# MOTION COMPENSATED VIDEO COMPRESSION USING ADAPTIVE TRANSFORMATIONS

*Zafer Diab and Paul Cohen*

## ABSTRACT

Block-based motion compensation fails to maintain an acceptable level of prediction error which makes the transmission of this error impossible for very low bit-rate coding owing to the small bit allocation. The reason is that the motion model assumed in block-based techniques cannot approximate the motion in the real world precisely. To develop an effective motion compensation method for very low bit-rate video coding, we address the issue of adopting more sophisticated motion model than block-based. The motion model discussed here is based on the representation of optical flow in its principal components domain. The performance of motion compensation based on this model is compared with MPEG using the PSNR mesure and qualitative experiments. Both of these criterias show a gain of compression in the ordre of 30%.

## 1. INTRODUCTION

Coding of digital image sequences has been the focus of much recent research interest. Applications include TV, HDTV, multimedia and videoconferencing. Considering the present accessibility of very low bit-rate channels that were initially designed to transmit speech or text, the realization of video communication at such rates may bring forth the future popularization of video codecs. Such low bit-rate channels include radio channels, public switched telephone networks (PSTN), and computer networks for transmitting electronic mail, which require bit-rates lower than 64 kb/s.

The main technical issue of very low bit-rate video communication is clearly the video coding method, which is required to accomplish the necessary bit-rate with sufficient image quality. The highest compression ratios are achieved by using motion compensation methods to reduce temporal pixel redundancies inherent in image sequences. Motion compensation requires the decoder to construct an estimate of the motion field and to use it to reconstruct images by interpolation. Various methods of motion compensation are characterized by how information about this motion field is obtained at the decoder.

All approaches to motion compensation estimate a motion field by relating pixel intensities in the previous frame of a sequence to their new locations in the current frame. A basic problem to any approach to motion compensation is that this motion field estimate must be reconstructed at the decoder where current frame pixel intensities are not available. The three standard solutions to this problem are block-based motion compensation, region-based motion compensation and dense motion compensation.

From the viewpoint of implementation, the block-based hybrid method of block matching [6] and discrete cosine transform [11] clearly has the biggest advantage. These block-based motion compensation strategies involve estimating the motion of each image block by a single vector [1]. At very low bit-rates, however, this coarse approximation usually results in visible block effects [7].

Dense motion fields make no assumption on the motion in the real world. They are generally represented by a 2-D vector for each pixel, resulting in twice the number of parameters needed to represent an image. This apparent overparametrization of the motion field limited its exploitation in motion compensation to the use of recursive approaches. These techniques predict the motion field at the current frame based on previously decoded frames [9, 13]. Since the motion field is computed from information already available to the decoder, no motion estimates need to be transmitted over the channel. However, recursive methods are often inaccurate in tracking both motion changes from frame to frame and motion field discontinuities within a frame [10].

In this paper, we propose a more accurate description of the motion based on the dense motion field. Since typical motion fields are far smoother than typical intensity fields [3], they have a bigger compression potential. We subdivide these motion fields into motion blocks and represent each motion block in a transformed domain. The transformation basis is learned

This work was supported by the NSERC-Noranda Chair in Mining Automation

and adapted in a self-organized manner to represent the motion of the scene. This allows us to reconstruct dense motion fields of invisible error, while transmitting the same amount of motion information as standard block-based methods. This has the advantage of limiting reconstruction error while achieving a compact motion description.

The paper begins in section 2 with a descrition of the coding strategy. Experimental results presented in section 3 demonstrate the superiority of our system to existing video coding systems. The paper concludes in section 4 with some final observations and pointers to future work.

## 2. DESCRIPTION OF THE CODING STRATEGY

Figure 2 illustrates the coding strategy. A spatial study on the differences between the incoming image $I(t+1)$ and the previously reconstructed image $\hat{I}(t)$ allows us to divide the image into dynamic and static regions. The optical flow in the dynamic regions is first computed and subdivided into motion blocks. The motion in each dynamic block is then transformed and the transform coefficients are quantized and transmitted to the receptor who reconstructs dense motion by inverse transformation. The reconstructed motion is then used to reconstruct images. The choice of the transformation for each block depends upon the class of the 2D motion present in the block. To avoid error accumulation, intraframe images are transmitted each time the error becomes higher than a threshold.

### 2.1. Detection of static blocks

A block $B_t$ at time $t$ is considered to be static, if the RMS error between this block and the same block of the previous image $B_{t-1}$ is less than a threshold. This block does not need to be coded and is simply replaced at the receptor by $B_{t-1}$.

The main issue here is the choice of the threshold to determine whether a block is static or dynamic. This choice should satisfy both of these restrictions:

1. the threshold should be independant from the noise present in the images;
2. all moving blocks should be considered to be dynamic.

Because of the first restriction, the threshold should be adapted to the images in its input. In other words, it should be a statistical measure on the RMS error of all blocks. At the same time, the threshold should always be higher than a limit $Tr$ because of the second restriction. This limit means that all blocks $B_t$ having $RMS(B_t, B_{t-1}) > Tr$ are considered to be dynamic. The criteria to decide if the block $B_t$ is static or dynamic is then:

$$RMS(B_t, B_{t-1}) > \min(Tr, E[RMS(B_t, B_{t-1})]) \quad (1)$$

### 2.2. Motion classification

A fixed number of motion classes (seven for the module and phase in the actual implementation) is adopted. A principal component transform of the module and phase of the image motion are associated to each class as you can see in the next section. We will note by $T_i^a$ and $T_i^\theta$ the matrix of transformation for the $i^{\text{th}}$ class of the module and phase respectively.

The module and phase of each particular motion block $B$ are associated with the class that is the most correlated with this block as shown in figure 1. The correlation mesure is chosen to be the scalar product between the block and the principal component of the class. We will note the principal component of the matrix $T$ by $T(1)$. The decision criteria for the classification problem is then:

$$a \in i \text{ if } \; T_i^a(1).a > T_j^a(1).a \;\; \forall j \neq i \quad (2)$$

$$\theta \in i \text{ if } \; T_i^\theta(1).\theta > T_j^\theta(1).\theta \;\; \forall j \neq i \quad (3)$$

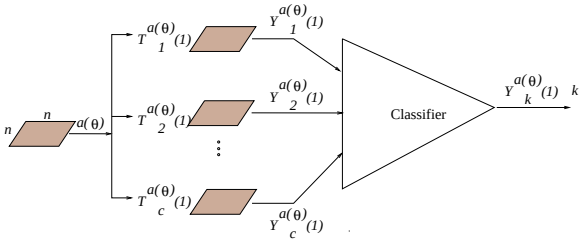



Figure 1: Classification system architecture. Inputs are the module (phase) of the motion blocks. The $c$ principal components ${}_i^a T_1$ (${}_i^\theta T_1$) consist of a basis image of size $n \times n$ and output the principal coefficient ${}_i^a y_1$ (${}_i^\theta T_1$). The principal coefficient to be sent is chosen to be the one having the maximum norm

### 2.3. Motion transformation and quantization

A neural network using the Hebbian learning rule is used to learn the principal components of the motion of each class [4]. The matrix of transformation $T$ is then adapted iteratively to converge towards the principal components of its input. If we note by $x$ and $y$ the input and output of the matrix respectively, the

adaptation law that we use can, then, be described by the equation [12]:

$$\Delta T = \eta(yx^t - LT[yy^t]T) \quad (4)$$

where $LT[.]$ sets all elements above the diagonal of its matrix argument to zero, thereby making it Lower Triangular. Let $\eta(t)$ be such that $\lim_{t\to\infty} \eta(t) = 0$ and $\sum_{t=0}^{\infty} \eta(t) = \infty$. Under these conditions, we can prove [12] that if $T$ is assigned random weights at time zero, then with probability 1, equation 4 will converge and $T$ will approach the principal component transform.

Since the transform coefficients are uncorrelated and ordered in descending order of variances, retaining the first few coefficients provides an adequate approximation of the input motion and achieves good compression. In the actual implementation, we keep only one coefficient for the module and phase. This coefficient is quantized using a uniform quantizer. The upper and lower limits of the quantizer are transmitted to the receptor as an overhead before each image.

The transforms associated to each motion class (and therefore the nature of each class) are adapted during the coding process, in order to reflect the particular structure of the motion present in the current image. Since the adaptation information is costly to transmit, it is established on the basis of previously transmitted and reconstructed motion flows.

### 2.4. Image reconstruction

Owing to the orthonormal nature of the transforms involved, the reconstruction of the image motion flow simply involves multiplication by the transpose of the forward transform matrices.

Image reconstruction is made by linear patch displacement interpolation. This method gives improved results since it assumes smoothness in the velocities rather than in image intensities [8].

Reconstruction errors tend to be mostly located at object boundaries and cause an error accumulation in time. We attenuate this error by transmitting intra-frame images each time the RMS error becomes higher than a threshold. The value of this threshold controls the desired compression. These images are transmitted uniquely for dynamic blocks and are coded by retaining a fixed number of coefficients in the $DCT$ domain.

## 3. EXPERIMENTAL RESULTS

The transform basis adaptation allows us to reconstruct dense optical flows of invisible error by retaining as less as 1 vector per dynamic block. The original and reconstructed optical flow for the "Rubik" sequence are given at figure 3. The mean angle error [2] between the original and the reconstructed flow is of 2°.

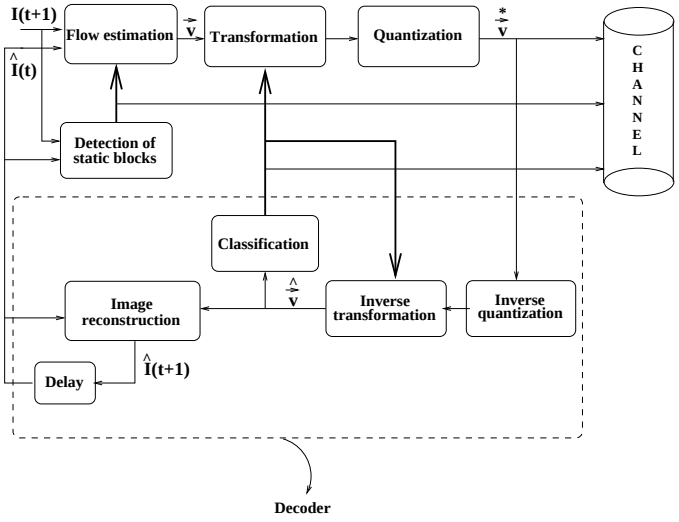



Figure 2: Block diagram of the coding strategy

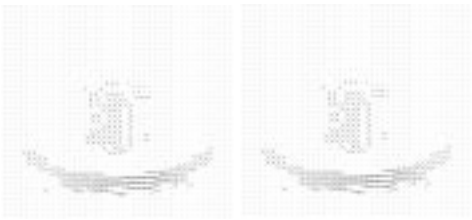
Using dense motion description allows us to work at low bit rates. At figure 4, we show the PSNR in the reconstructed images of the "Rubik" sequence with various compressions using our system compared to MPEG I [5]. We notice that our system with a compression of $79:1$ reconstructs images with less error than MPEG I with a compression of $66:1$ which is equivalant to a compression gain of 20%. Qualitative experiments confirm this result. In fact, 3 random persons preferred the images reconstructed with our system (compression of $79:1$) $4/4$, $1/4$, and $2/4$ of the time respectively. Typical reconstructed images for both systems are presented in figure 5.

Similar tests were conducted on the "Taxi", "Nasa", and "Tunnel" sequences and they showed, respectively, a gain of compression of 45%, 24%, and 42% compared to MPEG I.

## 4. CONCLUSION

In this paper, a new motion compensation scheme is presented. It particularly addresses the problem of dense motion modeling and compression. The motion was modeled in its principal components space which allowed us to reconstruct dense motion fields of invisible error by retaining as less as one vector per dynamic block. Quantitative and qualitative results show a gain of compression of 30% compared to MPEG I. This is adequate for the video transmission at low bit-rates.

Many problems still remain to be solved. Our future



(a) Original optical flow

(b) Reconstructed optical flow

Figure 3: Original and reconstructed optical flow for the "Rubik" sequence. Mean angle error is 2°

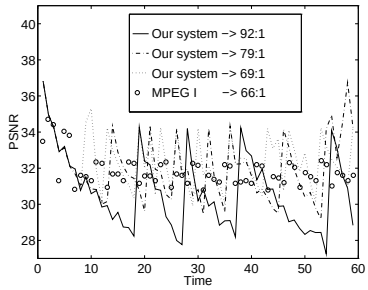



Figure 4: PSNR in the reconstructed images of the "Rubik" sequence with various compressions using our system compared to MPEG I

work is focused on the following subjects:

- use of the transmitted motion to interpolate the colour as well as the luminance;
- use of the position of static blocks to adapt the interpolation in the image reconstruction process.

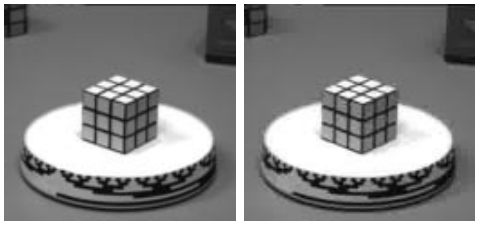
## 5. REFERENCES


[1] V. Bhaskarau and K. Koustantinides. *Image and Video Compression Standards*, volume 2. Kluwer Academic Pub., 1995.

[2] D. J. Fleet and A. D. Jepson. Computation of component image velocity from local phase information. *International Journal of Computer Vision*, 5(1):77–104, 1990.

[3] J.V. Gísladóttir and M.T. Orchard. Motion-only video compression. In *Proceedings of the IEEE International Conference on Image Processing*, volume 3, pages 730–734, 1994.

[4] D. Hebb. *The Organisation of Behaviour.* New York : Wiley, 1949.

(a) Our system (79 : 1)

(b) MPEG I (66 : 1)

Figure 5: Typical reconstructed images with our system and with MPEG I


[5] ISO. Coding of moving pictures and associated audio. *Committee Draft of Standard ISO11172: ISO/MPEG 90/176*, Dec 1990.

[6] J. R. Jain and A. K. Jain. Displacement measurement and its application in interframe image coding. *IEEE Transactions on Communications*, 29(12):1799–1808, 1981.

[7] H. Li, A. Lundmark, and R. Forchheimer. Image sequence coding at very low bitrates: A review. *IEEE Transactions on Image Processing*, 3(5):589–609, 1994.

[8] T. Lin and J. L. Barron. Image recontruction error for optical flow. Technical report, University of Western Ontario, 1994.

[9] A.N. Netravali and J.D. Robbins. Motion compensated television coding – part I. *Bell Systems Technology Journal*, 58:631–670, 1979.

[10] M.T. Orchard. Predictive motion-field segmentation for image sequence coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 3(1):54–70, 1993.

[11] K. R. Rao and Y. Yip. *Discrete Cosine Transform: Algorithms, Advantages, Applications.* New York: Academic, 1990.

[12] T.D. Sanger. Optimal unsupervised learning in a single-layer linear feedforward neural network. In *Neural Networks*, volume 2, pages 459–473, 1989.

[13] D.R. Walker and K.R. Rao. Improved pel-recursive motion compensation. *IEEE Transactions on Communications*, 32(10):1128–1134, 1984.