# EFFICIENT CAMERA MOTION CHARACTERIZATION FOR MPEG VIDEO INDEXING

*Jae-Gon Kim*, Hyun Sung Chang*, Jinwoong Kim*, and Hyung-Myung Kim***

*Electronics and Telecommunications Research Institute
Taejon, 305-350, KOREA
**Dept. of EE, Korea Advanced Institute of Science and Technology
Taejon, 305-701, KOREA

## ABSTRACT

In this paper, a novel approach to the camera motion analysis is proposed to index videos compressed in MPEG-1 or MPEG-2. Specifically, it fits the motion vectors in MPEG stream into the two-dimensional affine model to detect basic camera operations automatically. The proposed approach involves 1) the construction of motion vector fields (MVFs) by normalizing the types of motion vectors and filtering out noises; and 2) the qualitative interpretation of camera motions from the estimated model parameters in two levels (frame and temporal segment). A fine segmentation can also be obtained for a video, based on the homogeneity of the camera motion in each unit. The advantages of our method lie in its computational efficiency and robustness to noisy environments such as false motion vectors and object motions. The proposed approach is validated by the experiment with real compressed video sequences.

## 1. INTRODUCTION

With the expansion of the digital broadcasting, Internet and digital library services, the digital video data became very popular. Today content-based indexing techniques have been actively developed in order to access video material in an effective way. A basic and common approach for video indexing is to segment the video sequences first into shots and then to extract various features for shot characterization.

An important feature in video sequences is the temporal intensity change between successive pictures: apparent motion. The apparent motion is generally attributed to the motion of objects and the motion of the camera. Camera motion is a distinct feature that essentially characterizes the content of a shot. Camera operation also gives cues for inferring the high-level semantic meanings such as the intention of video producers. Furthermore, a video sequence is composed of consecutive camera operations and a shot is a sequence of frames with varying camera operations, but continuous filming. Therefore, a shot can be segmented into smaller units of sub-shot that keeps a homogeneous camera motion for more detailed manipulation.

Until now, some approaches have been developed to analyze camera motion in video sequences in the context of content-based indexing [1]-[7]. Most of the existing methods are based on analyzing the optical flow computed between consecutive images [1]-[4]. However, the estimation of the optical flow, which is usually based on gradient methods or block matching methods, is computationally expensive. One can classify two types of approaches for camera motion analysis through optical flow. The one defines a model for representing camera motion and estimates model parameters from the computed optical flow [1], [2]. The other directly analyzes the observed optical flow patterns without any motion model by using the angular distribution or the power of optical flow vectors [3]-[6].

On the other hand, since video data are usually available in the MPEG-compressed form, it is desirable to directly process the compressed video without decoding. A few methods that directly manipulate MPEG compressed video to extract camera motion have been proposed [5], [6]. These approaches use MPEG motion vectors as an alternative to optical flow which allows us to save high computational load in two steps: full decoding the bitstream and optical flow computation. The previous methods simply classify the limited set of camera motion through the analysis of the distribution of motion vectors. Besides, some authors have remarked possibility of the extension of their proposed techniques performed in raw video to compressed video as the further work [1], [2].

Concerning camera motion analysis in an MPEG compressed video, it is necessary to focus on the following factors in order to obtain sufficient quality of results for video indexing. First, in order to take the strong point of computational efficiency, a preprocessing is necessary to eliminate noisy motion vectors extracted from the bitstream. Second, it should be resilient to the presence of moving objects of large size. These two factors are related to the reliability of the method. Finally, it should be able to classify full set of camera motion types that are sufficient to describe various real video sequences. However, there has not been extensive work well addressing automatic camera motion characterization in compressed video in terms of the above factors.

In this paper, we propose a novel framework for the analysis of camera motion to index MPEG compressed video sequences. Figure 1 shows the overall procedure of the proposed approach. It consists of three main steps. We extract the raw motion vectors from MPEG bitstream by partial decoding and construct an MVF for every frame (step 1). Our approach is based on the estimation of a two-dimensional affine motion model using the constructed MVF accounting for apparent global motion between two consecutive frames (step 2). Then, we can recognize the segment, in which a specific camera operation is maintained, through the qualitative interpretation of the estimated affine model parameters. We define a set of camera motion class that consists

0-7803-6536-4/00/$10.00 (c) 2000 IEEE

of six well-known basic camera operations: They are *zoom, rotation, pan, tilt, object motion* and *static*, respectively. A shot is finally indexed by camera motion characterization and a sequence is segmented into the unit of sub-shot on the basis of camera motion (step 3).

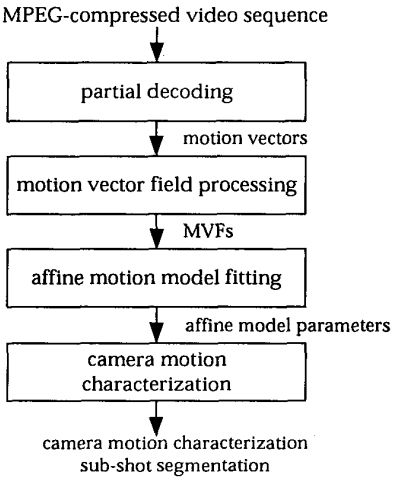



**Figure 1**. The overall procedure of the proposed approach.

The main features of the proposed method are as follows. 1) Camera motion analysis on the MVFs avoids heavy complexity spent in full decompression of MPEG stream and optical flow calculation. 2) The MVF processing step which includes the normalization of the motion vector types and noise vector filtering increases the reliability in the camera motion detection. 3) The qualitative interpretation of camera operations in this method works reliably even with the presence of moving objects of large size or noisy motion vectors since it utilizes the physical properties of camera operations. The proposed approach is advantageous over the others in terms of the efficiency and robustness.

## 2. MOTION VECTOR FIELD PROCESSING

First, we normalize the types of the motion vectors extracted from MPEG-1 or MPEG-2 bitstream to construct the MVFs for every frame. In other words, every motion vector should be converted to forward-predicted one with the prediction distance of one frame, regardless of the picture coding type and the prediction mode that is revealed in the macroblock (MB) type information. As to the picture structure, we only consider the frame picture and need to be extending for the field picture. To construct a MVF, as in our early work [8], we 1) convert two field-motion-vectors to a frame-motion-vector for the case of the field prediction in MPEG-2; 2) map the motion vectors of backward-predicted and bidirectionally-predicted types into forward-predicted ones; and 3) estimate motion vectors for I frames by interpolating two nearest P frames.

As well known, the MPEG motion vector does not always correspond to true optical flow since the estimation is carried out to minimize the prediction errors in the compression. Especially, in a nearly uniform region, the motion vectors look like random noises. Therefore, it is necessary to preprocess the constructed MVF to enhance reliability. We filter out the suspected noisy components from MVF by applying median filters to the magnitude of horizontal and vertical components of the motion vector separately. For our purpose, simple median filtering is effective enough to remove random noisy vectors in the background region.

## 3. CAMERA MOTION CHARACTERIZATION

Unlike the previous approaches also applied in compressed domain [5], [6], we use a two-dimensional parametric model to represent global motion, and then interpret qualitative terms of the camera motion from the estimated model parameters. The affine model was employed in the following considerations although there are more accurate models of higher order. First, the affine model is more resilient to the noisy and sparse MVF conditions. In additions, it can represent all of the basic-type camera motions to be used in content-based indexing. In the affine motion model, the motion vector $(u,v)$ is expressed as

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} a_2 & a_3 \\ a_5 & a_6 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} a_1 \\ a_4 \end{pmatrix}$$

at the position of a MB $(x,y)$, where $\Phi = (a_1, a_2, a_3, a_4, a_5, a_6)$ is the parameter vector estimated from the MVF of the undergoing frame by using the least square (LS) method.

After the estimation of the parameters, we can exploit this information for camera motion characterization aiming at detection of the predefined classes of camera motion and micro-segmentation in the unit of sub-shot. The parameters can be expressed in another basis of elementary fields as in [2], [9], which are more directly related to the physically meaningful camera motion, as follows:

$$\Phi' = (pan, tilt, div, rot, hyp_1, hyp_2), \text{ with :}$$
$$pan = a_1 \qquad tilt = a_4$$
$$div = \frac{1}{2}(a_2 + a_6) \qquad rot = \frac{1}{2}(a_5 - a_3)$$
$$hyp_1 = \frac{1}{2}(a_2 - a_6) \qquad hyp_2 = \frac{1}{2}(a_3 + a_5).$$

These terms of the transformed parameter vector $\Phi'$, *pan*, *tilt*, *div* and *rot* represent each of the component of the MVF induced by the camera operations of pan (or horizontal tracking), tilt (or vertical tracking), zoom (or forward/backward tracking), and rotation, respectively. For the camera motion characterization, we have developed a novel qualitative interpretation method that is based on global thresholding on the transformed parameters to detect a significant term of the camera motion. In practice, we use a modified term $hyp\,(hyp = |hyp_1| + |hyp_2|)$ instead of using $hyp_1$ and $hyp_2$ separatively, which can not be induced by normal camera operation, to detect the class of object motion. This class include the following cases: object motion is dominant due to moving objects of large size; ambiguous motion due to the completely meaningless MVF in the exceptional cases (for example, sky area).

The method consists of three steps: frame-level detection, segment-level detection, and residual segment processing. We can detect sequentially each of the camera motion class in order of zoom, rotation, pan, tilt, object motion and static by performing the above first two steps (i.e. frame-level detection and segment-level detection) on each of the related parameters. Let us take the example of the detection of zoom. In the frame-level detection step, we directly threshold the magnitude of *div* to determine if zoom is present in the undergoing frame. We also use the fact that the parameters $a_2$ and $a_6$ should have the same sign in the case of zoom.

After the frame-level detection for every frame of the sequence, we detect the segment that have a specific class of camera motion in the segment-level detection step. We observe that a type of camera operation is generally maintained for some time at least a half second. According to this observation, we perform the thresholding on the interval of the frames detected as the same class. These procedure are performed in turn on different classes to be detected. The detected camera motion can be applied to characterize the corresponding shots, therefore a shot may be indexed by a dominiant class or a series of several classes.

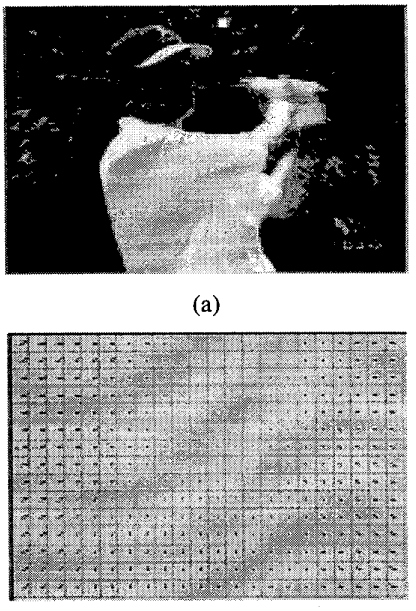
As stated earlier, it is useful to segment the video sequence into sub-shot of homogeneous camera motion for detailed manipulation, such as editing or retrieval. As results of the previous steps, some segments of the video detected as mulitple classes overlappedly and others may not labeled as any class. In the residual segment processing step, full sequence of the video is partitioned into sub-shot without any overlap through the merging of the residual segments into the neighbored classes according to the predefined rule derived from observation.

## 4. RESULTS

We validate the efficiency and robustness of the proposed method on a variety of real video sequences by comparing the detected camera motions with those determined from the ground truth built by manual observation. This section shows the experimental results on two real sport videos: *Golf* and *Soccer* sequences compressed in MPEG-1 format. Both of them contain very complicated motions including large object motions as well as various types of camera motions.

Figure 2 (a) and (b) show an example of the obtained MVF overlaid on the corresponding frame and its synthesized version by fitted affine model parameters, respectively. The estimated convergence point of the camera motion is also represented in Figure 2 (a) as a small white point.

To illustrate the procedure of the qualitative interpretation, we show an example of zoom detection applied to *Golf* sequence in Figure 3. The values for thresholds used in the method are easily found from the practical observations. Letting $T_{lin}$ and $T_{tr}$ denote the thresholds for the magnitude of the linear ( *div* , *rot* , *hyp* ) and translational motion parameters ( *pan* , *tilt* ) in the frame-level detection, respectively. They are set to the minimum values which make observable flow patterns to be induced by the parameters. Although these thresholds take fixed values ( $T_{lin} = 0.015$ and $T_{tr} = 1.0$ ) regardless of the types of video sequences, they turn to be quite stable.



(a)

(b)

**Figure 2.** A camera motion example. (a) The obtained MVF overlaid on the corresponding frame and (b) its synthesized version by fitted affine model parameters.

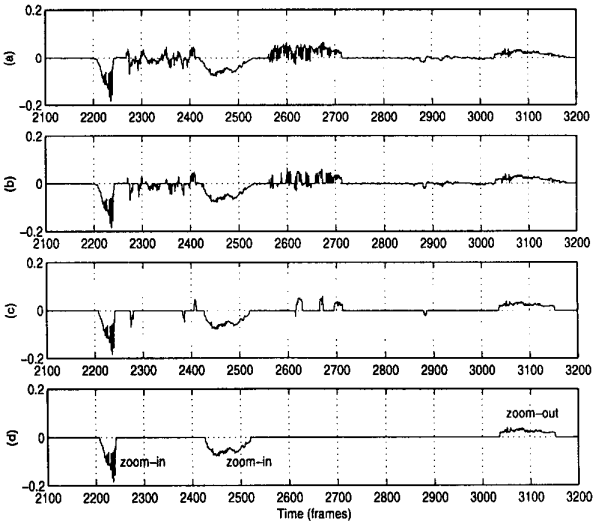



**Figure 3.** The zoom detection procedure applied to *Golf* sequence. The related parameter *div* is plotted against the frame number in each step. (a) temporal evolution of the estimated *div* from the MVF sequence (b) the result of the sign validation in the frame-level detection (c) the result of the magnitude thresholding in the frame-level detection (d) the result of the temporal thresholding in the segment-level detection and the detected segments for zoom-class.

Besides, in the segment-level detection, we set the threshold $T_{temp}$ as the minimum temporal length of each segment of homogeneous camera motion. It may be set differently according to a few types of video depending on the activity of camera operations. For example, we have set $T_{temp}$ to 30 frames for *Golf* sequence since it is composed of a series of relatively long camera operations, and to 15 frames for *Soccer* sequence in which active camera operations are rather short. After the detection of the segments, shown in Figure 3 (d), we can further identify the direction of the camera movement (zoom-in or zoom-out) by the sign of the parameters.

Figure 4 illustrates the sub-shot segmentation results for the *Golf* sequence. As shown in Figure 4, the sequence is completely partitioned into sub-shots without any overlap, each of which falls into one of six classes of camera motions. It turns out that the results are almost in accordance with those by manual segmentation.

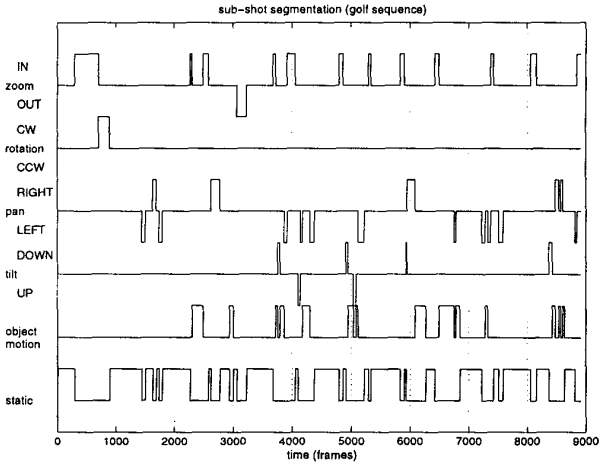



**Figure 4**. The results of the sub-shot segmentation for *Golf* sequence.

The performance of zoom detection was measured on two five-minute videos. The results are shown in Table 1, where the number of the segments classified as zoom is compared with the ground truth. It shows that the proposed method yields fairly good performance. For the *Soccer* video, two segments with zoom are missed due to unreliable motion vectors in slowly played scenes.

**Table 1**. Performance of zoom detection

| Sequences | # segments with zoom | Correctly detected | Missed | Falsely detected |
|---|---|---|---|---|
| Golf | 13 | 13 | 0 | 0 |
| Soccer | 19 | 17 | 2 | 2 |

# 5. CONCLUSION

In this paper, we described a novel framework for the analysis of camera motions to index MPEG compressed videos in content-based manners. It includes the construction of MVF from the motion vectors in MPEG stream and the characterization of camera operations through the proposed qualitative interpretation. We normalized the types of motion vectors and filtered out the noisy vectors to constitute a reliable MVF as an alternative to the optical flow at a low computational cost. Then, the proposed method utilizes such qualitative terms as the sign and magnitude of the affine model parameters estimated from the MVF and the continuity property of camera operation in temporal domain. Finally, it can classify the camera motion into six basic classes and partition the video sequence into sub-shots with homogeneous camera motion.

# 6. REFERENCES


[1] M. V. Srinivasan, S. Venkatesh, and R. Hosie, "Qualitative estimation of camera motion parameters from video sequences," *Pattern Recognit.*, vol. 30, no. 4, pp. 593-606, Apr. 1997.

[2] P. Bouthemy, M. Gelgon, and F. Ganansia, "A unified approach to shot change detection and camera motion characterization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 7, pp. 1030-1044, Oct. 1999.

[3] G. Sudhir and J. C. M. Lee, "Video annotation by motion interpretation using optical flow stream," *J. Vis. Commun. Image Represent.*, vol. 4, pp. 354-368, Dec. 1996.

[4] W. Xiong and J. C. M. Lee, "Efficient scene change detection and camera motion annotation for video classification," *Comput. Vision Image Understanding,* vol 71, no. 2, pp. 166-181, Aug. 1998.

[5] N. V. Patel and I. K. Sethi, "Video shot detection and characterization for video databases," *Pattern Recognit.*, vol. 30, no. 4, pp. 607-625, Apr. 1997.

[6] A. Akutsu, Y. Tonomura, H. Hashimoto, and Y. Ohba, "Video indexing using motion vectors," in *Proc. SPIE VCIP '92*, Vol. 1818, Nov. 1992, pp. 1522-1530.

[7] P. Joly and H. K. Kim, "Efficient automatic analysis of camera work and microsegmentation of video using spatiotemporal images," *Signal Process.: Image Commun.*, vol. 8, pp. 295-307, 1996.

[8] J.-G. Kim, K.-W. Lee, J. Kim, and H.-M. Kim, "Extraction of moving objects from MPEG-compressed video for object-based indexing," in *Proc. 1st European Workshop on CBMI*, Oct. 1999, pp. 131-139.

[9] E. Franeois and P. Bouthemy, "Derivation of qualitative information in motion analysis," *Image Vis. Computing*, vol. 8, no. 4, pp. 279-287, Nov. 1990.