ACTIVE MESH RECONSTRUCTION OF BLOCK-BASED MOTION INFORMATION

Xavier Marichal, Benoît Macq

Laboratoire de Télécommunications et Télédétection, UCL
2 place du Levant, 1348 Louvain-la-Neuve, Belgium
E-mail: {marichal,macq}@tele.ucl.ac.be

ABSTRACT

This paper proposes an asymmetric scheme for motion estimation/compensation. While the estimation is performed with a classical Block Matching Algorithm, the motion information is decoded by using an active mesh in order to implement the compensation stage. A mesh is positioned by taking into account the relevant spatial information of the image to be compensated and is used afterwards to reconstruct the motion information. Two main issues have to be addressed for conducting such a motion compensation technique: i) how to optimally design an active mesh, ii) how to reverse and interpolate a backward motion field estimated on an a priori grid of fixed-size blocks so as to determine a forward motion field on variable size triangular patches. While proposing a solution to these two problems, particular attention is paid to the computational burden. Such a scheme opens the possibility for added manipulation functionalities because of mesh capabilities.

1. INTRODUCTION

In the framework of visual communications over (very) low bitrate channel, motion estimation (ME, [17]) and compensation can be considered as the main techniques to reduce the temporal correlation inherent to video sequences so as to enable codecs to reach the required compression ratios.

The functioning principle of most motion estimation and compensation algorithms is as follows: given two consecutive images of a sequence, $I(t-1)$ and $I(t)$, the goal is to obtain the best possible prediction of $I(t)$ by exploiting the motion information existing between two frames. Motion estimation aims at detecting this motion field.

Since its introduction in 1981, the Block Matching Algorithm (BMA, [7]) has emerged as the ME technique achieving the best compromise between complexity and quality: a fast estimation procedure allows obtaining a block-based motion field that is transmitted at low-cost. These properties have allowed the BMA to be included in most video standards like H.263, MPEG-1,2 and MPEG-4 [13]. Unfortunately, the compensation stage provides a predicted image that suffers from so-called blocking artifacts.

Affine models perform a better compensation stage, as they are able, what with six free parameters, to tackle rotations and zoom effects in addition to the translation. Recently, Nakaya and Harashima [11] proposed to implement the affine transform by means of a triangular mesh surimposed on the image: the six parameters of the affine transform are uniquely determined by the three motion vectors of the tops of any triangular patch (mesh element). Dudon extended this work to active mesh that are automatically adapted to the spatial contents of the image [6].

The adaptation of the vertices' location to the spatial contents enables increasing the relevance of the estimated motion field by an a priori liaison between the spatial and the temporal information. Other advantages of such adaptive mesh structures is the huge domain of applications and added functionalities they permit to cover: some examples are 3-D modeling [2, 8] or the link with with fractal models for intra-coding [3], but also video editing effects like synthetic object transfiguration [16], augmented reality and other functionalities directly related to Synthetic and Natural Hybrid Coding [5] of MPEG-4.

The challenging idea of the research described here is to combine the advantages of both previously described techniques: the classical BMA and the implicit affine motion model developed by the triangular mesh. The principle is as follows: nothing is changed concerning the motion estimation; a classical BMA is applied and the resulting motion vectors are transmitted as such. The aim of this paper is research in how to modify the compensation (reconstruction) stage: the motion vectors are not merely used in order to displace blocks of the reference frame but rather to serve as an information set to warp a mesh that has been built on this reference image. Such a combination seems to be significant in several ways: first, it enables to heighten the subjective quality of the reconstructed pictures by taking spatial information into account (adaptive vertices location) and allowing a richer motion field parameterization (affine transform). This is performed with no increase either in the computational burden of the estimation or in transmission cost. Second, and most important, it improves the representation stage and allows for the use of new functionalities without loosing its compatibility with existing standards, since the bitstream is not modified.

Section 2 briefly exposes how the mesh is automatically adapted to the image content. Section 3 describes the way motion information is interpolated. Finally, section 4 presents an example image and the authors draw some conclusions in section 5.

This work is supported by a fellowship of the Belgian F.R.I.A.

2. MESH DESIGN

The first step of the reconstruction algorithm is the automatic design of a mesh on the reference image. As our aim is not only to improve the quality of the reconstructed picture but also to allow for further content manipulation, it is important to design a content-based mesh that matches boundaries of patches with relevant scene features. The retained method to design the mesh thus consists of i) detecting image corners to use them as mesh vertices; and ii) building the mesh by the well-known Delaunay triangulation [15].

As precise location on real edges seemed a priority [18], eventually the authors opted for a solution that:

1. Precisely locate image edges: the robust half boundaries detector of Noble [12] is used to detect edges as well as multiple points.
2. Track the detected edges in order to select high curvature points: the finite differentiation technique of Najman and Vaillant [10] implements this step.

3. MOTION TRANSCRIPTION

The aim of the motion transcription is to determine the motion vector to be applied to every vertex of the mesh designed in the previous step. The problem is of double nature:

1. While adapting the motion information to the active mesh, an important point must be taken into account: this information should be reversed. The BMA, as implemented in most standards, is effectively computed "backwards", while the mesh, implemented on the image to be motion compensated, needs "forward" vectors.
2. The second point of the transcription change is to interpolate the motion field: the BMA information can be considered as a coarse subsampling of a dense motion field. The motion of the vertices represents another subsampling of the same dense field. The problem is that the dense motion field is not known. Combined with the inversion problem mentioned above, an appropriate interpolation technique should be found. It should also remain as simple as possible so as not to increase the computational burden too much.

3.1. Reversing the Sense of the Motion Information

If one considers the estimation performed by the BMA as a coarse subsampling of a dense motion field, the detected vector $(\hat{u}, \hat{v})$ is assigned to the center of the estimated block. This means that, in the forward direction, it is equivalent to a vector $(-\hat{u}, -\hat{v})$ located at the position indicated by the vector.

3.2. Interpolating the Motion Values

Once motion samples placed on an irregular grid have been obtained, the goal is now to estimate the values of other samples, placed on another irregular grid. Such an estimation may be made thanks to a theory of 2-D interpolation established by the South African mining engineer Krige. This technique is referred to as kriging and has been further developed by Matheron [9]. It has already been successfully applied to motion vector interpolation problems [4].

Intuitively, kriging solves the problem in a roundabout way: the criterion to determine the unknown values of the searched samples is that, if those were in turn interpolated, they would have to provide known points with known values.

For this purpose, it uses a kriging operator that maps the unknown vector field on the known one. If one considers a known vector located on a position X belonging to a triangle ABC, the affine transform (implicitely implemented by the triangle) allows to establish a direct relation between the unknown motion vector $\vec{X}$ and the motion of the tops of the triangle ABC.

As every vector of the reversed BMA motion field belongs to one triangle of the mesh, such an equation can be iterated. According to the relative numbers of BMA vectors and mesh vertices and also to the pertinence of all kriging equations, the system may be under-, over- or simply determined. In the two first cases, no exact solution exists, and one is in front of a linear least square problem that may be solved using Singular Value Decomposition (SVD [14]).

Although the number of equations may be quite large (99 when the BMA is performed on every 16×16 block of a QCIF image), it has to be noted that only three parameters are non zero. The massive presence of null parameters in the system matrix fasten the so-called QR decomposition with column pivoting involved by the SVD algorithm.

Finally, it must be checked whether the motion vectors do not enforce the mesh to violate its connectivity constraint (such a constraint is inherent to any mesh). The troubling motion vectors are rectified as an interpolation of the motion vectors of the neighboring nodes (equations (47) and (48) of [1]).

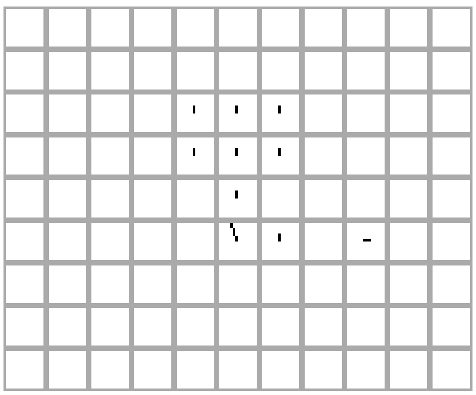
4. RESULTS

In order to demonstrate the validity of the proposed scheme, its results will be compared to those of a classical BMA compensation. Figure 1 summarizes the various steps of the process with the Akiyo images: the original images 27 (a) and 30 (b) are motion estimated via a classical "backward" BMA. Instead of merely applying the resulting motion field (c), which would result in image (d), an adaptive mesh is surimposed (e) on the original image (a). Thanks to kriging equations, this mesh is warped in order to obtain the final compensation (f).

In terms of Peak Signal-to-Noise (PSNR) ratios, the two compensation techniques (BMA and proposed scheme) lead to similar results. The increase of complexity introduced by the asymmetric reconstruction is therefore mainly justified by the availability to produce more pleasant reconstruction (without blocking discontinuities) and the possibility to directly interact with the content of the picture. The latter point is of major importance for the new framework offered by the MPEG-4 standard.

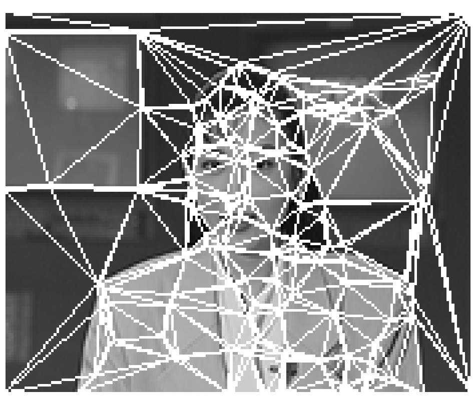
(a) Original image at time t-1

(b) Original image at time t



(c) Backward motion field from BMA

(d) BMA compensation



(e) Automatic mesh design

(f) Asymmetric compensation using mesh

Figure 1: Some steps of the asymmetric process on Akiyo sequence

5. CONCLUSION

The aim of this paper was to combine the advantages of two separate techniques used for motion estimation/compensation. On the one hand, the Block Matching Algorithm has been selected for its efficient estimation as well as for its compact representation of the motion field. On the other hand, affine models offer a more sophisticated representation of the motion that avoids blocking artifacts on the objects contours. When affine models are implemented via active mesh that establishes an a priori link with spatial information, they provide the user with new options: 3-D modeling, video editing, transfiguration, augmented reality, etc.

An asymmetric scheme has thus been proposed: after a classical BMA estimation of the motion field, the compensation stage is implemented via mesh warping. In so doing, solutions to several problems have been offered: a fast and accurate corner finder has been set up, the motion information has been reversed and it has been interpolated thanks to the kriging technique. All these innovations allow to automatically change the representation of the motion information, from block-based rigid motion fields to active ones designed on an adaptive mesh. Moreover, particular attention has been paid to the computational burden of the scheme so as not to penalize the decoder end.

The proposed scheme is totally compatible with existing compression standards. It could be for instance used to edit and manipulate MPEG video sequences stored in a database without having to recalculate all the motion information.

In addition to the new options the scheme offers, it has been shown to somehow improve subjectively the quality of the compensated images.

6. REFERENCES

[1] Y. Altunbasak and M. Tekalp. Closed-form connectivity-preserving solutions for motion compensation using 2-d meshes. IEEE Transactions on Image Processing, 6(9):1255–1269, September 1997.

[2] G. Bozdagi, M. Tekalp, and L. Onural. 3-d motion estimation and wireframe adaptation including photometric effects for model-based coding of facial image sequences. IEEE Transactions on Circuits and Systems for Video Technology, 4(3):246–256, June 1994.

[3] F. Davoine. Compression d'images par fractales basée sur la triangulation de Delaunay. PhD thesis, Institut National Polytechnique de Grenoble, December 1995.

[4] E. Decenciere and P. Salembier. Morpheco deliverable. chapter 3: Application of kriging to image coding. Morpheco Consortium, R2053/UPC/GPS/DS/R/016/b1, January 1996.

[5] P.K. Doenges, T.K. Capin, F. Lavagetto, J. Ostermann, I.S. Pandzic, and E.D. Petajan. Mpeg-4: Audio/video and synthetic graphics/audio for mixed media. Signal Processing: Image Communication, 9(4):433–464, May 1997.

[6] M. Dudon, O. Avaro, and G. Eude. Object-oriented motion estimation. pages 284–287, Sacramento, September 1994. Picture Coding Symposium (PCS).

[7] J.R. Jain and A.K. Jain. Displacement measurement and its application in interframe image coding. IEEE Transactions on Communications, 29(12):1799–1806, December 1981.

[8] F. Lavagetto and S. Curinga. Object-oriented scene modeling for interpersonal video communication at very low bit-rate. Signal Processing: Image Communication, (6):379–395, June 1994.

[9] G. Matheron. Estimating and Choosing. Springer-Verlag, Berlin, 1989.

[10] L. Najman and R. Vaillant. Topological and geometrical corners by watershed. volume LNCS 970, pages 262–269, Prague, September 1995. Computer Analysis of Images and Patterns.

[11] Y. Nakaya and H. Harashima. An iterative motion estimation method using triangular patches for motion compensation. pages 546–557. Society of Photo-Instrumentation Engineers, November 1991.

[12] J.A. Noble. Finding half boundaries and junctions in images. Image and Vision Computing, 10(4):219–232, May 1992.

[13] F. Pereira, K. O'Connell, R. Koenen, and M. Etoh. Special issue on mpeg-4, part 1: Invited papers. Signal Processing: Image Communication, 9(4), May 1997.

[14] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. Numerical Recipes in C - The Art of Scientific Computing. Cambridge University Press, Cambridge, 2 edition, 1992.

[15] S.W. Sloan. A fast algorithm for constructing delaunay triangulations in the plane. Adv. Eng. Software, 9(1):34–55, 1987.

[16] C. Toklu, T. Erdem, I. Sezan, and M. Tekalp. Tracking motion and intensity variations using hierarchical 2-d mesh modeling for synthetic object transfiguration. Computer Vision, Graphics, and Image Processing: Graphical Models and Image Processing, 58(6):553–573, November 1996.

[17] G. Tziritas and C. Labit. Motion Analysis for Image Sequence Coding. Elsevier, Amsterdam, 1994.

[18] Y. Yokoyama, Y. Miyamoto, and M. Ohta. Very low bitrate video coding using warping prediction adaptive to object contours. pages M–4, Tokyo, November 1995. International Workshop on Coding Techniques for Very Low Bit-rate Video (VLBV).