A MOTION COMPENSATION ALGORITHM BASED ON NON LINEAR GEOMETRIC TRANSFORMATIONS AND QUADTREE DECOMPOSITION

Andrea Cavallaro, Stefano Marsi, Giovanni L. Sicuranza
D.E.E.I. – University of Trieste, Italy
Phone: +39-40-6767147 ⋆ Fax: +39-40-6763460 ⋆ E-mail: marsi@ipl.univ.trieste.it

ABSTRACT

In this paper we present an efficient very low bit rate coder for CIF image sequences at 10 frames/s. The proposed algorithm performs motion compensation using a second order geometric transformation and a quadtree structure. High compression ratios have been reached adapting the features of the transformation and the block size to the motion. A linear transformation is used when there is translational motion, while a non linear transformation is applied when a block has to be warped. In addition the block size is increased in homogeneous zones and reduced in detailed moving zones. Simulation results show that the proposed coding method improves the predicted images quality and reduces the blocking artifacts with respect to traditional hybrid block based techniques.

1 INTRODUCTION

Very low bit rate image sequence coding is very important for video transmission and storage application. There are many practical applications of this kind of coder, ranging from video transmission by narrow band channels, such as conventional telephone lines or mobile telephone networks, to multimedia applications used in data bases and electronic mail. Very low bit rate video coding requires very high compression ratios. Conventional block matching based techniques are unable to maintain an acceptable quality under this coding condition.

The fundamental goal of image sequence coding is to remove spatial and temporal redundancy. Temporal redundancy can be reduced using a motion compensated prediction of the current frame from the previous reconstructed one. Spatial redundancy left in the prediction error is then reduced by a transform coder. For most video coders there are two important drawbacks for motion estimation: the fixed block size and the use of non realistic transformation to find the best matched block in the previous frame. Our aim is to overcome the former with a quadtree structure and the latter with a non-translational motion field.

2 THE PROPOSED ALGORITHM

2.1 Geometric transformations

In the block matching technique the same displacement vector is assigned to all the pixels of each block and therefore the quality of the estimation is conditioned by block size. Large blocks may include pixels whose real displacement is very different from the one represented by the calculated displacement vector. It is possible to overcome this limit by geometrically transforming part of the previous frame using a transformation capable to specify a different motion vector for each pixel within a block. The choice of the transformation depends on the kind of motion we want to compensate with motion estimation: the best tradeoff between computational complexity and estimation properties has to be found. Let us write a generic second order geometric transformation as

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} d_1 & d_2 & d_3 & d_4 & d_5 & d_6 \\ d_7 & d_8 & d_9 & d_{10} & d_{11} & d_{12} \end{pmatrix} \begin{pmatrix} u^2 \\ u \\ uv \\ v^2 \\ v \\ 1 \end{pmatrix} \quad (1)$$

where (u, v) and (x, y) are the spatial coordinates respectively of the current and the previous frame. The more parameters we use, the better is the prediction and the more expensive is the transmission cost. A complex method allows a better description of the motion in a sequence and thus it is able to represent efficiently the motion of the image.

The transformation used by the block matching technique is unable to compensate accurately for changes in the image such as zooming, rotation and uneven stretching. On the other hand it has been shown ([1], [2], [3]) that more complex motion models can provide a significant improvement in the accuracy of the prediction error in the new image. In this paper we present an efficient way to apply the non linear geometric transformation defined in (1), whose degrees of freedom let compensate for more realistic kinds of motion. Our aim is to overcome the problem of computational complexity due to the use of the non linear transformation using a fast technique for parameter calculation and for their application. In order to define the deformation properties of (1), the parameters d_i have to be fixed, through the knowledge of the mapping of a number of pixels (*control points*) related to the number of the parameters. A transformation characterised by $2k$ parameters needs at least k control points to be defined. A complete processing for parameter calculation should perform a global optimisation within each block based on the following stages:

- displacement of *one* control point,
- parameters calculation,
- application of the transformation,
- calculation of prediction error.

These steps are repeated for all the control points positions within the search window. The parameters that lead to the minimum prediction error are selected as the block parameters. The proposed method for the evaluation of the parameters uses a local optimisation greatly faster than the previous one. The stages of this process are:

- determination of the new position of the control points by block matching,
- parameters calculation.

Using this sub-optimal method we have to carefully choose the number and the position of the control points. Experimentally we found that the best tradeoff between computational complexity and prediction accuracy is given by the symmetrical disposition of nine points: the four vertices, the mid points of each side and the centre of the block. Now we have to define the way to calculate the parameters

$$\begin{aligned} \vec{d'} &= (d_1, d_2, \ldots, d_6)^T \\ \vec{d''} &= (d_7, d_8, \ldots, d_{12})^T, \end{aligned} \tag{2}$$

of (1). Let (u_i, v_i) and (x_i, y_i) be the generic control points respectively in the current and in the previous frame. Let their coordinates be collected in the vectors

$$\begin{aligned} \vec{x} &= (x_1, x_2, \ldots, x_9)^T \\ \vec{y} &= (y_1, y_2, \ldots, y_9)^T. \end{aligned} \tag{3}$$

The matrix

$$A = \begin{pmatrix} u_1^2 & u_1 & u_1v_1 & v_1^2 & v_1 & 1 \\ u_2^2 & u_2 & u_2v_2 & v_2^2 & v_2 & 1 \\ \vdots & & \ldots & & & \vdots \\ u_9^2 & u_9 & u_9v_9 & v_9^2 & v_9 & 1 \end{pmatrix}, \tag{4}$$

made up of the control points coordinates values in the current image, is a rectangular matrix. Therefore the solutions of the equations

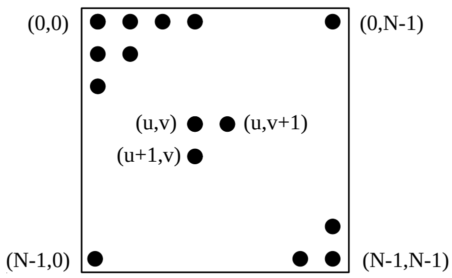

Fig. 1: The regularity of the location of pixels within a block lets calculate the mapping of a pixel starting from the knowledge of the mapping of the *previous* one.

$$\begin{aligned} \vec{x} &= A\vec{d'} \\ \vec{y} &= A\vec{d''} \end{aligned} \tag{5}$$

are the vectors d' and d'' calculated using the generalised inverse of matrix A:

$$\begin{aligned} \vec{d'} &= (A^TA)^{-1}A^T\vec{x} \\ \vec{d''} &= (A^TA)^{-1}A^T\vec{y} \end{aligned} \tag{6}$$

To have a unique solution the matrix A^TA has not to be singular. This hypothesis is satisfied when the columns of matrix A are linearly independent. But they are composed by the control points coordinates, that for their location do not generate linearly dependents columns.

Once we have the parameters we can apply the transformation using an efficient method to reduce the computational complexity of the non linear geometric transformation: the *line scanning technique*. This technique is based on the regularity of the location of pixels within a block (Fig. 1), so that it is possible to calculate the new position of a pixel starting from the value of the previous one:

$$\begin{aligned} F(u, v+1) &= F(u, v) + [d_4 + d_5 + d_3u + 2d_4v] \\ G(u, v+1) &= G(u, v) + [d_{10} + d_{11} + d_9u + 2d_{10}v] \end{aligned}$$

for row pixels, and

$$\begin{aligned} F(u+1, v) &= F(u, v) + [d_1 + d_2 + d_3v + 2d_1u] \\ G(u+1, v) &= G(u, v) + [d_7 + d_8 + d_9u + 2d_7u] \end{aligned}$$

for column pixels of each warped block.

2.2 Quadtree

The classical video and image compression algorithms are based on a previous image division into square blocks of fixed sized and on the independent coding of each block. The choice of the block size constitutes one of the problems of this kind of algorithm. Large-size blocks allow a higher compression degree, but with a degradation of subjective quality of the coded image in highly detailed moving regions. On the other side, small blocks allow a better adaption but with a reduced compression efficiency. All the problems raised by the division of images in blocks of fixed sizes suggest the segmentation of the image in blocks of variable size, therefore images will be divided into large or small size blocks according to the complexity of the region to code. Furthermore in the moving areas we are proposing an algorithm that allows us to use not only variable size blocks but also variable kind of transformation to estimate the motion.

The proposed algorithm has been designed in order to obtain a coder with an high degree of adaptation and an implementation complexity as low as possible. Adaptability is a very important feature in very low bit rate coding, as it allows an efficient management of the limited number of available bits. When the coder is adapted to the spatial and temporal variation of the image features, it is possible to improve the quality of the coded images for a constant channel bit rate with respect to the traditional methods. Quadtree decomposition permits a natural division of the image, allowing a careful treatment of the highly detailed areas. This is achieved by the proposed method using two sizes of blocks and a three stage motion compensation algorithm. With respect to H.263, the increasing of information due to the definition of the quadtree structure is compensated by the use of larger blocks. The proposed method works on two resolution levels: 32×32 and 16×16 size blocks. The peak signal-to-noise ratio (PSNR) of the luminance component is used as a distortion measure.

The first stage at each level concerns the stationarity of the block. If the PSNR is greater than a fixed threshold, then the block has not to be motion compensated. If the block is not stationary, the algorithm looks for the best matched block in the previous frame using the traditional two parameters transformation. If at the first stage the block matching technique is unable to give a satisfactory prediction, the algorithm decomposes the 32×32 block into its 16×16 quadrants and repeats the process to each sub block. If translational compensation does not allow a good prediction at the second level, therefore the block has to be warped with non linear transformation, whose ability to approximate warping an uneven stretching is displayed in Fig. 2. The particular is taken from the sequence "Miss America" and shows the different predictions obtained by linear and non linear transformations when an eye closes: the former is not capable to give a good prediction at all, while the latter succeeds in providing at least a coarse approximation of the real motion. This improvement corresponds to a decreasing of the prediction error.

With the quadtree structure and the three stage compensation method, this structure achieves an high degree of adaptability, without increasing the number of bits required for description due to the use of a fixed subdivision. The limit of the fixed subdivision can be partially overcome without increasing the computational complexity and the overhead bits with a further subdivision that does not need calculation of new motion vectors. If neither the warped block succeed in being over the PSNR threshold, it is subdivided in its 8×8 quadrants and then they are weighted with the corresponding quadrants of the translated block. Therefore we can treat in a different way each quadrant according to its kind of motion. Let A be the translated 16×16 block and B the warped one. We consider the respective quadrants A_i and B_i, so that

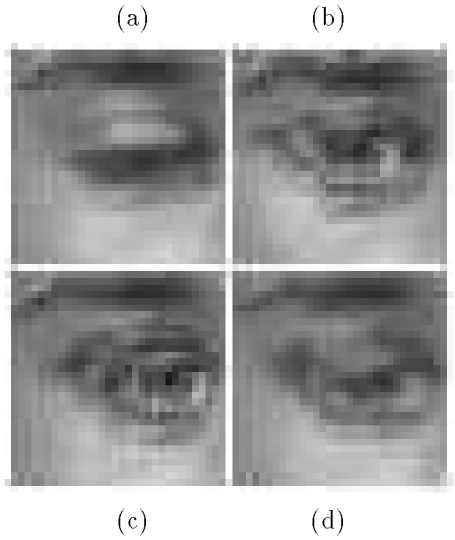

Fig. 2: Comparison between particulars of the predicted image of the test sequence Miss America: (a) original, (b) block matching (32 pixel side block), (c) block matching (16 pixel side block) and (d) proposed method (32 and 16 pixel side block).

$$A = \begin{pmatrix} A_1 & A_2 \\ A_3 & A_4 \end{pmatrix} \qquad and \qquad B = \begin{pmatrix} B_1 & B_2 \\ B_3 & B_4 \end{pmatrix}$$

and we weight the corresponding ones using coefficients α_i and β_i, with $\beta_i = 1 - \alpha_i$ and $\alpha_i \in [0, 1]$. In order to code the weights in an effective way, α_i is chosen with step 0.1, therefore there is a fixed set of possible coefficients that reduces the number of bit required for transmission. One generic 8×8 weighted quadrant is given by

$$C_i = \alpha_i \cdot A_i + \beta_i \cdot B_i$$

and therefore

$$C = \begin{pmatrix} C_1 & C_2 \\ C_3 & C_4 \end{pmatrix}$$

will be the 16×16 generic weighted prediction. The method will choose the block C made up of the quadrants C_i that led to the best PSNR. This strategy together with the non linear motion compensation allows us to improve the prediction as shown in Fig. 3.

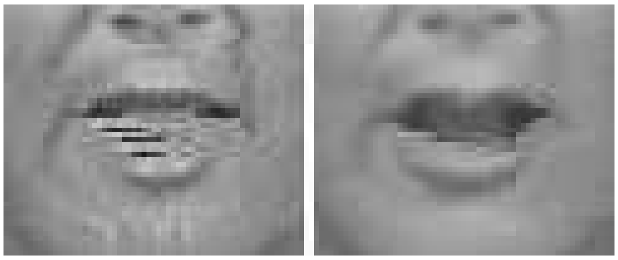

Fig. 3: Comparison between particulars of the predicted image of the test sequence Miss America. Left: H.261/3 based method; right: proposed method.

Once a block is motion compensated, the following information has to be transmitted to the decoder: the motion compensation technique and the block size, the displacement vectors for linear transformation or the non linear transformation parameters and, finally, the weights. For the motion compensation technique and the block size the same code word is used. According to the statistics due to simulation results for each information a Huffman table is used, except for the non linear parameters that are quantised and then transmitted.

3 SIMULATION RESULTS

The coder has been used for simulations with the luminance component of "Miss America", "Mother and Daughter" and "Akiyo" sequences on channels from 16 to 32 kb/s. The original sequences have a spatial resolution of 288×352 (CIF) and a temporal resolution of 30 images per second. Input sequences have been temporally down-sampled at 10 Hz. The system has been set up so as to work at constant bit rate. Quadtree information is followed by the transmission of the prediction error according to the residual bandwidth.

Experimental results are shown in Fig. 4. These plots describe the peak signal-to-noise ratio with respect to the frame number of the original sequence. The results show that the proposed coding method using non linear geometric transformation and quadtree decomposition outperforms the H.261/3 based method (labelled *reference*). There is a significant improvement in the subjective image quality, that can be appreciate in the detailed areas such as the eyes and the mouth of the speakers. This improvement is due especially to the reduction of blocking artifacts and to the more realistic prediction of non translational movement made by the proposed method.

4 CONCLUSIONS

We proposed a motion compensation method that allows to improve the quality of predicted images in a hybrid coding scheme. This technique works on two resolution levels and uses a linear transformation where motion is

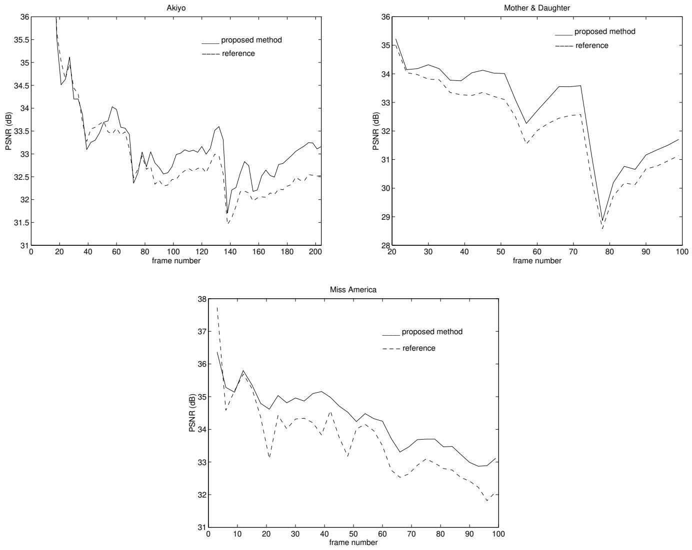

Fig. 4: PSNR of decoded images (CIF). (top left) Akiyo at 16 kb/s, (top right) Mother & Daughter at 32 kb/s, (bottom) Miss America at 32 kb/s.

purely translational and an efficient application of a non linear transformation in detailed moving areas. The result obtained with the warping process, if required, is weighted with the one obtained with the linear transformation. Thanks to this solution it is possible to achieve a remarkable reduction of prediction error and of blocking artifacts than H.261/3 based method.

Large block sizes let reduce the total number of bits and the use of a fast approach for non linear parameters calculation and of the line scanning technique allows us to reduce the computational complexity. Experimental results have shown that the use of non linear geometric transformations directed to the areas of images characterised by complex motion permits to improve the quality of reconstructed frames of a very low bit rate coder.

References

[1] M. Ghanbari, S. de Faria, I. N. Ghon, K. T. Tan, *Motion compensation for very low bit-rate,* Signal processing: image communication, vol. 7, n. 4-6, November 1995.

[2] Y. Nakaya, H. Harashima, *Motion compensation based on spatial tranformations,* IEEE Transactions on circuits and systems for video technology, vol. 4, n.3, June 1994.

[3] C. A. Papadopoulos, T. G. Clarkson, *Motion compensation using second-order geometric transformation,* IEEE Transactions on circuits and sistems for video technology, vol. 5, n. 4, August 1995.