GLOBAL MOTION ESTIMATION FOR MPEG-ENCODED STREAMS

Renan Coudray, Bernard Besserer

L3I, University of La Rochelle
Av. Michel Crépeau 17042 La Rochelle cedex 1, FRANCE
E-mail: renan.coudray@univ-lr.fr, bernard.besserer@univ-lr.fr

ABSTRACT

This paper explains a method for global motion characterization using a Hough-Transform-like technique to estimate the motion parameters. Motion information provided by each picture sample is gathered in a accumulator corresponding to a parameter space, which is derived from our simple motion model. The document then focuses on the use of MPEG stream as input data, dealing with the **available motion compensation information** as input vectors for our system. Pros, cons and improvements are discussed.

1. INTRODUCTION

As the amount of archived videos continuously grows, the demand for video annotation and metadata generation increases in order to catalog, sort or categorize the huge amount of sequences often stored in digital form[1]. Several approaches for video indexing based on single image (snapshots) had be issued [2], but the processing of the dynamic behavior of the sequences could improve sequence characterization [3]. Global motion estimation (generally induced by camera motion, sometimes referred as camera motion) is a key technology for video annotation applications. The proposed approach relies on three predominant points :

- The input data is a MPEG stream [4], since the MPEG1 or MPEG2 standards are widely used for digital video storage. A proper and direct use of the information available in compressed form can achieve very fast processing. Global Motion Estimation performed in MPEG4 or MPEG7 is not exploited because its use is still marginal and digital video broadcasting standard (DVB [5]) handle MPEG2 or MPEG1 streams.
- The used model for camera motion is quite simple, but satisfactory to spot the most prevalent motions such as panning or zooming. A coarse sequence classification can be attempted.
- After the global motion estimation, the whole sequence can be reverse-compensated for this motion. Residual motion is then more discriminant and could be used as unique descriptor for the sequence (fingerprinting).

The philosophy of our method for global motion estimation is very similar to the Hough technique for finding shapes. The idea is that each sample holding a relevant motion vector puts its contribution to a globally consistent solution. Uncoupling the basic camera displacement led to three accumulators for the search of an optimum within these parameters spaces, characterizing the global motion.

2. SIMPLE MOTION MODELS

For the further discussion on camera motion model and parameters estimation, we assume that the input data is a motion vector field. The used notation is :

V_x, V_y	: motion vector components for each sample within the image
x, y	: spatial position of the sample

In our approach, the video global motion is assimilated to the camera movement. The approximation used are explained below along the model equations.

2.1. Translation

We assume that the camera translates only, and panning and tilting movements are approximated as translation from frame to frame (acceptable for small displacements, a distant scene and a high frame rate). Thus, the displacement along the x-axis is called *pan factor*, and displacement along the y-axis *tilt factor*. According to the given approximations, a pure side traveling results in a identical translation vector for each sample.

t_x : pan factor	
t_y : tilt factor	

$$V_{T_{(t_x,t_y)}}(x,y) = \begin{pmatrix} t_x \\ t_y \end{pmatrix} \quad (1)$$

2.2. Zoom

Zooming and forward traveling (dolly) are considered equivalent; this is true for small amplitudes (typically for motion vector computed from frame to frame). For a pure zoom, all

0-7803-8554-3/04/$20.00 ©2004 IEEE.

apparent motion vectors point to (or from) the zoom center, which matches the optical center. But several motions could be combined, so the apparent zoom center could be anywhere inside or outside the frame. The motion magnitude attached to each sample depends of its proximity to the zoom center ; Moreover, the x-magnitude of the zoom is proportional to the x-distance between the sample and the zoom center (y-magnitude prop. to y-dist., respectively).

$$\boxed{\begin{array}{r} z : \text{zoom factor} \\ z_x,\, z_y : \text{zoom center} \end{array}} \quad V_{Z_{(z,z_x,z_y)}}(x,y) = \begin{pmatrix} z.(x-z_x) \\ z.(y-z_y) \end{pmatrix} \quad (2)$$

2.3. Rotation

As for zooming, the magnitude of motion vectors is proportional to the distance to the rotation center, but orthogonal to the line connecting the sample and the rotation center. If zoom and rotation have the same center : $V_R \perp V_Z$. For composite motion cases, the motion is broken up as translation followed by a centered rotation.

$$\boxed{\begin{array}{r} r : \text{rot. factor} \\ r_x,r_y : \text{rot. center} \end{array}} \quad V_{R_{(r,r_x,r_y)}}(x,y) = \begin{pmatrix} -r.(y-r_y) \\ r.(x-r_x) \end{pmatrix} \quad (3)$$

3. PARAMETER ESTIMATION

In literature, several approaches point toward a grouped estimation of the global motion parameters, using all the data at once and regression methods. Theoretically, the global motion affects all samples within the recorded image, but this method works well if the sequence motion is also homogeneous. But in practice, many objects in the scene shows erratic movements. Thus, reject algorithms are used and the parameter estimation relies on time-consuming recursive technics[6, 7].

Our parameter estimation operates rather like the Hough transform : motion data associated to each sample is converted into contributions to a parameter space representing possible motions according to our models. First, each possible motion (pan, tilt, zoom, rotation) will be considered separately, then we give a method to combine these estimations.

3.1. Translation

This this the simplest case, the translation is described comparably by each sample :

$$\begin{array}{ll} t_x & = V_x(x,y) \\ t_y & = V_y(x,y) \end{array} \quad (4)$$

For each motion vector belonging to each sample, we assume that it contains a translation information due to camera motion. According to the model, we plot the couple (V_x, V_y) in the accumulator. For each image, the highest peak in the accumulator will correspond to the translation parameters (t_x, t_y). Repercussion from other moving objects visible in the scene are distributed around the point of the true parameters (t_x, t_y), thus the right peak in the accumulator may be blurred, noisy or presents ambiguity towards other peaks, so maximum (or main mode) extraction is still a problem. Consider that the Hough approach is suitable for problems having enough data to support the expected solution.

3.2. Zoom

Again, we consider a pure zooming. According to our model, the zoom factor is computed from the directional derivative of the motion vectors for each sample. Direction and amplitude of the motion vectors belonging each sample depends of their location compared to the apparent optical axis.

The directional derivative $\nabla_u f(x_0, y_0)$ is the rate at which the function $f(x,y)$ changes at a point $X_0 = (x_0, y_0)$ in the direction u. The directional derivative can be formulated as :

$$\begin{array}{lll} \text{continuous} & \nabla_u f(x_0,y_0) & = \lim_{h\to 0} \frac{f(X_0+hu)-f(X_0)}{h} \\ \text{discrete} & \nabla_u f(x_0,y_0) & = f(X_0+u) - f(X_0) \end{array} \quad (5)$$

Expressing z as $z = z \cdot (x+1-z_x) - z \cdot (x-z_x)$ and using the zoom model equation (2) and the directionnal derivative in a discrete space (5) gives :

$$\begin{aligned} z &= V_{Zx}(x+1,y) - V_{Zx}(x,y) \\ &= \nabla_x V_{Zx} \end{aligned} \quad (6)$$

In a similar way, the zoom factor z can be expressed in relation to the directional derivative in the y-direction :

$$\begin{aligned} z &= z \cdot (y+1-z_y) - z \cdot (y-z_y) \\ &= V_{Zy}(x,y+1) - V_{Zy}(x,y) \\ &= \nabla_y V_{Zy} \end{aligned} \quad (7)$$

3.3. Rotation

Apparent motion for a rotation is orthogonal to apparent motion induced by zooming. This relationship appears as well in the parameter estimation. Again, let us express the rotation factor r as :

$$r = r \cdot (x+1-r_x) - r \cdot (x-r_x) \quad (8)$$

Therefore, from (3) and (8), the rotation factor can be stated as :

$$\begin{aligned} r &= V_{Ry}(x+1,y) - V_{Ry}(x,y) \\ &= \nabla_x V_{Ry} \quad \text{and} \quad \text{also} \quad r = -\nabla_y V_{Rx} \end{aligned} \quad (9)$$

3.4. Composite Motion

Camera motion in real video sequences can be very complex, and the resulting motion is often a mix. According to our projections into the parameter spaces, a translation (sideways traveling for example) will not alter the estimation for the zoom and rotation parameters. The translation adds a constant value (t_x,t_y) to the motion vectors, but the contribution for the zoom and rotation parameter space relies on directional derivatives :

$$\nabla_x V_{Tx} = \nabla_y V_{Ty} = \nabla_y V_{Tx} = \nabla_x V_{Ty} = 0 \qquad (10)$$

Moreover, the zoom parameter estimation is not affected by the presence of a rotation and conversely :

$$\begin{cases} \nabla_y V_{Zx} = \nabla_x V_{Zy} = 0 \\ \nabla_x V_{Rx} = \nabla_y V_{Ry} = 0 \end{cases} \qquad (11)$$

But a zoom or a rotation affects the estimation for the translation. We propose a two-step method, by estimating first the zoom and rotation parameters and performing a back-compensation for each motion vector according to these parameters, and finally estimating the translation parameters (**Fig. 1**).

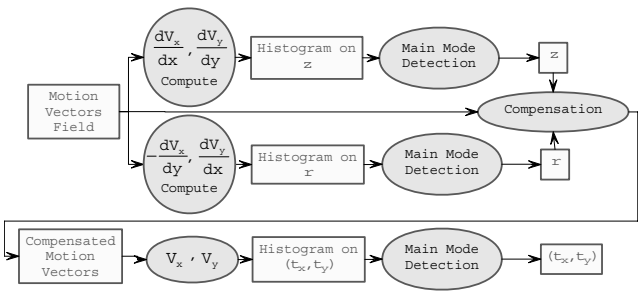


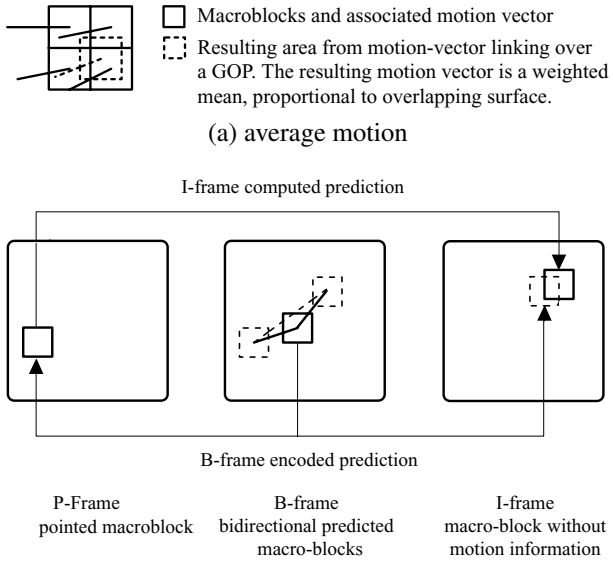
Fig. 1. Global motion estimation process

4. USING MPEG STREAMS AS INPUT

As stated before, the input data for our application is an MPEG stream. MPEG compression uses motion compensation in order to achieve higher compression rates. Briefly, a MPEG stream is structured as a succession of GOPs (Group Of Picture). Each GOP begins with an I-frame (Intraframe, fully encoded image without motion compensation vectors), followed by P (Predicted) and B (Bidirectionnal) frames, which hold motion compensation information (attached to macroblocks as motion vectors), thus directly readable from the stream. Let us point out two difficulties : first, motion compensation use the block matching technique, therefore the motion vectors are not always accurate, and secondly, a motion compensation vector is available only for every macroblock (16×16 pixels). We though use this coarse rastering as input, getting a rather sparse vector field.

4.1. Motion Vector Aggregation

The main problem with MPEG streams is the fact that all pictures aren't motion compensated (I-Frame, for example). Also, motion compensation could be forward or backward, or both. We override this by connecting motion vectors over a complete GOP, so each I-Frame holds for each macroblock a motion vector pointing to the previous I-Frame. Additionally, the complete picture information available at an I-Frame is usefull for the rejection rule (see 4.2). Globally, the linking is based on P-Frames, but motion vectors cannot be easily added from an P-Frame to the next : motion compensation vectors are referenced to the macroblock raster of a particular frame, and points to the previous frame at any place with a half-pixel accuracy. The corresponding location of the compensated macroblock therefore overlaps up to four macrocrocks belonging to the raster of this previous frame. The vectors for the new locations are computed from the available motion vectors like shown in **Fig. 2.a**, according to their respective overlapping surface. Also, at the end of a GOP, B-Frames have to be used to close the gap to the next I-Frame (**Fig. 2.b**).



(a) average motion

(b) I-Picture Movement

Fig. 2. Motion vector linking over a GOP

4.2. Fast Reject

We remember that the motion compensation for MPEG compression is intended for compression. Specifically within large homogeneous regions, a macroblock from frame t is very similar to a neighboured macroblock from frame $t+1$, regardless of its position. Therefore, these motion vectors are erroneous if assimilated to apparent optical flow. Our global motion estimation gives better results if these vectors are discarded. Only vectors attached to macroblock

with relevant information (edges, textures) should be used as contributors for the accumulation in the parameter space. The DCT encoding used in MPEG can be directly used, the sum of the squared DCT coefficients expressing an energy or texture degree of the block, according to [8]. If this degree is low, block's motion information is ignored.

5. EXPERIMENTATION ANS RESULTS

Actual implementation fulfills the partial MPEG decompression, the GOP motion vector field chaining and completion and the Global Motion Estimation (GME). Accumulated parameter space is filtered (gaussian) and a polynomial approximation around the peak is performed to improve r and z accuracy (limited by the parameter space discretisation) and a simple maximum detection permit to estimate t_x and t_y. In the example shown (**Fig. 3**), the motion vectors induced by the walk of the character, although covering a large surface, does not alter the global motion estimation.

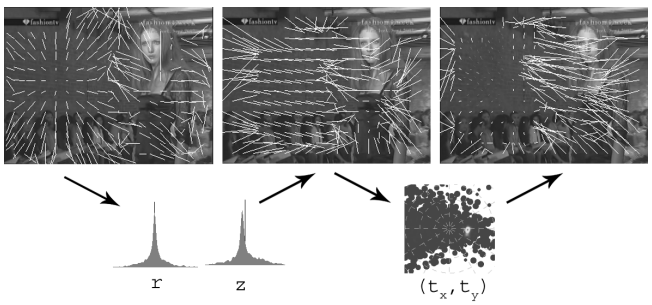


Fig. 3. Global Motion Estimation : (left) Initial accumulation for estimating r and z, (middle) back-compensation using the estimated r and z, (right) new accumulation in t_x, t_y space for the estimation of translation

The complete computing time for r, z and t_x, t_y estimation is about .08 ms (P*4* 2.4GHz) on MPEG2 streams emitted by DVB broadcasters (12-frames GOP, 720 $\times$ 576), so we are ca. 6 times faster than video real time, freeing computing power for useful exploitation of the GME results.

In term of accuracy, we have compared our parameters with the parameters issued by the Motion2D[1] software which use uncompressed data. Motion2D computes GME for all consecutive pictures ; for comparison, we summed GME parameters over 12 consecutive frames, as our software computes GME GOP-wise. The results shown (**Fig. 4**) have been performed on the "Mobile and Calendar"[2] sequence, using a software based MPEG encoder to feed our Global Motion estimator.

6. CONCLUSION AND FUTURE WORKS

Although the results are convincing, some sequences lead to faulty motion estimation, especially whose with text insets (subtitles, logos). Rather than a plain maximum search, a forthcoming more clever algorithm will check the distribution, which will be useful if ambiguities exist (two distinct

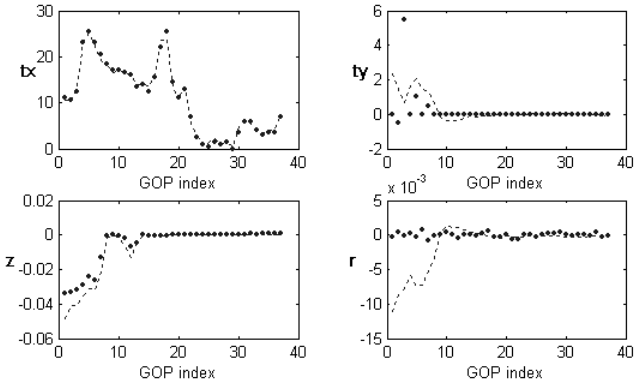


Fig. 4. GME comparison : Round dots illustrate the r, z and t_x, t_y estimation of our method, dotted lines are for the Motion2D software. Results are very similar, slightly more precise for rotation by our method than the estimation provided by Motion2D. Our method shows a strong outlier on the t_y estimation (third GOP), because the vertical movement of the falling calendar became dominant and no temporal filtering is used.

modes). At last, the algorithm will be tested for video retrieval, using a signature computed from the residual motion after compensation.

[1] http://www.irisa.fr/Vista/Motion2D/
[2] ftp://ftp.tek.com/tv/test/streams/Element/index.html

7. REFERENCES

[1] A. Murat Tekalp, *Digital video processing*, Prentice Hall Ed., 1995.

[2] V. Kobla, D. Doermann, and K. Lin, "Indexing and retrieval of mpeg compressed video," *Journal of Electronic Imaging*, vol. 7, no. 2, 1998.

[3] S. Porter, M. Mirmehdi, and B. Thomas, "Video indexing using motion estimation," *The British Machine Vision Conference*, 2003.

[4] *ISO/IEC 13818-1 and ISO/IEC 13818-2*, 2000.

[5] H. Benoit, *Digital Television MPEG-1, MPEG-2 and Principles of the DVB System, second edition*, Focal Press, 2002.

[6] R. Wang and T. Huang, "Fast camera motion analysis in MPEG domain," *IEEE Trans. Image Processing*, vol. 3, pp. 691–694, 1999.

[7] I. Grinias and G. Tziritas, "Robust pan, tilt and zoom estimation," *Int. Conf. on Dig. Signal Processing*, 2002.

[8] R. Fablet, P. Bouthemy, and P. Perez, "Texture discrimination using discrete cosine transformation shift-insensitive (DCTSIS) descriptors," *Pattern Recognition*, vol. 33, no. 10, pp. 1585–1598, 2000.