Coding algorithm with region-based motion compensation

Edouard FRANÇOIS, Jean-François VIAL, Bertrand CHUPEAU

THOMSON multimedia R&D FRANCE

Avenue de Belle Fontaine, 35510 Cesson-Sévigné, France

Phone : +33 99 25 42 00

Fax : +33 99 25 43 34

E-mail : vialjf@tcetbs1.thomson.fr

Abstract

This paper describes a region-based coding scheme which has been developed for MPEG-4 tests of November 95. In the coding algorithm, the segmentation into regions is based on an advanced motion analysis, insuring a good temporal coherence of the segmentation and an identification of covered or discovered areas. This approach is therefore well suited for object-based picture representation. Simulation results and MPEG-4 tests of November 95 and January 96 have shown that the algorithm is as efficient as a reference block-based coding scheme derived from MPEG-1, while offering higher frame rates and potential object-based functionalities.

1. Introduction

This paper describes a region-based video coding algorithm which has been developed and submitted to the MPEG-4 tests of November 1995 and complementary tests of January 96. This algorithm consists in a region-based coding scheme, using an advanced motion analysis tool which allows to obtain an accurate motion description. Motion is encoded by partitioning into regions, coherent in time, which can each be described by the parameters of an affine motion model. The accurate motion description allows to achieve good motion compensation for coding topics but also for image format conversion. Moreover, such a region-based description provides with the possibility of content-based functionalities required by MPEG-4 standard. In particular, motion segmentation generally corresponds to object segmentation. Hence it becomes possible to achieve post-production tasks such as keying or mixing. However, for the MPEG-4 tests, this coding algorithm only addressed the compression functionality, mainly because of lack of time.

This paper is organized as follows. Section 2 briefly describes the coding algorithm structure. The two next sections describe the motion analysis, which consists in a dense motion field estimation (section 3), then a motion segmentation (section 4). Section 5 deals with the motion and texture coding. Section 6 presents the intermediate frame interpolation process which allows to reach high frame rates. Section 7 finally discusses about results of this algorithm and provides some conclusions and extensions of this work.

2. General description of the coding scheme

Figure 1 describes the general coder structure. As in MPEG 1 and 2 standards, three types of pictures are defined : intra (I-frame), predicted (P-frame) and bidirectional (B-frame). P-frame *t* is coded with respect to the nearest previous I-frame or P-frame, that is frame *t-K*, after a forward prediction. B-frames from time *t-K+1* to time *t-1* just use both future and path I- or P-frames as a reference.

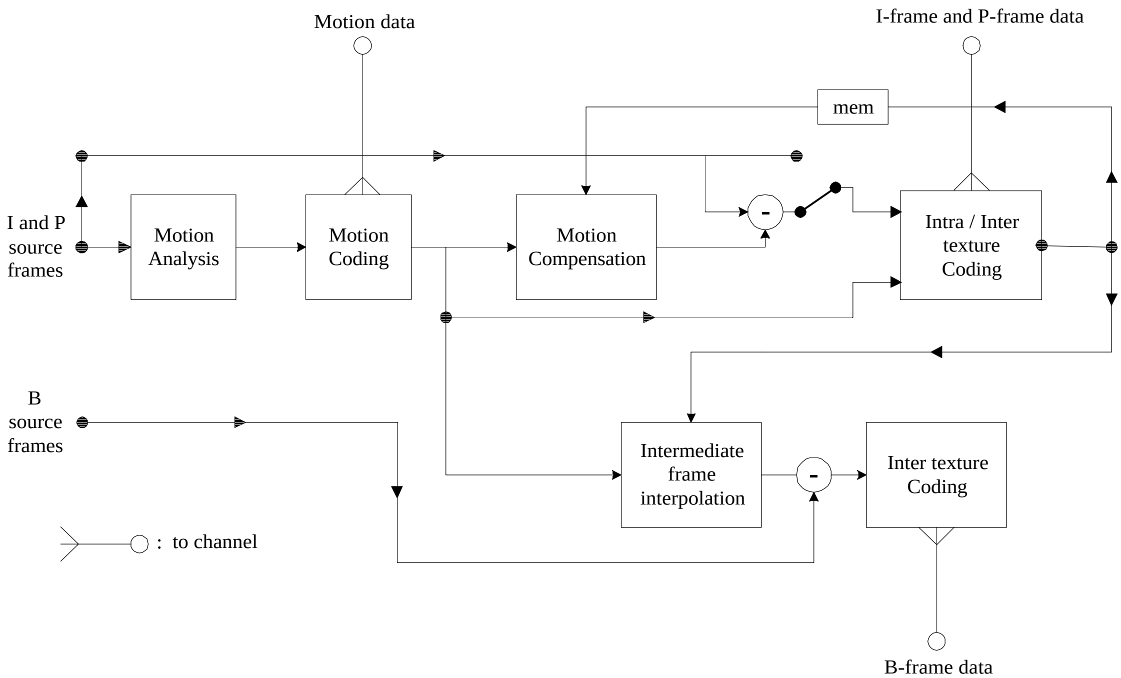

Figure 1 : simplified coding scheme.

- The main block of this scheme is the motion analysis block. It achieves successively motion estimation and motion segmentation at a reduced temporal rate *K*. For each motion region, a linear motion model parameters set describing its apparent velocity field is provided. Successive motion partitions are, as far as possible, linked in time. This block makes possible to detect uncovered areas by projection of the previous dense motion field on the current one; holes of this projection correspond to uncovered areas.
- Motion coding block encodes both motion segmentation and motion parameters set. Motion segmentation is lossless encoded by chain-cod, while motion parameters are encoded after a particular quantization.
- Motion compensation block takes as input data the motion data and the previous I- or P-frame and provides the predicted frame. Texture coding is based on Shape-Adapted DCT [SIK95], and uses segmentation provided by motion segmentation step.
- The last block is the intermediate frames interpolation block. It performs successive B-frames from the two bounding I- or P-frames and motion between these two frames. For each B-

frame, the process is essentially based on a motion temporal interpolation (from current or previous motion data). No additional motion information is used to encode B-frames

3. Dense motion field estimation

Motion analysis works in two steps : first, a dense motion field is estimated, then this motion field is segmented in homogeneous regions with an affine motion model. Motion estimation algorithm is described in [CHU94]. The computation of dense 2D-motion vector field is performed using a multiscale approach based on gradient technique. The luminance images of the input video sequence are firstly decomposed into multiscale low-pass gaussian pyramids. The same estimation process is then repeated at each resolution level, from top to bottom. It can be described as a sequence of four independent processing modules : multi-prediction of the estimates, correction of prediction vectors, computation of associated confidence values and confidence-based smoothing of the vector field.

Confidence measures are computed along with each pixel vector, at each scale, in order to indicate the degree to which it can be trusted. This confidence check is independent from the generation of the displacement field itself. The confidence value in a given direction is in fact proportional to the spatial gradient in that direction and inversely proportional to the Displaced Frame Difference. These confidence measures are used to regularize, at each scale the estimated vector field, by an iterative resolution.

4. Motion segmentation

After the motion estimation step, a motion segmentation is performed from the estimated dense velocity field. As in [BOU93], the segmentation process works at each time t in three main steps:

1 - segmentation map prediction;

2 - detection of new motion regions and estimation of their motion parameters;

3 - optimization of the current segmentation map, first based on the dense velocity field, then based on the displaced frame difference.

The segmentation is based on an affine model representation of the velocity field. To each motion region R is associated a global affine motion parameters vector θ_R which allows to describe its motion field.

Segmentation map prediction

Prediction of current motion segmentation from the previous one is achieved in two steps :

1 - projection in the motion direction of the previous segmentation ;

2 - merging of holes in the projected segmentation.

The previous segmentation projection consists in reconstructing the dense velocity field of time *t-1* from the motion segmentation and parameters of time *t-1* (frame *t-2* to frame *t-1*). Then the label of each pixel of this segmentation is projected in its own motion direction. For pixels where there is a conflict (several vectors point to the same pixel), the chosen label l is the one which minimizes an error function based on the dense velocity field. For each remaining hole region R (regions where no vector points), a merging step is applied. It consists in choosing the neighboring label (that is, adjacent region to R) which minimizes a similar function.

New motion regions detection

This is the basic block of the motion segmentation algorithm. It works in three steps :

1 - identification of motion parameters of each region ;

2 - detection of ill-labeled pixels ;

3 - labeling of connected ill-labeled pixels as new regions.

An efficient way to estimate dominant motion parameters of a given region consists in using a robust estimation method [HOL77], as achieved in [ODO95]. The robust estimation method avoids to take into account erroneous samples, which is useful when a new motion region appears in a previous one.

Then ill-labeled pixels (those for which the current motion parameters do not fit) are detected. A labeling process affects a region number to each ill-labeled area whose size is large enough. The three steps 1-2-3 are reiterated for each new region, until no more region is created.

Optimization of the current segmentation map

A regularization step is applied to the segmentation map in order to adjust the regions frontiers as well as possible and to obtain a clean and quite homogeneous result. This is obtained by minimizing an energy function (derived from a Markovian modelization of the segmentation map and a bayesian criterion). This function contains three terms, a spatial contextual energy which constraints a pixel to have the same label as it neighboring, a second energy which takes into account the observations, that is the velocity vector or the DFD, and a temporal contextual energy which constraints the segmentation to be near the projection in the motion direction of the previous segmentation.

The minimization process is achieved by an ICM (Iterated Conditional Mode) algorithm, [BES86], through a multilevel process derived from [PER94]. The algorithm makes possible to detect uncovered areas by projection of the previous dense motion field (reconstructed from the previous motion segmentation and parameters) on the current one; holes of this projection correspond to uncovered areas.

For the first segmentation map, the algorithm starts with one single region labeled 1. The new motion regions detection then creates the real different motion regions inside this first region. For the following frames, this detection is achieved on the projected segmentation map based on the motion field.

5. Motion and texture coding

Motion coding

Motion coding consists in motion partition encoding and extra parameters and motion parameters encoding. Motion segmentation is lossless intra-encoded by chain-cod technique. First a smoothing process is applied to contours of the segmentation map [GU94], and then closed contours are encoded by chain-cod [MAR93].

The 2D linear model of the motion of each region is represented by three motion vectors $\{(dx_i,dy_i), \text{i=1..3}\}$ appropriately chosen to correctly represent the linear part of the model. The transformation from 2D linear model to these three motion vectors is basically based on the bounding box of the analyzed region. These vectors are then uniformly quantified with a quantization step depending on the desired accuracy of the motion field, and encoded.

Texture coding

Texture coding is region-based and uses Shape-Adaptive DC [SIK95]. Quantization is achieved on a region by region basis. VLC are adaptation of MPEG-2 ones. The quantizer stepsize for each region can be determined from the confidence measurements computed over the region. In this case, as the decoder cannot compute this confidence measure based on successive original frames, it is necessary to encode the quantization step of each region, which can become rather costly. Practically, a simpler quantization function is used : a larger quantizer step is applied to the background region than to the other ones. This avoids to provide supplementary data to the decoder.

6. Intermediate frames interpolation

Intermediate frames interpolation is applied to B-frames. If the current I- or P- frame has been encoded at time *t*, this interpolation takes as input data the two I or P-frames *t-K* and *t*, correspondent encoded motion data and provides the successive B-frames from time *t-K+1* to time *t-1*. It works in two steps :

1 - intermediate segmentation interpolation based on current or previous segmentation ;
2 - intermediate frame interpolation.

For a given B-frame, it tries for each segmentation interpolation a forward and a backward frame interpolation, and chooses the best interpolation (for which the PSNR is maximum).

Intermediate segmentation interpolation consists in projecting the reference segmentation in its motion direction, and then in filling remaining holes by a deterministic relaxation. Once this segmentation is computed, the intermediate frame interpolation is achieved by the same motion compensation process used for the P-frames interpolation.

7. Conclusion

After MPEG-4 tests of November 95 and January 96, it can be concluded that this coding algorithm provides results as good as the MPEG-1 reference algorithm for the compression functionality, while offering an higher frame rate (in this case, 60 Hz, progressive scanning) and potential object-based functionalities.

The advanced motion analysis allows to reach an accurate motion description. Thanks to this motion description, high frame rates can be reached by a specific intermediate frame interpolation process. Moreover it provides successive motion segmentations linked in time which could be used for content-based functionalities such as post-production tasks.

Of course, some parts of the coding algorithm could be improved : contours coding by introducing polygonal or spline approximations and temporal prediction, or quadtree coding, subband texture coding (which could largely reduce the blocking effects of SA-DCT). Finally, an important issue of motion segmentation is to be able to describe the scene in different planes which could each be individually manipulated. This cannot really be done for the while (segmentation just provides different regions, and not planes; hence a region cannot be displaced without remaining a hole at the place where it was before). The concept of plane hence implies a more sophisticated management of covered and uncovered areas in order to reconstruct planes from regions.

References

[BES86] BESAG J., *On the statistical analysis of dirty pictures*, Journal Royal Statistic Society, Vol.48, Serie B, No 3, pp 259-302, 1986.

[BOU93] BOUTHEMY P., FRANCOIS E., *Motion segmentation and qualitative dynamic scene analysis from an image sequence*, Int. Journal of Computer Vision, Vol.10, No 2, pp 157-182, April 1993.

[CHU94] CHUPEAU B., SALMON P., *Synthesis of intermediate pictures for autostereoscopic multiview displays*, Proc. Int. Workshop on HDTV'94, Torino, Italy, Oct 94.

[GU94] GU C., KUNT M., *Contour simplification by a new non-linear filter for region-based coding*, SPIE Vol.2308, Proc. of VCIP'94, pp 1180-1191, Chicago, Sept 94.

[HOL77] HOLLAND P.W., WELSCH R.E., *Robust regression using iteratively reweighted least squares*. Commun. Stat. - Theor. Meth., A6, pp 813-828, 1977.

[MAR93] MARQUES F., SAULEDA J., GASUL A., *Shape and location coding for contour images*, Proc of PCS'93, Lausanne, March 93.

[ODO95] ODOBEZ J.M., BOUTHEMY P., *Robust multiresolution estimation of parametric motion models*, Journal of Visual Communication and Image Representation, Vol.6, No 4, Dec 1995.

[PER94] PEREZ P., HEITZ F., BOUTHEMY P., *Multiscale minimization of global energy functions in some visual recovery problems*, CVGIP : Image Understanding, Vol.59, No 1, pp 125-134, January 1994.

[SIK95] SIKORA T. and MAKAI B., *Shape-Adaptive DCT for Generic coding of Video*, IEEE Trans. Circuits and Systems for Video technology, Vol.5, No; 1, Feb 1995.