SIM 92/23

COST 211-ter
Simulation Subgroup
Ankara 4-5 February 1992

Source: U.C.L, Belgium
Title: Shape coding with an optimized morphological region description

Abstract

This document describes an algorithm for the shape coding of objects. A region is approximated by an increasing family of subsets included in the region. Each approximation is obtained by adding a subset to the previous approximation. The shape description results in a list of (x, y, B) where (x, y) are the locations in the plane and B is the subset. For coding purposes, we transform this description in order to take advantage of the progressive filling of the plane. An efficiency comparison between our shape description and a classical run-length contour coding is also provided.

1 Introduction

1.1 A complete object coding scheme

Many techniques based on information theory have been developed for the coding of pictures. The most efficient ones use a D.C.T. or a subband transform. As an alternative to these techniques, Kunt [3] has introduced the idea of describing a picture in terms of objects. Each object is characterized by its borders (contours) and its aspect (texture); these entities seem to be more natural in that they coincide with psychological concepts of vision.
The first need in an object coding approach consists in detecting the objects contained in the picture. This operation is called *segmentation*. Figure 1 is the block diagram of a complete object oriented coding scheme. After the segmentation step, contours and textures are coded, transmitted and decoded in the receiver.
Finding a good code for contours and textures is difficult because interactions exist between these two entities. In fact, they are not totally independent. We propose to solve the question in injecting the contour information during the coding and the decoding of textures, hoping to suppress redundancy in this way. As a result, the receiver has to process the decoded contour information for interpreting the texture code correctly.

1.2 Our concern: the coding and the decoding of contours

This document is devoted to the coding and the decoding of contours.

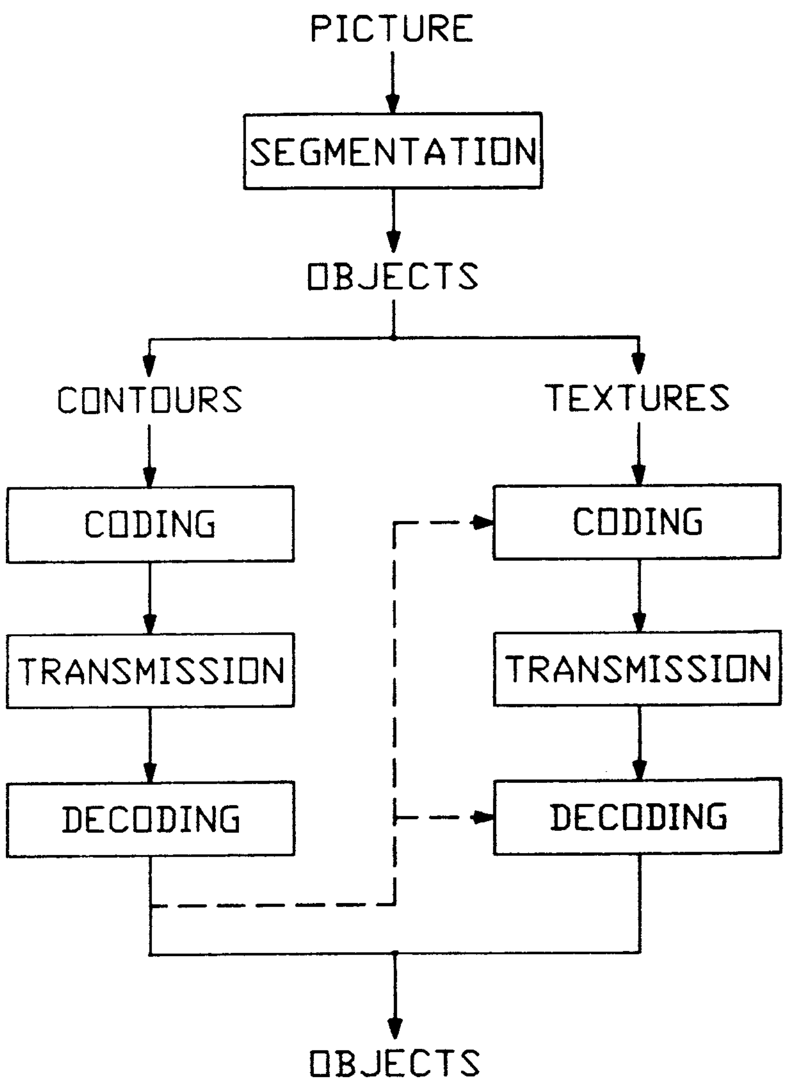

Figure 1: Block diagram of a complete object oriented coding scheme

A precise shape description deserves many applications; it is used in pattern recognition, motion compensation, computer graphics, picture coding, image understanding, etc. As a matter of fact, literature covers a plethora of shape descriptions.
The class of methods following the contour points are very popular. Pitas [4] called this category of shape descriptions "the external representation category". The principal drawback is the inherent difficulty to introduce scale parameters because the data are one-dimensional.
There exists also a second class of contour representations: the "internal shape representation". This time all the efforts are concentrated on coding the surface of each region, whose borders form the contours.
Pitas described a morphological surface representation with a unique geometrical shape of different sizes. Ronse and Macq [5] extended this idea in allowing geometrical overlapping over regions already coded. In this way, a choice is made between regions so to code the easiest region at every moment. Unfortunately, the lack in experiments does not permit to conclude about the efficiency.
Our method completes this last research. We describe a global strategy for the coding of contours. Section 2 is devoted to the description of the coding algorithm. Section 3 compares

our method to a classical run-length technique in a simple case.

2 Basic ideas

This first requirement for a consistent coding description is the synthesis of a region ***representation.*** All the computer transformations are done on this representation; it acts as a complete shape description and should include the significant details. The representation is directly followed by a *reconstruction* operation, which combines the information contained in the representation to reconstruct the region in its original context. Finally, the reconstruction leads to an *approximation.*

Mathematically speaking, if we write X for a region, the representation is a concept $\rho(X)$ which is not necessary in the same domain as X. The reconstruction $\tau(X)$ has $\rho(X)$ as input and the result is an image in the domain of X. This is very similar to applying a direct and an inverse transform, because this case is a particular region description. The approximation $\pi(X)$ results form the cascading of ρ and τ: $\pi(X) = \tau.\rho(X)$. For example, let X be a square of size c, whose location is (a, b) in the plane. A possible representation regroups the four corner positions $\rho(X) = \{(a,b),(a+c,b),(a,b+c),(a+c,b+c)\}$. The reconstruction interpret this data in order to obtain the original square. The primal representation could be more synthesized. It is easily replaced by $\rho(X) = \{[(a,b),c]\}$ containing the upper left position and the square size. We proceed in a similar way, beginning with a simple and naive object representation, and transform it gradually into a more synthesized representation.

- *Morphological representation of a unique region*
  The underlying idea in the construction of a shape description is straightforward: the region is covered with a predefined finite family of templates in an optimal sense. Considering the region as a set of points, we use the theory of mathematical morphology (see [2] for an introduction) to complete the modelization. In mathematical morphology, a set called ***structuring element*** is translated through the plane and its interactions with the region under study serves the result.

  Without entering the difficult language of mathematical morphology, let us consider a family $\mathcal{B}$ of m subsets: $\mathcal{B} = \{B(1), B(2), ..., B(m)\}$. The representation we are looking for is formed by $B(j)$ subsets translated through the plane by p –this set is noted $B_p(j)$–, and included in the region X to represent; the subsets may overlap each other but may not be included in another one. In conclusion, the representation $\rho(X)$ of a unique region is composed of positions and indications concerning the attached subsets. For example, $\rho(X) = \{[j,(a,b)] \mid 1 \le j \le m, (a,b) \textit{ is in the plane and } B_{(a,b)}(j) \subseteq X\}$. The corresponding approximation $\pi(X)$ is $\{B_{(a,b)}(j) \mid ([j,(a,b)] \in \rho(X)\}$. The reconstruction can be made perfect if the subset family contains the single pixel subset.

- *Representation of several regions*
  The procedure for X is used for the representation of n regions $X_1, X_2, ..., X_n$, region after region. In order to compress the representation, we permit the overlap of regions previously coded. The receiver is able to decode the shape information and to detect the overlap because there is an order in the representation.

The above exposed developments constitute the basis for the construction of an interesting shape description. However for efficiency, we need to transform the representation and to enrich its content. This is done by an ordering of regions to be coded, a position entropy reduction and a graph model for the regions. We discuss these three points now.

- *Performance criteria*
  During the representation construction process, the subset we add to the previous approximation is the one which has the most border points in common with the region border not already considered in the representation.

  The task is more difficult for ordering the regions. In a sense, the better region requires the lower number of subsets. Certainly this implies that small regions are preferred to large regions. However in practical situations, it is often indicated to transmit large regions first, from coarse to fine. Moreover the coding of a large region reduces significantly the further possible positions for subset translations and, the more the uncertainty decreases, the more the code is compact.

  The performance criteria giving the best results in our experiments is a contour point to number of subset used ratio. In this way, a choice is possible between small and large regions.

- *Progressive filling of the plane and associated position entropy reduction*
  As we remark previously, during the representation construction, the insertion of a triplet $[j,(a,b)]$ in the representation reduces the number of positions for further subset translations. For example, if Y is the region already represented and $B(j)$ is a structuring element, then all the translations $B_{(a,b)}(j)$ that give $B_{(a,b)}(j) \subseteq Y$ are irrelevant.

  In order to exploit the progressive filling of the plane, the possible positions associated with a given structuring element are transformed in a linear equivalent, starting the counting form left to right and from top to bottom. Furthermore, for each region we regroup the linear positions depending of a same structuring element. The resulting representation looks like $\{[B(1),k_1,k_2,...],[B(2),l_1,l_2,...],...\}$. The position entropy reduction is maximum if the linear positions are such that k_u precedes k_v when $u < v$.

- *Graph model for regions*
  The objects of a picture have a spatial organization based on a neighbourhood relationship. The representation conserves this organization. A simple mathematical model, a graph in fact, renders perfectly the spatial arrangement.

  The spatial arrangement is represented by means of a graph, where the vertices denote the region; two vertices are joined by an edge if the region are neighbours in the picture.

  The graph is adapted during the construction of the representation: when a region is coded, the edges ending at the corresponding vertex are suppressed in the graph model. The interesting point to exploit is the creation of subgraphs or isolated regions. The consideration lying behind is that borders are shared by two regions. Both regions are able to code the commom border points. This is why isolated vertices or, more generally, subgraphs appear in the model. As a consequence, the subgraphs are treated separately and this contributes to reduce the number of possible positions for the structuring elements to smaller spatial zones of the picture.

In conclusion, the graph model translates the spatial organization of regions in an useful form, leading to a substantial win when subgraphs appear. In counterpart, we need some bits, only a small amount, to indicate if each subgraph is composed of different regions or if it is a single region.

3 Result

The coding algorithm was computed for the simple picture drown in figure 2 (128x128 pixels). The seven structuring elements used are squares of size 32, 16, 8, 4, 2, 1 and a rhombus of diagonal length equal to 7.

We limit ourselves to an entropy estimation, where probabilities are estimated by means of experimental frequencies.

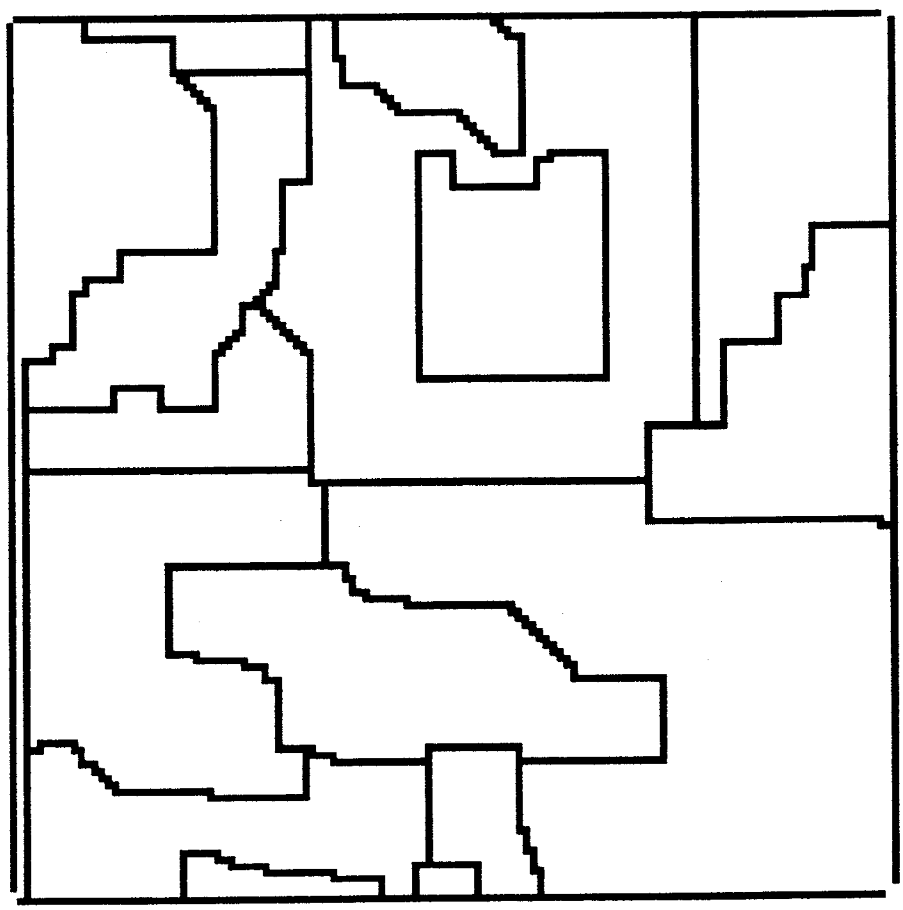

Figure 2: A simple set of shapes for testing the shape representation.

The total cost regroups the entropic costs of the structuring elements and the linear positions, plus the information for the graph model of regions. The result is 0.107 bit per pixel. For comparison, contours coded with a run-length technique –we used Eden's method [1], i.e. 1.5 bit for each border point plus starting points– lead to a total cost of 0.113 bit per pixel.

4 Conclusion

In this document, we develop a complete shape coding technique. The representation is a collection of subsets contained in the different regions. We added the possibility of overlap over regions already coded, a transformation in a linear equivalent for all the positions and a graph model of the regions for efficiency. In a simple case, the result is comparable to a classical run-length technique. For complicated pictures, this is actually not the case.

References

[1] M. Eden, M. Kocher, *On the performance of a contour coding algorithm in the context of image coding*, Signal Processing **8**, 1985, 381-386.

[2] R. Haralick, S. Sternberg, X. Zhuang, *Image analysis using mathematical morphology*, IEEE Trans. on Pattern Analysis and Machine Intelligence **9**, 1987, 532-550.

[3] M. Kunt, M. Bénard, R. Leonardi, *Recents results in high-compression image coding*, IEEE Trans. on Circuits and Systems **34**, 1987, 1306-1336.

[4] I. Pitas, A. Venetsanopoulos, *Morphological shape decomposition*, IEEE Trans. on Pattern Analysis and Machine Intelligence, 1990, 38-45.

[5] C. Ronse, B. Macq, *Morphological shape and region description*, Signal Processing **25**, 1991, 91-105.