneficial material of two structures in their offspring. We will study in detail the one–point crossover operator which swaps the genes of two parent chromosomes after a randomly selected point.

An important characteristic of GAs is that their operators manipulate chromosomes without knowledge of the decoded world. This is also the case in nature, however, the natural genetic manipulators have run through an evolution process themselves. Among the infinity of possible chromosome alterations, only a subset of them is used. If we wish to increase the performance for a specific problem, the operator should be manually optimized for this problem.

For this purpose, let us consider all possible positions of crossover–points in a one–point crossover operator and study the induced modifications in the structuring functions $\delta_1(x)$. In Fig. 2, a structuring function $\delta_1(x)$, its chromosome representation and all possible crossover–cuts are illustrated. As can be observed, it is not possible to cut a structuring function vertically! Indeed, there are many situations which can not occur when using a traditional one–point crossover as shown in Fig. 3.

Pixel positions in the representation of $\delta_1(x)$ that are spatially close are likely to describe significant shape features and they should not be separated easily. In order to take into account the configurations of Fig. 3, the chromosome representation is defined on a matrix rather than on a bit–string. The one–point crossover is defined as follows: a point x_1 is chosen at random on one of the four matrix borders. A second point x_2 is chosen in the same way but is not allowed on the same side as x_1. The line which links them defines the crossover point.

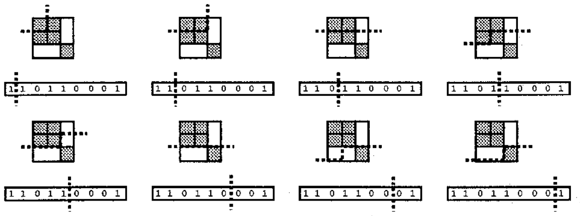

Figure 2: Illustration of all possible cuts by a traditional one–point crossover operator.

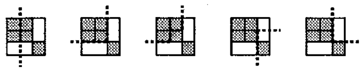

Figure 3: Illustration of the cuts which are not possible with a traditional one–point crossover.

The shape oriented crossover operator provides faster convergence than the traditional crossover operator, as shown in Fig. 4. Moreover, it is capable to convergence to better fit structures in the same amount of time.

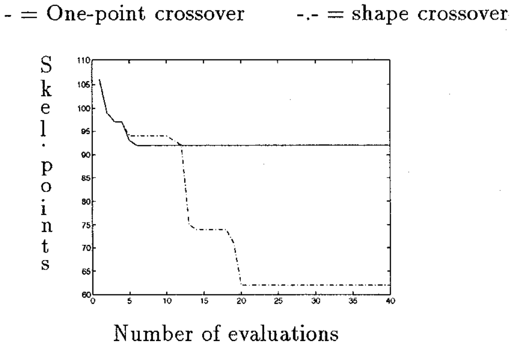

Figure 4: Illustration of the convergence of a traditional one–point crossover and a shape–oriented one–point crossover (less skeleton points = better fit).

4. APPLICATIONS

4.1. Coding

In a previous paper [4], we have already shown the utility of the morphological skeleton for coding of segmented images, based on a geodesic approach. The skeleton decomposition was based on square structuring functions in the digital lattice. The material presented in this paper allows to extend the work to arbitrary shaped structuring functions. The improvements in terms of bit requirements are remarkable and illustrated in

Proceedings of the 1995 International Conference on Image Processing (ICIP '95)

Table 1 for a segmented image. Furthermore, the arbitrary shaped structuring functions allow an improved progressive contour transmission: the skeleton points with the largest associated "radius" already contain some shape information in their reconstructed representation, which is not the case when a square structuring function is used for the decomposition.

Method	Skel points	Bits
Normal skeleton	1867	13832
Optimized skeleton	1172	11272

Table 1: Comparison between optimized and non-optimized skeleton decomposition

4.2. Pattern recognition

Let us suppose that we want to find a feature of a binary image for classification purposes. The detected feature will be compared to a library of templates to classify the binary image. We will show two such applications, an artificially generated and a real world example.

In the artificial example, a binary image was constructed by repeating an elementary, arbitrary shape of maximal size 5 × 5 at different spatial positions, magnifying it by different factors and then taking the union (Fig. 5). The feature was repeated once and magnified by a factor of 10 to produce the shape of Fig. 5, which has been correctly found by the GA (Fig. 6).

Another example of a real application is shown in Fig. 4.2. Different features are computed for different numbers and they could be subsequently used in a classificatin algorithm.

Figure 5: Object to be analyzed

Figure 6: Feature of object of Fig. 5

Figure 7: Examples of shape features.

5. CONCLUSIONS

In this paper, we have presented a solution for the computation of arbitrary shaped structuring functions in the skeleton decomposition. The skeleton is a geometrical shape representation, and the arbitrary shaped structuring functions can be considered as shape features. These features are useful in image coding, as they allow a compaction of the region boundary into fewer skeleton points. Furthermore, a few skeleton features already provide an approximate region description which can be advantageously used in a progressive coding scheme. A second application consists of shape feature extraction for image description and analysis.

The optimization of the structuring functions is guided by genetic algorithms. They provide a robust and efficient search in a complex search space. The introduction of a shape-oriented crossover operator allows a faster and better convergence for the particular problem of skeleton decomposition.

6. REFERENCES

[1] ISO/IEC DIS 13818-2. Information Technology – Generic Coding of Moving Pictures and Associated Audio Information – Part 2: Video. Technical report, International Organization for Standardization, 1994.

[2] H. Blum. Biological shape and visual science (part i). *J. theor. Biol.*, 38:205–287, 1973.

[3] P. Brigger, S. Ayer, and M. Kunt. Morphological shape representation of segmented images based on temporally modeled motion vectors. In *Proc. of the IEEE International Conference on Image Processing*, volume III, pages 756–760, Austin, Texas, November 1994.

[4] P. Brigger, F. Meyer, and M. Kunt. The geodesic morphological skeleton and its fast reconstruction. In J. Serra and P. Soille, editors, *ISMM'94, Mathematical Morphology and its applications to signal processing II*, pages 133–140. Kluwer Academic Publishers, Fontainebleau, France, September 1994.

[5] L. Davis. *Handbook of Genetic Algorithms.* Van Nostrand Reinhold, 1991.

[6] M. Eden and M. Kocher. On the performance of a contour coding algorithm in the context of image coding. Part I: Contour segment coding. *Signal Processing*, 8:381–386, 1985.

[7] J.H. Holland. *Adaptation in natural and artificial systems.* The University of Michigan Press, 1975.

[8] Ch. L. Karr. Air-injected hydrocyclone optimization via genetic algorithm. In L. Davis, editor, *Handbook of genetic algorithms*, chapter 16, pages 223–236. Van Nostrand Reinhold, 1991.

[9] M. Kunt, A. Ikonomopoulos, and M. Kocher. Second-generation image coding techniques. *Proc. IEEE*, 73(4):549–574, April 1985.

[10] J. Serra. *Image analysis and Mathematical Morphology*, volume 1. Academic Press, 1982.

[11] J. Serra. Skeleton decompositions. In SPIE, editor, *Image Algebra and Morphological Image Processing III*, volume 1769, pages 376–386, 1992.