

PREDICTION ERROR IMAGE CODING USING A MODIFIED STOCHASTIC VECTOR QUANTIZATION SCHEME

Luis Torres, Montse Mora and Josep R. Casas

Dept. of Signal Theory and Communications
Universitat Politècnica de Catalunya
Gran Capità, 08034 Barcelona, Spain
Tel: (343) 401 64 49, Fax: (343) 401 64 47
E-mail: luis@gps.tsc.upc.es

ABSTRACT

The objective of this paper is to provide an efficient and yet simple method to encode the prediction error image of video sequences, based on a Stochastic Vector Quantization (SVQ) approach that has been modified to cope with the intrinsic decorrelated nature of the prediction error image of video signals. In the SVQ scheme, the codewords are generated by stochastic techniques instead of being generated by a training set representative of the expected input image as is normal use in VQ. The performance of the scheme is shown for the particular case of segmentation-based video coding although the technique can be also applied to motion-compensated hybrid coding schemes

1. INTRODUCTION

A well established practice in video coding is to use a motion-compensation hybrid coding scheme. In these schemes a predicted frame is found by motion estimation techniques and the prediction error image is usually obtained finding the difference between the original frame and the motion-compensated frame. The resulting prediction error image is encoded and sent to the receiver along with the motion information. The prediction error image is usually subjected to a two-dimensional intraframe transform coding operation, where the Discrete Cosine Transform is normally applied. This approach is used in all current video standards such as MPEG 1, MPEG 2, H.261 and the new H.263 for very low bit rate video applications [1, 2, 3, 4]. A hybrid approach has also been proposed in the coding of video sequences using segmentation-based coding schemes [5]. For more details on segmentation-based video coding the reader is referred to [6]. In this last case what is compensated are the regions obtained in a segmentation process instead of the square blocks in which the image is divided in the current video standards.

The objective of the intraframe transform applied to the prediction error image is to decorrelate and further compress the motion-compensated frame difference obtained in the temporal domain. If the motion estimation/compensation scheme has been well designed, then the obtained prediction error image is practically uncorrelated and behaves as noise-like in nature, with a flat spectrum, for which further

processing is irrelevant as far as image compression is concerned [7]. On the other hand is very well known that the efficiency of transform coding relies on the fact of having input data which is correlated which seems to be contradictory when used to encode the prediction error image. As an example, let us mention that the Discrete Cosine Transform and the Discrete Walsh-Hadamard Transform perform the same for prediction error images, what proves the high inefficiency of these schemes for these type of images [8].

The objective of this paper is to provide an alternative and efficient method to encode the prediction error image, based on a Stochastic Vector Quantization (SVQ) approach [9, 10] that has been modified to cope with the intrinsic decorrelated nature of the prediction error image of video signals. In the SVQ approach, the codewords are generated by stochastic techniques instead of being generated by a training set representative of the expected input image as is normal use in VQ [11]. The performance of the scheme is shown for the particular case of segmentation-based video coding [5].

The organization of the paper is as follows: Section 2 presents a basic study on the characteristics of the prediction error image, Section 3 introduces the modified Stochastic Vector Quantization used in the coding of the prediction error image, Section 4 presents some results on the coding of prediction error images in the context of segmentation-based schemes and finally Section 5 is devoted to conclusions.

2. CHARACTERISTICS OF THE PREDICTION ERROR IMAGE

The efficient encoding of the prediction error image relies on a good understanding of the characteristics of this type of image. To that end, we have performed a series of studies to better know the behaviour of the prediction error image. This type of image is mainly composed of an homogeneous background (zero prediction error) and many small contours of different gray levels. These small contours are due to errors in the motion estimation process and also due to the appearance or disappearance of objects in the image. From a mathematical point of view, the gray values can be represented by a very narrow Gaussian-like distribution centered in 0. These many different small contours make the image highly uncorrelated. Previous works have drawn the same conclusions [12]. In particular, for highly active sequences the horizontal and vertical first order autocorrelation coefficients are in the range of 0.49 and 0.21 respectively, meanwhile

This work was supported by the European Union through the MORPHECO project of the RACE program and the Spanish government through the grant TIC95-1022-C05-05

for the original image the same coefficients are in the range of 0.97 and 0.93 [8].

Our own studies show that the prediction error image can be divided in three types of important pixel information:

- The majority of the pixels have a value near or equal to zero presenting a uniform distribution
- Groups of similar pixel values quite different from zero spread out on the image
- Isolated pixels placed randomly in the image with pixel values very different from zero

Some very important questions arise naturally: 1) which of these pixels give the most important perceptual information and 2) how do they have to be encoded? The first group of pixels represents areas that have been well motion-compensated and do not have to be further processed. It seems clear that pixels belonging to the second and probably the third group are the most important as they represent the *weakness* of the motion estimation stage. A very important property shared by the second and third group of pixels is their position. In addition, we have noticed that the most important pixels of the prediction error image are grouped in *clusters* of different shapes. Our results show that it is crucial, for the prediction error image, to encode the position of the corresponding pixels and to maintain the shape of the clusters very accurately in addition to encode their luminance value.

A very normal approach is to encode the prediction error image using a transform scheme. But it seems clear that transform schemes are not very appropriate due to the uncorrelated nature of the prediction error image. In addition, it should be kept in mind that transform schemes attempt to code the coefficients value and not the coefficients position. Some alternative attempts to encode the prediction error image with Vector Quantization schemes are promising, although no outstanding results have been provided yet. An early paper on the use of VQ on prediction error images is [13] and a review can be found in [7]. Our experiments to encode prediction error images with Vector Quantization using the LBG algorithm [11] have been quite disappointing even if the original sequence is included in the training sequence. Transform and Vector Quantization schemes share in common the *weakness* that they do not maintain the position of the encoded pixel as both methods rely only on the distribution of the pixel values.

3. CODING OF THE PREDICTION ERROR IMAGE

We have designed a two-step procedure to encode the prediction error image. This two-step procedure takes special care to encode the position of the corresponding pixels. After a prefiltering and simplification process, the first step encodes the position of the most important pixels of the prediction error image using a modified Stochastic Vector Quantization (SVQ) scheme [9, 10] and the second step encodes their pixel value using also a modified Vector Quantization approach. A block diagram of the overall coding process is shown in Figure 1. Each block is described next.

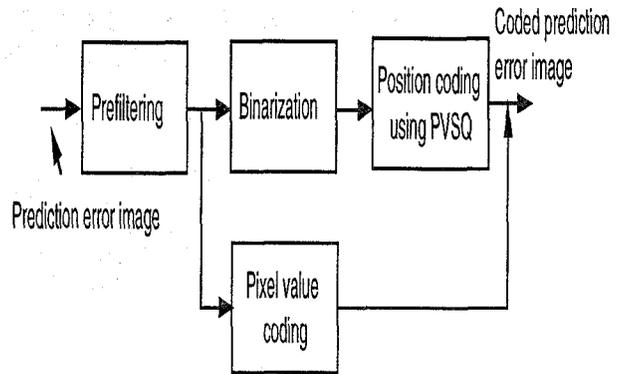


Figure 1: Prediction error image coding

3.1. Prefiltering and simplification

The goal of this step is to prepare the prediction error image to be as *accurate* as possible for the encoding block. This means a decision is made concerning which pixels are kept and which pixels are eliminated of the image. This decision includes two steps: similarity and neighbourhood filtering. Following a similarity criteria, the pixel values between $0-e$ and $0+e$, where e is a predefined threshold, are filtered out and set to zero. The simplified pixels are not considered important in the encoding of the prediction error image and are not coded. When the original prediction error image is compared to the simplified one, no visual difference is noticed. After the simplification stage, a second process of simplification consisting in identifying the most important pixels to be coded is made. Following a neighbourhood criteria, it is decided which are the pixels of each cluster that will be further eliminated. One pixel is considered to belong to a cluster if it is different than zero and is similar enough to one of its eight neighbours. Isolated pixels are removed and are not coded.

3.2. Binarization

In order to code the position of the pixels of the prediction error image, the pixels remaining of the simplification process are first binarized. Each one of the pixels which is different of 0 is assigned a value of 128. All pixels equal to 128 are considered relevant information. This is, the position of these pixels is significant and has to be efficiently encoded. Pixels equal to 0 are not important (the motion compensation has had a good behaviour) and do not have to be exactly coded. The basic objective of this binarization process is to emphasize between the important and not important pixels of the prediction error image.

3.3. Position coding

The coding of the position of the pixels is done using a modified Stochastic Vector Quantization scheme [9, 10]. In conventional Vector Quantization, the codebook is generally designed using the LBG algorithm [11] which uses a large training set of empirical data that is statistically representative of the images to be encoded. In SVQ, a model for the blocks of the image is computed first and then the codewords

are generated according to this model and not according to some specific data sequence. Stochastic Vector Quantization provides an alternative way for the generation of the codebook based in random number generators. Notice that the only difference between conventional VQ and SVQ is the way of generating the codewords. The process of selection of the best codeword and the distortion criterium are exactly the same in both techniques.

The resulting binarized prediction error image is divided in blocks of 8x8 blocks and then coded using the modified SVQ scheme. In the sequel, the new SVQ scheme will be called Position Stochastic Vector Quantization (PSVQ). In the PSVQ scheme, the codewords are pseudo-random generated according to some geometrical characteristics previously calculated from the binarized prediction error image. These geometrical characteristics provide information about the spatial distribution of the clusters of the pixels needed to be coded in the prediction error image. Specifically, the codewords are pseudo-randomly generated according to the amount of 0's and 128's present in the binarized image, according to how many pixels there are in each block, and according to the main spatial orientation of the pixels (North, South, etc). Notice that the codewords of the PSVQ will have only binary values composed only of 0's and 128's and will have a spatial distribution similar to the clusters of the pixels of the prediction error image.

Once the codewords are generated, the binarized prediction error image, that is the position of the clusters of pixels, is encoded using a MSE distortion as in conventional LBG. Some other distortion measures could be devised to improve the technique. The generation of the codebook using PSVQ requires only random number generators, which makes the technique very fast as no training set is required. The same generators have to be available in both the encoder and the decoder.

This technique provides an efficient lossy method for the encoding of the position of the more important pixels of the prediction error image. Some lateral information has to be transmitted in order to rebuild the codewords at the receiver side. An average value of 0.5 bits are needed to encode the position of each pixel. Notice that lossless position coding techniques regularly used such as chain code [14, 15], need a minimum of 1.3 bits and therefore are not adequate to encode the position of pixels of prediction error images.

3.4. Pixel value coding

After the encoding of the position of each pixel, the associated pixel value information has also to be transmitted. Different schemes could be used to that end. We have tried to encode the pixel value using scalar quantizers but the resulting bit stream was not adequate for the range of compression desired. Another possibility is to encode the mean value of each cluster of pixels, but we have not obtained good visual reconstruction quality. Still another option is to use some sort of run-length approach having into account that there are many prediction error pixels whose value is zero or near zero.

As the SVQ has given good results in the encoding of gray value information, and the PSVQ has provided a way to encode the pixel position, why not to combine both approaches?

We have designed a technique in which the prediction error image is divided in different *binary partitions*. The division is done in such a way that the addition of the partitions gives the original image. Each partition has only two types of pixels: one type is always zero and the other corresponds to the gray level value of the prediction error image. For instance, assume that the prediction error image has pixel values comprised between -20 and +20. The first partition will be formed by pixels of values 0, -20, the second by 0, -19, and so on until the last partition which will be formed by 0, +20. Once the partitions are formed, the PSVQ approach is used to encode each partition. Please notice that it is then possible to encode simultaneously the position of the pixels and their gray value.

4. RESULTS

The prediction error image coding approach has been tested on the segmentation-based video coding scheme presented in [5]. In that paper the prediction error image was coded using orthogonal polynomials [16] and good visual results were shown for generic video sequences in the range of 18 - 32 kbits/s. Although our research using the PSVQ scheme is at an early stage, the encoding of the prediction error using this approach at least equals the visual quality obtained in [5] for the same bit-rate, but it is much faster, what proves the practical application of the proposed method. As an example, Figure 2 shows some results. Figure 2.A shows the original image Carphone. Figure 2.B shows the resulting prediction error using the scheme presented in [5]. Figure 2.C presents the prediction error image coded using orthogonal polynomials as proposed in [16]. Figure 2.D shows the reconstructed image using Figure 2.C. Figure 2.E shows the prediction error image coded using the approach presented in the paper and finally Figure 2.F presents the reconstructed image using Figure 2.E. Both reconstructed images have the same bit-rate of 26 Kbits/s and a similar visual quality but the proposed PSVQ approach is much faster.

5. CONCLUSIONS

A new Vector Quantization scheme called Position Stochastic Vector Quantization has been introduced to encode prediction error images. The main features of the method relies on a fast generation of the codebook following some geometric characteristics of the image. Although the approach has been tested on segmentation-based video coding schemes, it can be further extended to encode the prediction error of conventional block-based video coding standard schemes.

6. REFERENCES

- [1] ISO-IEC IS 11172 (MPEG-1). Coding of moving pictures and associated audio for digital storage media up to about 1.5 Mbit/s. Technical report, Motion Picture Experts Group, 1993.
- [2] ISO-IEC DIS 13818 (MPEG-2). Generic coding of moving pictures and associated audio. ITU-T recommen-

ation H.262. Technical report, Motion Picture Experts Group, March 1994.

- [3] ITU-T Recommendation H.261. Video codec for audiovisual services at px64 kbit/s. Technical report, ITU, 1993.
- [4] Draft ITU-T Recommendation H.263. Video coding for narrow telecommunication channels at less than 64 kbit/s. Technical report, ITU, July 1995.
- [5] P. Salembier, L. Torres, F. Meyer, and C. Gu. Region-based video coding using mathematical morphology. *Proceedings of the IEEE*, 83(6):843–857, June 1995.
- [6] L. Torres and M. Kunt. *Video coding: the second generation approach*. Kluwer Academic Publishers, Englewood Cliffs, 1996.
- [7] R. J. Clarke. *Digital compression of still images and video*. Academic Press, 1995.
- [8] A. N. Akansu and R. A. Haddad. Discrete Walsh - Hadamard transform vector quantization for motion-compensated frame difference signal coding. In *International Conference on Acoustics, Speech and Signal Processing*, pages 1862–1865, Glasgow, May 1989.
- [9] L. Torres and E. Arias. Stochastic vector quantization of images. In IEEE, editor, *International Conference on Acoustics, Speech and Signal Processing*, volume III, pages 385–388, San Francisco (CA), USA, March 1992.
- [10] D. Gimeno, L. Torres, and J.R. Casas. A new approach to texture coding using stochastic vector quantization. In IEEE, editor, *Image Processing Conference*, pages 119–123, Austin (TX), USA, November 1994.
- [11] Y. Linde, A. Buzo, and R. M. Gray. An algorithm for vector quantization design. *IEEE Trans. on Communications*, 28(1):84–95, January 1980.
- [12] B. Girod. Motion compensation: visual aspects, accuracy and fundamental limits. In M. I. Sezan and R. L. Lagendijk, editors, *Motion analysis and image sequence processing*, pages 125–152. Kluwer Academic Publishers, 1993.
- [13] M. Bage. Interframe predictive coding of images using hybrid vector quantization. *IEEE Transactions on Communications*, 34(4):411–415, April 1986.
- [14] M. Eden and M. Kocher. On the performance of contour coding algorithm in the context of image coding. Part 1: Contour segment coding. *Signal Processing*, 8(10):381–386, 1985.
- [15] F. Marqués, J. Sauleda, and A. Gasull. Shape and location coding for contour images. In *Picture Coding Symposium*, pages 18.6.1–18.6.2, Lausanne, Switzerland, March 1993.
- [16] M. Gilge, T. Engelhardt, and R. Mehlan. Coding of arbitrarily shaped image segments based on a generalized orthogonal transform. *Signal Processing: Image Communication*, 1(2):153–180, October 1989.

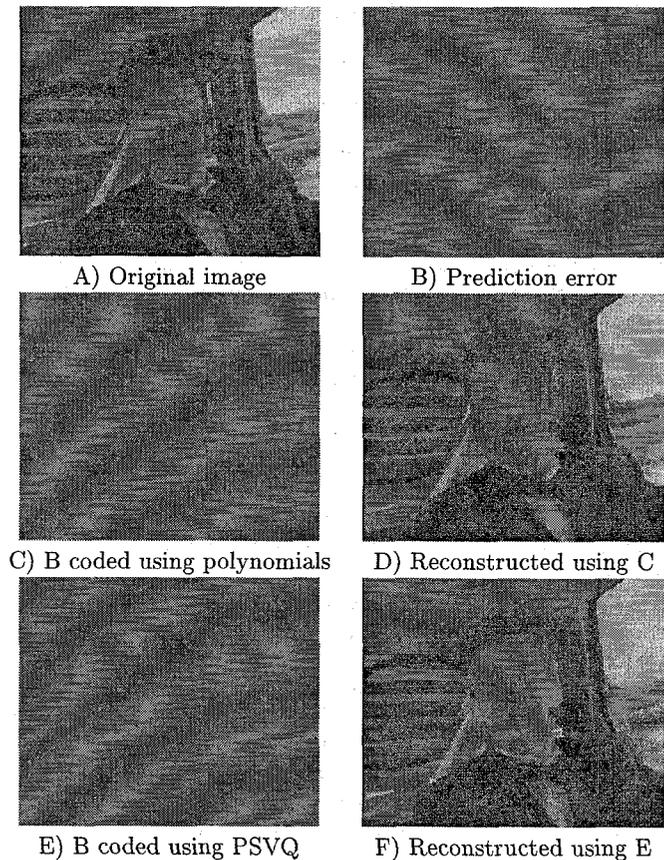


Figure 2: Results