AN EFFICIENT TECHNIQUE OF TEXTURE REPRESENTATION IN SEGMENTATION-BASED IMAGE CODING SCHEMES

L. Torres   J. R. Casas   J. Campins

Dept. of Signal Theory and Communications
Universitat Politècnica de Catalunya
Gran Capitán s/n Barcelona, SPAIN
e.mail: torres@tsc.upc.es

ABSTRACT

In segmentation-based image coding techniques the image to be compressed is first segmented. Then, the information is coded describing the shape and the interior of the regions. A new method to encode the texture obtained in segmentation-based coding schemes is presented in this paper. The approach combines 2-D linear prediction and stochastic vector quantization. To encode a texture, a linear predictor is computed first. Next, a codebook following the prediction error model is generated and the prediction error is encoded with VQ. In the decoder, the error image is decoded first and then filtered as a whole, using the prediction filter. Hence, correlation between pixels is not lost from one block to another and a good reproduction quality can be achieved.

1. INTRODUCTION

In segmentation-based image compression techniques the image to be compressed is first segmented [1]. The pixels are divided into mutually exclusive spatial regions based on some criteria. After segmentation, the image consists of regions separated by contours. The interior of the regions are represented by encoding, for example, the coefficients in the polynomial models describing each region, or for flat regions, the average gray level of the pixels in each region. These coding schemes are currently being introduced in the coding of video sequences with very good results [2, 3].

Almost all the effort in inside coding (texture coding) has been devoted to orthogonal polynomial coding or transform coding [4], what has to some extent limited the improvement of these schemes. The quality of the encoded textures is in some cases acceptable, but there is a need to develop more powerful schemes that may provide better visual quality results still giving very high compression.

The objective of this paper is to present new texture coding schemes that greatly improve the visual quality of the compressed image in a segmentation-based scheme using a Vector Quantization approach.

If the segmentation process has been well defined and the obtained regions are homogeneous, then it is possible to design a specific codebook suited to the statistics of each region. The approach is to design the codebook according to some previously defined prediction error model for the regions of the image found in the segmentation process. That is, to generate the codebook according to the model obtained for each region and not according to some specific data sequence as it is usually done in non-segmented images. The generation of the codebook for segmented regions is then both conceptually and computationally very simple.

The basics of our method were presented in [5] where a new technique, called Stochastic Vector Quantization Linear Prediction (SVQLP), for encoding homogeneous textures was introduced. Some previous approaches were also presented in [6] and [7]. Similar techniques were published independently in [8] for normal images.

The present paper discusses the integration of the approach in a segmentation-based coding scheme for still images and presents the basic advantages and drawbacks of the system.

3. STOCHASTIC VECTOR QUANTIZED LINEAR PREDICTION (SVQLP)

For completeness purposes, a very short review of the texture coding technique presented in [5] is in order here. For more details, please refer to the specified reference.

Given a monochrome digital image $u[x, y]$, a *linear predictor* $\hat{u}$ can be defined by

$$\hat{u}[x,y] = \sum_i a_{\alpha_i \beta_i} u[x-\alpha_i, y-\beta_i] \qquad (1)$$

The *prediction error* $e[x, y]$ is the difference between $u[x, y]$ and its prediction:

$$e[x,y] = u[x,y] - \hat{u}[x,y]$$
$$= u[x,y] - \sum_i a_{\alpha_i\beta_i} u[x-\alpha_i, y-\beta_i] \quad (2)$$

Eq. (2) defines a system with input $u[x, y]$ and output $e[x, y]$, called the *prediction error filter (PEF)*. The inverse system has input $e[x, y]$ and output $u[x, y]$ and is called the *prediction filter (PF)*:

$$u[x,y] = \hat{u}[x,y] + e[x,y]$$
$$= \sum_i a_{\alpha_i\beta_i} u[x-\alpha_i, y-\beta_i] + e[x,y] \quad (3)$$

Please notice that this equation represents the well known autoregressive (AR) model.

We are interested in the stochastic coding of the prediction error using a vector quantizer. It must be kept in mind, however, that our ultimate objective is not to encode the prediction error itself, but the original image $u[x, y]$ instead.

Our objective is to find out the predictor required to get a small prediction error and to efficiently encode such error. To that end, rather than to generate a codebook following the image model, a codebook following the prediction error model is generated and it is the prediction error that is encoded using VQ. In the decoder, the error image is decoded first and then filtered as a whole, using the PF.

The prediction error is coded using a modified VQ scheme. The blocks of the error image are *not* coded in such a way that the distortion in reproducing them is minimized, but rather the distortion in reproducing the original image is minimized, taking into account the effect of surrounding blocks after filtering. The objective of the procedure outlined above is to reproduce, at the encoder, the decoding process, so that the codewords can be chosen taking into account the way they will be used. The generation of the codebooks is depicted in Figure 1. Figure 2 shows the structure of the prediction error codec.
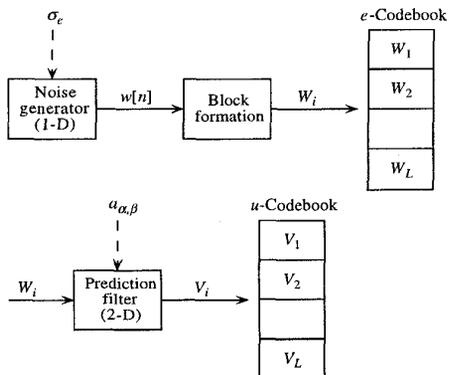
Figure 1. Codebook generation

The noise samples have Gaussian first-order PDF with the same variance as $e[x, y]$ and are mutually
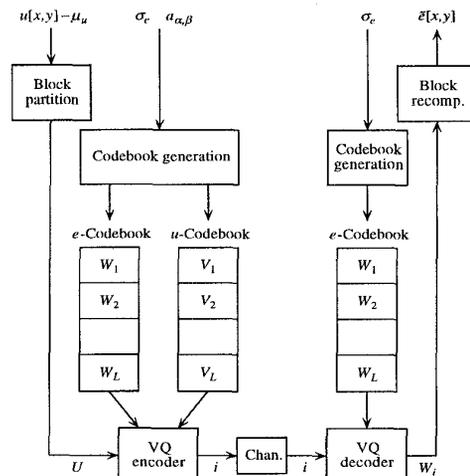
Figure 2. Codec of $e[x, y]$.

incorrelated After filtering with the PF, the codewords V_i have statistics (locally) "similar" to those of $u[x, y]$. The decoder does not need them since the whole prediction error is globally filtered after decoding it. The only parameters that need to be encoded are the mean of the region, the filter coefficients and the prediction error variance σ^2_e.

4. SEGMENTATION-BASED IMAGE CODING

Next step is the integration of the SVQLP approach above presented into a segmentation-based image coding scheme. The segmentation scheme corresponds mainly to the intraframe part of the video algorithm presented in [2]. Originally the texture in that algorithm was coded using orthogonal polynomials in the way proposed in [4]. What is proposed now is the integration of the SVQLP in the intraframe part of the overall scheme. As the performance of the SVQLP highly relies on the segmentation technique used, a brief summary of the scheme is in order here for completeness purposes. A more complete description can be found in [2].

The segmentation algorithm relies on tools based on Mathematical Morphology. In particular, the intraframe segmentation relies on the hierarchical method proposed for still images in [9]. The algorithm first produces a simplified segmentation in the sense that it involves a reduced number of regions. Then, the segmentation is progressively improved by introducing new regions. Each segmentation level involves four basic steps: modeling, simplification, marker extraction and decision.

Modelling: after the first segmentation, a estimation of the partition is known. Then, the next hierarchical segmentation level will have to improve this estimation

dealing with the regions that cannot be well represented by the coding process. To have information about these regions, each region is actually coded. Then, the difference between the coded image and the original one, called the modelling residue, is computed. Since this residue concentrates all the information about the poorly coded regions, it is used as the signal to be segmented in the following steps.

Simplification: The images are simplified to make them easier to segment. The simplification controls the nature and amount of information that is kept for segmentation at this level of the hierarchy. Size and constrast have proven to be useful as visually important criteria. Using a contrast criterion means that regions of high contrast are extracted. Highly textured regions are the most difficult regions to code in segmentation-based coding schemes. It is important to notice at this point, that the SVQLP technique has proven to be useful especially for this kind of regions. Morphological h-maxima or h-minima operators are particularly suitable for contrast simplification [2].

Marker extraction: the goal of this step is to detect the presence of homogeneous regions. It produces markers identifying the interior of the regions that will be segmented.

Decision: The process of assigning pixels to a region is known a decision. The morphological watershed algorithm is used as decision process.

5. SVQLP AND SEGMENTED REGIONS

The application of the SVQLP to segmented images arises some important questions. The regions to be coded are divided in blocks in the same way that is done in conventional VQ schemes. It can be noticed that an important number of blocks overlap the contours of the regions, thus implying that the block has pixels of the same region and pixels from neighbour regions. This is the case for instance of blocks 1 and 2 in figure 3. In order to obtain the parameters of the SVQLP, the pixels belonging to neighbour regions are not taken into account. In this case the pixels outside the particular region are set to zero. Some other problems need to be solved before we can apply the SVQLP scheme to the segmented regions.

Let us now assume that the image has been segmented in two regions as shown in figure 3. One region is the shadowed one (foreground) and the other the background. In this case, for example, blocks 1 and 2 belong to more than one region what implies that these blocks are coded twice, or expressed in a different way, the area covered by all the coded blocks is bigger than the area of the original segmented image. On the other hand it is also evident in this example that blocks 1 and 2 have more pixels belonging to the background region than to the foreground regions what can lead to a waste of bits and coding efficiency when both blocks are used to

code the foreground region. To solve these problems we have applied simultaneously two different solutions: the first is to set a threshold in such a way that if the number of pixels of the region is below the threshold, then the region is not coded with SVQLP but with the mean of the region. In our work this threshold has been set to 100 pixels. Secondly only the blocks that represent the majority of the pixels are coded. This means that only blocks that have more than 10% of the total number of pixels are coded.
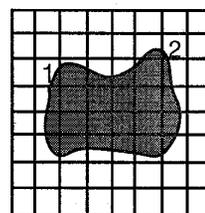

Figure 3. Block savings in SVQLP

Another important point is to know when the SVQLP scheme can be applied. That is, to detect the regions where the SVQLP will have a good coding performance when compared to other schemes. If the obtained region is spatially flat or presents smooth variations it may happen that other coding techniques are more adequate. In our case we have applied a very simple rule which has proven to be effective to know which regions have to be coded with the SVQLP. Once the prediction error of the segmented regions has been obtained the regions are divided in blocks and then the median of the variances of each block is computed. A threshold of 4.5 has been set to select the regions that will be coded with SVQLP. The scheme based in orthogonal polynomials presented in [4] is applied to the regions with a value smaller than 4.5.

6. RESULTS AND CONCLUSIONS

The SVQLP scheme has been applied to a variety of images with similar results. We have selected the well known 256 x 256 image Camman to present our scheme. This image is composed of a variety of smooth regions as well as highly textured regions. The segmentation scheme presented in [2] has been used in intraframe mode and the contour coding has been implemented using the lossles chain code technique presented in [10] with an average of 1.34 bits/contour pixel.

Image 1 presents the original image. Image 2 shows the encoded image at 0.21 bits/pixel. The total number of regions is 83 from which 32 are coded with SVQLP. The SVQLP has been applied using 64 codevectors of size 6x6 and a first order predictor. The mean value of each region has been coded with 6 bits while the predictor coefficients have been coded with 3 bits. Notice that the

SVQLP presents a good performance especially in highly textured homogeneous regions. It is also worth mentioning that the quality of the segmentation greatly affects the quality of the coded regions. In the case of inhomogeneous regions the systems shows its limitations. As the number of bits assigned to code the contours is high, it is clear that to achieve higher compression ratios the number of regions have to decrease. But on the other hand if the size of the regions is high, the regions will loose homogeneity what degrades the performance of the SVQLP. There is a trade-off between the compression ratio and the quality of the regions encoded with SVQLP.

The scheme is now being applied to the encoding of the prediction error in video sequences. As the basic philosophy of the SVQLP is the coding of the original image through the prediction error, for the case of video sequences what is used is the prediction error of the prediction error of the original sequence. As the prediction error of the original sequence is not homogeneous, the SVQLP in its actual stage is not adequate for this purpose. We have made some conceptual modifications to the basic scheme that seem very promising.

As a summary we may conclude that an efficient technique for encoding homogeneous regions in a segmentation-based coding scheme has been presented. If high compression ratios want to be achieved, the scheme has to be combined with other texture coding approaches.

References

[1] M. Kunt, A. Ikonomopoulos and M. Kocher, "Second Generation Image Coding Techniques", Proceedings of the IEEE, vol. 73, pp. 549-574, April 1985.

[2] P. Salembier, L.Torres, F. Meyer, C.Gu, "Region-Based Video Coding Using Mathematical Morphology", Proceedings of the IEEE, Special Issue on Digital TV, June 1995.

[3] H. Musmann, M. Hötter, J. Ostermann " Object-oriented Analysis-Synthesis coding of moving images", Image Communication, Vol.1, No. 2, pp. 117 - 132, October 1989.

[4] M. Gilge, T. Engelhardt and R. Mehlan, "Coding of Arbitrarily Shaped Image Segments Based on a Generalized Orthogonal Transform", Image Communication, Vol.1, No. 2, pp. 153 - 180, October 1989.

[5] D. Gimeno, L. Torres, J.R. Casas, "A New Approach to Texture Coding Using Stochastic Vector Quantization", International Conference on Image Processing, Vol. 1, pp. 119 - 123, Austin, Texas, USA, November 1994.

[6] L. Torres, E. Arias, "Stochastic Vector Quantization of Images", International Conference on Acoustics, Speech and Signal Processing, Vol. III, pp. III-385 - III-388, San Francisco, USA, March 24-27, 1992.

[7] L. Torres, J.R. Casas, S. de Diego, "Segmentation-based Coding of Textures Using Stochastic Vector Quantization", International Conference on Acoustics, Speech and Signal Processing, Adelaide, Australia, Vol. 5, pp. V-597 - V-600, April 19-22, 1994.

[8] S. Gupta, A. Gersho, "Image Vector Quantization with Block Adaptive Scalar Prediction", SPIE Visual Communications and Image Processing 1991, Vol. 1605, pp. 179 - 189.

[9] P. Salembier and J. Serra, Morphological Multiscale Segmentation of Images, Proc. of SPIE VCIP'92, Boston, MA, pp., 1992.

[10] F. Marqués, J. Sauleda, A. Gasull, "Shape and Location Coding for Contour Images" Proc. of the Picture Coding Symposium 1993, Lausanne, March, 1993.

Image 1. Original image Camman

Image 2. Segmented image Camman 0.21 bpp