SIGNAL PROCESSING VII: Theories and Applications
M. Holt, C. Cowan, P. Grant, W. Sandham (Eds.)
© 1994 European Association for Signal Processing

Feature-Based Video Coding using Mathematical Morphology *

Josep R. CASAS, Luis TORRES

Department of Signal Theory and Communications, Universitat Politècnica de Catalunya
PO Box 30.002, 08071 Barcelona, Spain
Tel/Fax: +34 3 401 6458 / 401 6447
E-Mail: josep@tsc.upc.es

Abstract. This paper puts forward a new approach to second generation image coding. The concept of "image feature" is introduced in order to deal with those "objects" that cannot be properly described as regions within a segmentation framework. Visual features such as open contours or texture details are extracted from the original images using morphological operators. As mathematical morphology deals with the shapes and structures of the images, the resulting features are very close to what would be obtained from a visual perception point of view. If these features are efficiently coded with suitable techniques, such coding model is able to reach higher compression ratios than purely segmentation-based techniques. Numerical results are given at the end of the paper to prove the goodness of the feature-based coding scheme in a very-low bit-rate video coding application.

1. Introduction

A major objective of image coding is to represent an image with as few bits as possible, while preserving the level of quality and intelligibility required for the given application. Although the standardization process in image compression is currently very well defined for a variety of applications –as in JPEG, H.261, and MPEG standards– the emergency of new visual communication systems and the constant need for bandwidth savings indicate that there is a need for more efficient compression schemes that should provide very high compression rates while keeping image quality.

On the other hand, a major breakthrough in image coding should rely on the use of object-based techniques and not on the classical pixel-based approach. As an example, in the context of the future MPEG 4 standard, it is generally accepted that it will be based in techniques relying on the concept of objects. Such techniques make use of second generation image models, which are devised to match the human visual perception. This allows to exploit not only the spatial-temporal correlation of adjacent pixels but also the properties of the observer's visual system.

Segmentation-based coding techniques, for instance, perform this perceptual approach by modeling the image as a set of homogeneous regions that are supposed to be perceived as objects by the observer. The regions are coded individually as homogeneous distributions of texture together with the discontinuities in between.

This paper proposes a new approach to second generation image coding. The concept of "image feature" is introduced to deal with those objects that cannot be properly described as regions within a segmentation framework. Features such as open contours or texture details bring new possibilities for the coding scheme to reach higher compression ratios than purely segmentation-based techniques. Moreover, the coding of visual features contributes towards the perception-based image models in the direction of what have been proposed for second generation techniques [1].

In section 2, after reviewing the basic principles of segmentation-based coding techniques, the need for the introduction of image features is discussed. Different visual features are proposed in section 3 as possible targets for image coding. Section 4 puts forward mathematical morphology as a powerful set of image processing tools for addressing the problem of feature extraction from video signals. Finally, the results of two experiments on feature extraction and coding are presented in section 5 followed by some conclusions in section 6.

2. Segmentation-based Image Coding Techniques

In segmentation-based image coding techniques the image to be compressed is first segmented. The pixels are divided into mutually exclusive spatial regions based on some homogeneity criteria. After segmentation, the image consists of regions separated by contours. Then, the visual information is coded describing the shapes and the interiors of the regions. This description forms the object-based representation of the image.

In most segmentation-based compression schemes, the shape of the regions is represented by encoding the segment boundaries. The interiors of the regions are represented by encoding, for example, the coefficients in

* This work has been partially supported within the RACE MORPHECO Project of the European Union.

the polynomial models describing each region or, for flat regions, the average grey level of the pixels inside the region. Segmentation-based compression methods reported in the literature achieve compression rates in the range of 20–70 for still images, depending on the image quality.

A dual representation, such as contour/texture, is very useful for very low bit rate video coding applications. However, the contour/texture approach presents some limitations. Objects can be found in an image that do not correspond to the basic idea of *region* employed in segmentation-based schemes. There can be very significant contours that are not necessarily closed. The fact of considering them as closed contours, either makes the coding inefficient or forces the introduction of false contours. To illustrate this point, figure 1 shows an open contour in a synthetic image (on the left). Similar features are met in natural images, as the shadows in Lenna's face (fig. 1, right). Another point where segmentation-based schemes use to fail is in the extraction of small regions. When the number of pixels of the area to be considered as a region –homogeneous and different from the neighbors– decreases, independently of the homogeneity criteria being used, this criteria and the segmentation itself become less reliable. A large number of meaningless small regions may be found that are too expensive to code.

Furthermore, for moderate compression applications, segmentation-based approaches are not so useful, since most of the visual quality relies on the texture information (see JPEG for example). On the other hand, for extremely high compression, contours are just too expensive to code. In order to have very efficient coding schemes, more general and less rigid representations than the contour/texture should be introduced. To improve second-generation image coding techniques, the concept of object-based coding approach should change into a more flexible approach. Let us call it a feature-based coding approach.

3. Contour Features and Texture Features

In the context of segmentation-based image coding, a contour is ultimately the position of a *closed transition* between regions whereas texture is a *homogeneous distribution* of grey level or color information for all the pixels of the region independently of the spatial position. These definitions may be relaxed to introduce more general representations of the visual information.

3.1 Contour Features

Transitions or edge information are often seen as the most important feature of images from the human perception point of view [2]. A good approximation to an original image can be made from a sketch image –containing only the strongest edges– where the smooth areas between transitions are obtained from the transitions themselves. Coding schemes based on sketch components have been reported in the literature [3, 4].

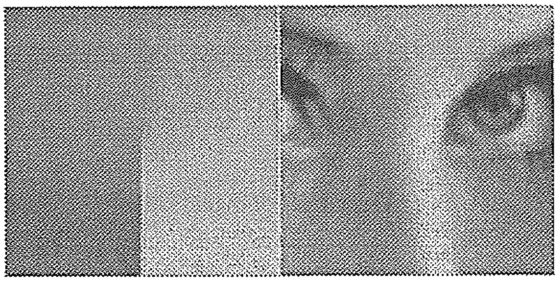

Figure 1. Open contour in a synthetic image (left) and similar features in a natural one (right).

The edge information coded in these schemes presents two main differences compared to segmentation-based schemes:

- Not only the spatial position of the transition is coded but also the grey level or color gap it involves. This approach leads to a feature carrying both spatial and amplitude information. Moreover, such features are coded with suitable techniques that may be adaptive to the local importance of the transition.
- The constraint of closed curve is removed. Edges representing important features of the signal are coded without the restriction (and the resulting cost) of being closed.

3.2 Texture Features

In segmentation-based coding schemes, all pixels of a given region are of the same nature and equally important. Texture coding techniques could benefit from the introduction of several types of pixels within the texture of a region:

- One can extract some "significant" isolated points or isolated lines or network of lines, to add position information in texture codes. This leads to the extraction of "wire mesh" models where the information is split between the line network and its facets.
- One important class of texture features that must be taken into account are visual details [5]. Details are small image components, usually showing high contrast level, which are significant for the visual system. Their significance is often related to the background where they are placed. For instance, for the same contrast level a detail will be more meaningful if it is placed in a smooth background than in a highly textured area or near to an important transition. Even for high compression ratios, details must be considered to match their perceptual significance. The use of a complex perceptual criteria in the extraction and coding of these features may overcome the problem of segmenting small regions discussed in section 2.

3.3 Merging Texture and Contours in a Feature-based Image Coding Model

Using the described features, complex representations of video signals may be built where position and amplitude information are jointly considered for the coding step.

Both texture and contour features lead to a three component image model for coding purposes; the components of this model being the strong transitions/ smooth areas component, the details component and the soft texture component. Such a model is similar to those presented in [3] (two components) and [4] (three components), but the techniques that will be presented in next section for the decomposition are quite different.

4. Morphological Tools for Feature Extraction and Coding

In order to address the extraction of significant features from digital images, a technique strongly related to the physical image structure is required. Such a technique should deal with the "shapes" contained in the video signal for a better matching of the visual perception process. Mathematical morphology [6] provides tools that give a good insight into the structure of the images. Morphological filters perform geometrical analysis and derive quantitative measures from this point of view. Useful morphological operators for this application are the opening and closing by reconstruction for image simplification, the half morphological gradient and the top-hat transform for extraction of significant edges and detail features, and a morphological interpolation process in order to obtain the smooth areas from strong edge information.

As the detail coding algorithm was reported in [5,7], we will concentrate on the extraction of strong edges and on the interpolation process. Figure 2 illustrates the first technique. The original image on the right of figure 1 is first simplified with alternate open-close by reconstruction of increasing sizes to remove small noisy components. The half morphological gradient (dilation–original and original–erosion) is computed to assign to each pixel the grey level difference to the smallest (resp. largest) of its neighbors. In order to mark significant connected components in the gradient images, the same operator used in [5] for detail extraction based on the top-hat transform is applied to give the result shown in the bottom images. The bottom right image shows two lines for the upper and lower brims of each strong edge obtained from the disjoint points of these two images.

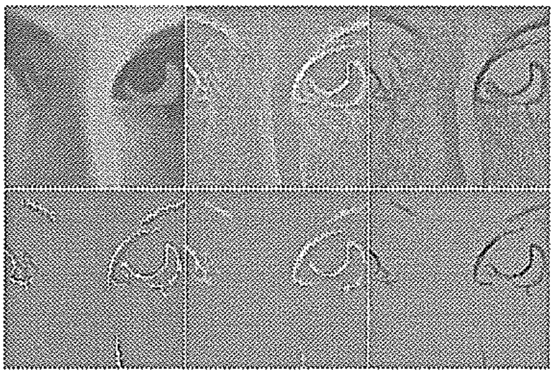

Figure 2. Morphological strong edge extraction. Clockwise: simplification, half gradients (dilation/erosion), feature selection (with top-hat) and upper/lower brims.

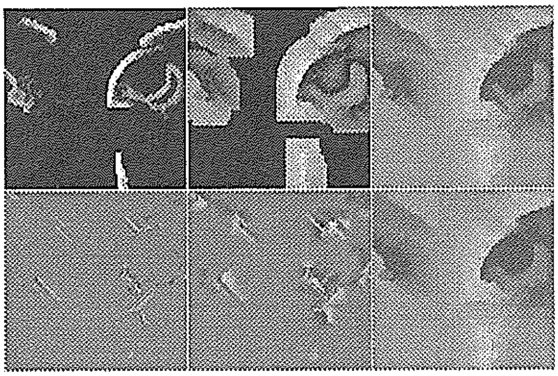

Figure 3. Morphological interpolation from strong edges First row: steps first, tenth and last of the interpolation. Second row: false contours, correction image, result.

If we use information about image amplitudes on both sides of the transitions, the reconstruction of the smooth image becomes a problem of spatial interpolation with the constraint that the reconstructed image should obviously be smooth between contours [3]. To perform the interpolation from the strong edges, the amplitude levels of the edge brims are propagated with exponential attenuation based on the geodesic distance. The first row of images in figure 3 shows the progression of the process. Sometimes, new edges are formed in the pixels where two propagation floods merge. The wrong edges may be extracted on the reconstructed image with the same operator that was applied to the original. The interpolation process is then performed to obtain a correction image from the wrong edges. When the correction image is subtracted from the previous reconstruction the wrong transitions disappear.

The extracted features are coded using suitable coding techniques. Chain code is used for the encoding of the transition locations whereas the amplitude values on the lower and upper brims may be DPCM coded or by means of polynomial approximations –as done in [3]. 3D run-length coding performs efficiently to encode the detail component in video sequences [7]. Finally, for moderate compression, the smooth texture component –computed as the coding error of the reconstructed image produced by the two first components– can be coded using pixel oriented techniques such as VQ or DCT.

5. Results

The original images of figure 1 have been coded twice: first with a segmentation-based algorithm using second order polynomials for the textures and then with the feature-based approach using only the strong edge component. As shown in figure 4, the original image features have been more accurately represented by means of open edge features than using regions with closed contours. The segmentation algorithm gave three regions for the first image and eight regions for the second one with respectively 309 and 1062 contour points, whereas the number of pixels coded for the strong edge component was 136 and 802 respectively.

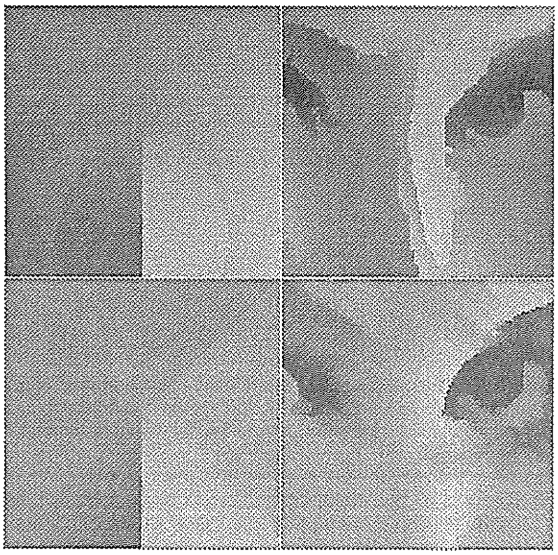

Figure 4. Segmentation-based (first row) versus feature-based coding (second row) for the images of figure 1.

Figure 5 presents the application of feature coding in a very low bit-rate video coding application. As before, the original image has been simplified with a morphological filter and the strong edges are computed. The result of the morphological interpolation is presented in the center row. The number of pixels coded for the strong edge brims are 1556, with 31 upper brim contour pieces and 33 lower brim ones. Using the coding scheme proposed in [3] for the strong edges, with 1.3 bits per contour pixel and 9+18 bits per contour brim, about 3750 bits are needed for this component.

The details' component extracted from the original image with the morphological technique described in [5] is shown in the lower row (left). In this image, there are 9 bright details and 6 dark ones, and 1232 bits were necessary to encode this component. It is worthwhile to observe the subjective quality improvement that for a low bit-rate represents the inclusion of the details. The inter-frame coding of edge features using motion compensation is still under study, but the bit-rate for intra-frame coding at 8 Hz frame-rate is 40 Kbit/s with acceptable image quality (last image of fig. 5).

6. Conclusions

A morphological-based technique for the extraction of strong edges and image details from original images has been presented. Morphological operators are also useful for an efficient interpolation from the strong edge component that yields a good approximation of the image. These features may be coded very efficiently in order to reach high compression ratios. An interesting property is that the extraction operators may be tuned to select only the most significant ones with a result very close to the visual criteria. The perceptual significance of visual features from a subjective point of view must be taken into account for second-generation video coding techniques. The aim is to find a proper matching of the coded features in the image model to the visual features in real images, by relaxing some restrictions in segmented image models. This approach may lead to a better performance in both visual rendition and bit-rate of object-oriented video coding techniques, when a compromise has to be found between bit-rate and visual quality of the encoded image.

Acknowledgements

The authors would like to thank P. Salembier for the helpful suggestions which greatly contributed to the origin and development of this work.

References

[1] M.Kunt, A.Ikonomopulos, M.Kocher. Second generation image coding techniques. *Proc. IEEE, vol 73(4), 549-574,* April 1985

[2] T.Cornsweet.Visual perception.*Academic Press* 1970

[3] S.Carlsson. Sketch based coding of grey level images. *Signal Processing 15, 57-83,* 1988

[4] X.Ran, N.Farvardin. Adaptive DCT coding based on a three-component image model. *Proc. ICASSP'92, vol III, 201-204,* San Francisco, March 1992.

[5] J.R.Casas, L.Torres.Coding of details in very-low bit-rate video. *IEEE Trans. CAS Video Tech. Special issue on very-low bit-rate coding*, June 1994.

[6] J.Serra. Image analysis and Mathematical Morphology' vol II: Theoretical advances.*Academic Press,* 1988.

[7] J.R.Casas, L.Torres. Residual image coding using Mathematical Morphology. *Proc. ICASSP'94, vol V, 553-556*, Adelaide, April 1994.

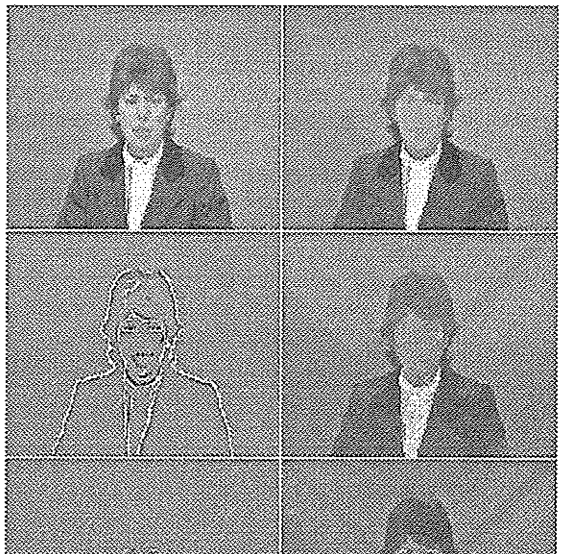

Figure 5. Feature coding for very low bit rate application 1st row, original and filtered; 2nd row , strong edges and strong edge component; 3rd row: coded details and result