Shot-Level Camera Motion Estimation Based on a Parametric Model

Ling-Yu Duan[1,3], Jinqiao Wang[1,2], Yantao Zheng[1], Changsheng Xu[1], Qi Tian[1], Jesse S. Jin[3], Hanqing Lu[2]

[1]Institute for Infocomm Research, 21 Heng Mui Keng Terrace, Singapore 119613
[2]National Lab of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100080, China
[3]The School of Design, Communication and Information Technology, University of Newcastle, NSW 2308, Australia

ABSTRACT

This paper presents a system for shot-level camera motion analysis via the estimation of global motion from image sequences. An affine model is used to parameterize the global motion resulting from motion of a camera (e.g. pan, zoom, tilt, rotation). A singular value decomposition (SVD) based gradient descent is executed to minimize the prediction error. To assure the convergence of the gradient descent algorithm, we employ a modified n-step search matching to estimate initial translation and apply the gradient descent over a pyramid of input images. *M*-estimator is used to remove the influence of outliers derived from local object motion. The recovered parameters can be tied to the qualitative analysis of physically meaningful camera motion by simple transformation. Towards the shot-level characterization of camera motion, several simple rules are carried out to condense the series of transformed parameters extracted from a whole shot. Those rules mainly concern the magnitudes and temporal persistency of each meaningful parameter. We submit seven runs to TRECVID 2005. Their run-ids are D_0, D_1, …, D_6. They use the same set of algorithms with different parameters tuned for the rules described above. Based on the results, we have the findings: 1) the design of rules and related parameters' tuning is significant but difficult to cover many cases from diverse scene contents, 2) various sizes of camera shots (e.g. close-up, medium shot, long shot) makes it infeasible to secure a proper parametric assumption of camera motion as the global motion in large amounts of video data, and 3) the context dependent training could improve application performance wherein the context information may include video genre, shot size, and shot categories according to a priori knowledge of a scene.

1. INTRODUCTION

Motion characterization plays a critical role in video indexing. Camera movements and mobile objects are two main sources of dynamic information contained in the video. Motion content can be used as a powerful cue for structuring video data, similarity-based video retrieval, and video abstraction. As motion features are of key significance in video indexing, MPEG-7 has selected a set of motion descriptors including motion activity, camera movement, mosaic, trajectory, and parametric motion [1]. Shot-level camera motion estimation is one of five tasks in TRECVID 2005. This paper discusses our system on performing camera motion analysis on a large news video corpus used in TRECVID 2005 evaluation.

Generally speaking, motion in a sequence of images results from motion of a camera and from displacement of individual objects. The former is often referred to as global motion and the latter as local motion. Various global motion models have been used: six-parameter affine model, eight-parameter quadratic model, eight-parameter perspective model, etc. As the motion model cannot account for local motion, local object motion may create outliers and therefore bias the estimation of global motion parameters. *M*-estimates are usually the most relevant class of model fitting, which is insensitive to small departures from the idealized assumptions for which the estimator is optimized. We use *M*-estimator to remove the influence of outliers derived from local object motion.

The recovery of a linear or non-linear parametric motion model is the problem of data modeling. That is, given a set of observations, one often wants to condense and summarize the data by fitting it to a “model” that depends on adjustable parameters. A *figure-of-merit function* is designed to measure the agreement between the data and the model with a particular choice of parameters. The parameters of the model are then adjusted to achieve a minimum in the merit function, yielding *best-fit parameters*. The adjustment process is thus a problem in minimization in many dimensions [2]. When the model (e.g., eight-parameter perspective model) depends nonlinearly on the set of unknown parameters, the minimization must proceed iteratively. Given trial values for the parameters, a gradient descent procedure is repeated to improve the trial solution. At each iteration step, singular value decomposition (SVD) is employed to calculate the parameter increments that, added to the current approximation, give the next approximation [2].

It is not uncommon in fitting data to discover that the merit function is not uni-modal, with a single minimum. In order to assure the convergence of the gradient descent procedure and a much better fit, we employ a modified n-step search matching to estimate initial translation and apply the gradient descent over a pyramid of input images. The gradient descent is carried out at each level of the pyramid starting from the coarsest level. The gradient descent at the current level is initialized with the parameters projected from the last level. The motion parameters from one level are projected onto the next one simply by multiplying or dividing a value (say 2). Due to the coarse initial estimation and the hierarchical implementation, the method is very fast.

In order to associate those parameters with the physically meaningful camera motion patterns (i.e. Pan, Tilt, Zoom, and Rotation), we utilize several simple transformations introduced in [3] for qualitative interpretation of 2D velocity field and 3D motion parameters. The transformations are applied to the series of motion parameters calculated between pairs of consecutive frames within a shot. The resulting series quantitatively indicating Pan, Tilt, and Zoom is required to be appropriately condensed for modeling human perception of camera motion at the shot level. It is hard to extract a uniform set of features to characterize the patterns of time series aiming at shot-level camera motion. Finally, a set of simple rules is experimentally determined to accomplish the mapping from frame-level to shot-level. Since there is no common set of training data, the rules are designed according to our experience on a limited set of labeled data (we have manually labeled about 24 video files from the training set).

Our work is closely related to that of [3] and [4].

2. OVERVIEW OF THE SYSTEM COMPONENTS

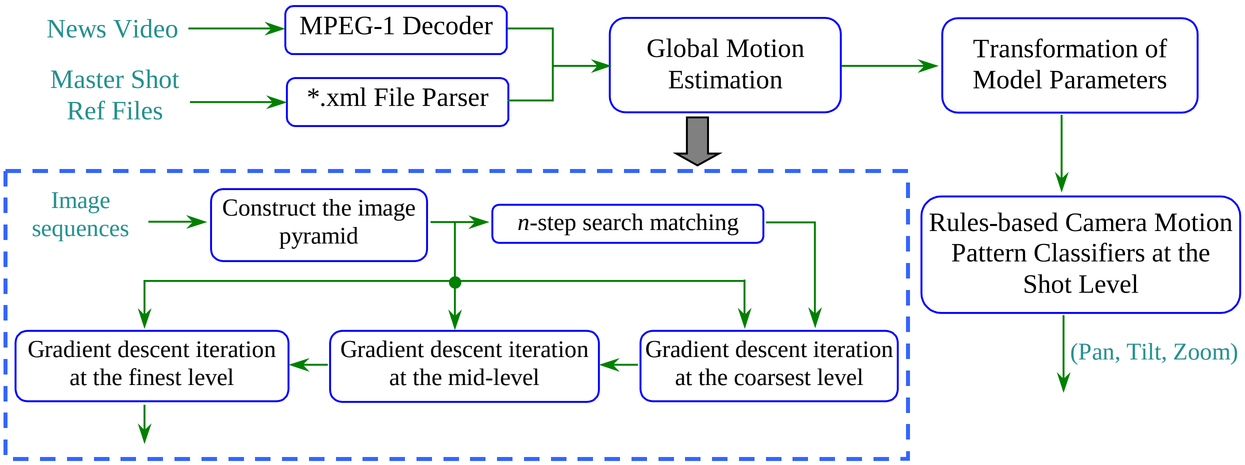
News Video
MPEG-1 Decoder
Master Shot Ref Files
*.xml File Parser
Global Motion Estimation
Transformation of Model Parameters
Image sequences
Construct the image pyramid
n-step search matching
Rules-based Camera Motion Pattern Classifiers at the Shot Level
Gradient descent iteration at the finest level
Gradient descent iteration at the mid-level
Gradient descent iteration at the coarsest level
(Pan, Tilt, Zoom)

Fig.1. Block diagram of the proposed system with a three-level hierarchical implementation

Fig.1 illustrates our system with a three-level hierarchical implementation. Three major modules are included: global motion estimation, transformation of model parameters, and rules-based camera motion pattern classifiers at the shot level. An MPEG-1 decoder is developed using MS DirectShow. A *.xml file parser is programmed to extract the shot boundary information from the master shot references by [5].

2.1 Global Motion Estimation

2.1.1 Camera Motion Model

We consider eight-parameter perspective motion model defined as follows:

$$\begin{aligned} x_i^{'} &= (a_0 + a_2 x_i + a_3 y_i)/(a_6 x_i + a_7 y_i + 1) \\ y_i^{'} &= (a_1 + a_4 x_i + a_5 y_i)/(a_6 x_i + a_7 y_i + 1) \end{aligned} \tag{1}$$

where $(a_0, \ldots, a_7)$ are the motion parameters, (x_i, y_i) denotes the spatial coordinates of the *i*th pixel in the current frame and $(x_i^{'}, y_i^{'})$ denotes the coordinates of the corresponding pixel in the previous frame. Various motion models can be derived from this model. For example, in the case of $a_6 = a_7 = 0$, it is reduced to an affine model; in the case of $a_2 = a_5, a_3 = -a_4, a_6 = a_7 = 0$, it is reduced to a translation-zoom-rotation model.

2.1.2 Gradient Descent

The global motion estimation is designed to achieve a minimum of the sum of squared differences between the current frame and the motion compensated previous frame. The model to be fitted is

$$\rho = \rho(x_i, y_i; \mathrm{a}) \tag{2}$$

and the χ^2 merit function is

$$\chi^2(a) = \sum_{i=1}^{N} \left[\frac{\rho_i - \rho(x_i, y_i; \mathrm{a})}{\sigma_i} \right] \tag{3}$$

where N is the number of pixels within image boundaries, a denotes the motion parameters $(a_0, a_1, \ldots, a_7)$, σ_i denotes the known standard deviations of each data point ρ_i.

It is well known that, by taking some particular point $\mathbf{p}$ as the origin of the coordinate system with coordinates $\mathbf{z}$, any function f can be approximated by its Taylor series

$$f(\mathbf{z}) = f(\mathbf{P}) + \sum_i \frac{\partial f}{\partial z_i} z_i + \frac{1}{2} \sum_{i,j} \frac{\partial^2 f}{\partial z_i \partial z_j} z_i z_j + \cdots \tag{4}$$

$$\approx c - \mathbf{b} \cdot \mathrm{z} + \frac{1}{2} \mathrm{z} \cdot \mathbf{A} \cdot \mathrm{z} \tag{5}$$

where

$$c \equiv f(\mathrm{P}) \qquad b \equiv -\nabla f \big|\mathbf{p} \qquad [\mathbf{A}]_{ij} = \frac{\partial^2 f}{\partial z_i \partial z_j} \big|\mathbf{p} \tag{6}$$

The matrix $\mathbf{A}$ whose components are the second partial derivates matrix of the function is called the *Hessian matrix* of the function at $\mathbf{p}$ [2].

In the approximation of (5), the gradient of f is easily calculated as

$$\nabla f = \mathbf{A} \cdot \mathrm{z} - \mathbf{b} \tag{7}$$

In Newton's method we set $\nabla f = 0$ to determine the next iteration point:

$$\mathrm{z} - \mathrm{z}_i = -\mathbf{A}^{-1} \cdot \nabla f(z_i) \tag{8}$$

Instead of calculating the left-hand side of (8), one solves the set of linear equations

$$\mathbf{A} \cdot (\mathrm{z} - z_i) = -\nabla f(z_i) \tag{9}$$

The gradient of χ^2 with respect to the parameters a, which will be zero at the χ^2 minimum, has components

$$\frac{\partial \chi^2}{\partial a_k} = -2\sum_{i=1}^{N} \frac{[\rho_i - \rho(x_i, y_i; \mathrm{a})]}{\sigma_i^2} \frac{\partial \rho(x_i, y_i; \mathrm{a})}{\partial a_k} \qquad k = 0,1,\ldots,7 \tag{10}$$

Taking an additional partial derivative gives

$$\frac{\partial^2 \chi^2}{\partial a_k \partial a_l} = 2\sum_{i=1}^{N} \frac{1}{\sigma_i^2} \left[\frac{\partial \rho(x_i, y_i; \mathrm{a})}{\partial a_k} \frac{\partial \rho(x_i, y_i; \mathrm{a})}{\partial a_l} - [\rho_i - \rho(x_i, y_i; \mathrm{a})] \frac{\partial^2 \rho(x_i, y_i; \mathrm{a})}{\partial a_l \partial a_k} \right] \tag{11}$$

It is conventional to remove the factors of 2 by defining

$$\beta_k \equiv -\frac{1}{2}\frac{\partial \chi^2}{\partial a_k} \qquad\qquad a_{kl} \equiv \frac{1}{2}\frac{\partial^2 \chi^2}{\partial a_k \partial a_l} \tag{12}$$

making $[a] = \frac{1}{2}\mathbf{A}$ in equation (6), in terms of which that equation can be rewritten as the set of linear equations

$$\sum_{l=1}^{N} a_{kl} \delta a_l = \beta_k \tag{13}$$

In the context of least squares, the matrix $[a]$, equal to one-half times the Hessian matrix, is usually called the *curvature matrix*.

SVD is finally utilized to solve the overdetermined set of linear equations (13).

2.1.3 Initial Estimation and Robust Estimation

In terms of minimization or maximization of functions, an extremum can be either global or local. Finding a global extremum is, in general, a very difficult problem. In order to select a suitable start point $\mathbf{P}$ to initiate the iteration process describe above, it is necessary to perform initial motion modal estimation.

A three-step search (TSS) algorithm [6] is utilized to accomplish block-based searching at the coarsest level of the pyramid of the image. The search range is set to ±4, ±2, and ±1 in the first, second, and third step. The knowledge of the displacement (motion vector) of each block is used to estimate the translation components (a_0, a_1) in equation (1). TSS is used for its simplicity and also robust and near optimal performance. To further improve robustness and efficiency, a hierarchical scheme is implemented. A three-level pyramid of the image is built by using a three-tap filter with coefficients $\left[1/4 \quad 1/2 \quad 1/4\right]$. The estimated motion parameters at a give level is projected onto the level of higher resolution and serves as an initial value to compute some more increments.

M-estimator is employed to augment robustness against outliers. Tukey's biweight function is selected as the weighting function:

$$\psi(z) = \begin{cases} z(1 - z^2/C^2)^2 & |z| < C \\ 0 & |z| > C \end{cases} \tag{14}$$

The more deviant point, the greater is the weight. Very deviant points (the true outliers) are not counted at all in the estimation of the parameters. In practice, we take $C = 7$.

2.2 Transformation of Model Parameters

In [3], any vector field is approximated by a linear combination of a divergent field, a rotation field, and two hyperbolic fields. The relationship between motion model parameters and symbol-level interpretation is established:

$$\begin{aligned} Pan &= a_0 & Tilt &= a_1 \\ Zoom &= \frac{1}{2}(a_2 + a_5) & Rotation &= \frac{1}{2}(a_3 - a_4) \\ Hyp1 &= \frac{1}{2}(a_5 - a_2) & Hyp2 &= \frac{1}{2}(a_3 + a_4) \end{aligned} \tag{15}$$

With equation (15), a series of $(Pan, Tilt, Zoom, Rotation, Hyp1, Hyp2)$ are derived for each video shot. In the context of TRECVID'05 low-level feature extraction task, we concern $(Pan, Tilt, Zoom)$.

To determine the presence of Pan, Tilt, or Zoom, we have to decide whether those estimated values $(Pan, Tilt, Zoom)$ are significant or not. The reasons are twofold: 1) due to noise, estimator errors, and the use of an approximate model, these quantities cannot be strictly equal to zero even if it should be the case; 2) human perception of global motion is related to the association of magnitude and duration over the temporal dimension. A likelihood ratio test has been proposed in [3] to avoid the delicate and unstable threshold selection. Despite the claimed better control of thresholds on a likelihood ratio, our implementation resorts to direct thresholding since the shot-level decision requires magnitudes rather than symbols only. Moreover, the additional computation introduced by likelihood ratio tests is considerable for processing a large video corpus.

2.3 Rule-based Camera Motion Pattern Classifiers

Several rules have been discussed in [7] to deal with the ambiguities of creating the truth data. The subset used for evaluation is a collection of samples culled from the test set by dropping those unreliable or uncertain samples, which are derived from handheld camera, lack of background, complexity of motion in multiple dimensions at once, blending in of digital effects, etc.

We resort to a set of simple rules as follows:

1) Accumulate the duration of estimated pan rate with the magnitude above a threshold Mag_{pan} when the pan is in a consistent direction and reset the duration to zero when the pan direction is changed. PAN is declared once the persistent process's duration is more than a threshold $Duration_{pan}$. K Instants with a threshold less than Mag_{pan} are allowed for each accumulation process. $K = 2$.
2) One special kind of PAN is also declared that one can easily perceive: larger rate (greater than 3 times Mag_{pan}) with short duration (longer than $0.3 \times Duration_{pan}$).
3) Rules 1 and 2 do not count the instants when the magnitude of zoom rate is greater than L times the magnitude of pan rate. $L = 4$.
4) Accumulate the duration of estimated zoom rate with the magnitude above a threshold Mag_{zoom} for two directions Zoom-in and Zoom-Out, respectively. If any duration is greater than $Duration_{zoom}$, then ZOOM is declared.
5) Similarly, Rules 1 and 2 are applied to detect TILT by simply replacing the pan rate with the tilt rate. Two thresholds Mag_{tilt} and $Duration_{tilt}$ are predefined.

3. EXPERIMENTAL RESULTS

We have manually labeled 24 video files in training set as the ground truth. Accordingly, seven groups of parameters are adjusted to favor different recall/precision settings over the training data. Table I lists the parameters corresponding to our submitted seven runs $D_0, D_1, \ldots, D_6$. Table II lists the results. The best performances in *F1* measure have been highlighted in Table II for PAN, TILT, ZOOM, and MEAN, respectively.

TABLE I
PARAMETERS OF SEVEN RUNS

	Mag_{pan}	$Duration_{pan}$	Mag_{zoom}	$Duration_{zoom}$	Mag_{tilt}	$Duration_{tilt}$
D_0	45	38	48	40	41	30
D_1	20	30	41	35	30	35
D_2	60	21	35	45	20	40
D_3	40	35	60	55	40	45
D_4	40	25	40	40	55	35
D_5	48.2	24	48.2	31	39	33
D_6	46.8	32	48.4	35	50	35

TABLE II
RESULTS OF SEVEN RUNS

		D_0	D_1	D_2	D_3	D_4	D_5	D_6
PAN	*Recall*	0.724	0.874	0.838	0.707	0.840	0.826	0.770
	Precision	0.998	0.961	0.970	0.995	0.970	0.976	0.991
	F1	0.839	0.915	0.899	0.826	0.900	0.894	0.867
TILT	*Recall*	0.510	0.533	0.548	0.481	0.543	0.490	0.424
	Precision	0.991	0.982	0.966	0.990	1.00	0.99	0.989
	F1	0.673	0.691	0.699	0.647	0.704	0.656	0.593
ZOOM	*Recall*	0.695	0.761	0.689	0.560	0.716	0.771	0.738
	Precision	0.978	0.973	0.981	0.990	0.973	0.970	0.977
	F1	0.813	0.854	0.809	0.715	0.825	0.859	0.841
MEAN	*Recall*	0.643	0.723	0.692	0.583	0.700	0.696	0.644
	Precision	0.989	0.972	0.972	0.992	0.981	0.979	0.986
	F1	0.779	0.829	0.808	0.734	0.817	0.814	0.779

The measures are defined as:

$$\begin{aligned}
&\text{Precision} = TurePositive/(TruePositive + FalsePositive) \\
&\text{Recall} = TruePositive/(TurePositive + FalseNegative) \\
&\text{Mean Precision} = \left(\text{Precision}_{\text{pan}} + \text{Precision}_{\text{zoom}} + \text{Precision}_{\text{tilt}}\right)/3 \\
&\text{Mean Recall} = \left(\text{Recall}_{\text{pan}} + \text{Recall}_{\text{zoom}} + \text{Recall}_{\text{tilt}}\right)/3 \\
&F1 = 2 \times \text{Recall} \times \text{Precision}/(\text{Recall} + \text{Precision})
\end{aligned} \tag{16}$$

4. CONCLUSIONS

We have introduced our work on TRECVID'05 low-level feature extraction task. A parametric approach is employed to discover camera motion patterns. Compared with nonparametric methods, parametric models have the advantage of simplicity. That is, there is no need to manually label many samples of motion field for supervised training. Although the achieved performance of 0.704 – 0.915 is not bad, it is worthwhile to mention that the subset used for evaluation is not a random sample from the whole test set. Quite a number of ambiguous samples have been dropped. In real video processing, one doesn't know which shot is reliable for camera motion estimation. The evaluation results thus cannot well reflect the performance in practice.

Shot-level camera motion pattern is considered as a low-level feature in TRECVID'05. According to our experiences learned from manual labeling, the precise decision on camera motion is related to context such as shot sizes, shot categories, dynamic scene content, etc. The assumption of global motion as camera motion is sometimes impractical. Camera motion is not a complete low-level feature. A robust solution has to rely on certain context. Learning is an approach to incorporate context information. Moreover, the mapping from frame-level to shot-level such as the selection of thresholds and durations is not an easy job. It is difficult to cover all cases derived from dynamic scene contents by summarizing a limited set of rules. It may be necessary to establish a common set of labeled data to seek the relationship between feature-level temporal behaviors and human perception in terms of shot-level camera motion.

5. REFERENCES

[1] S. Jeannin and A. Divakaran, "MPEG-7 visual motion descriptor," *IEEE Tran. Circuits and Systems for Video Technology*, vol.11, no.6, pp. 720-724, 2001.

[2] W.H. Press, B.P. Flannery, S. A. Teukolsky, and W.T. Vetterling, *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge, U.K.: Cambridge Univ. Press, 1988, pp.59-70, pp.656-706.

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

[4] F. Dufaux and J. Konrad, "Efficient, robust and fast global motion estimation for video coding," *IEEE Tran. Image Processing*, vol.9, no.3, pp.497-501, Mar. 2000.

[5] C. Petersohn, "Fraunhofer HHI at TRECVID 2004: shot boundary detection system," in *Proc. of TREC Video Retrieval Evaluation Online*, TRECVID 2004,
[Online] http://www-nlpir.nist.gov/projects/tvpubs/tvpapers04/fraunhofer.pdf.

[6] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishiguro, "Motion compensated interframe coding of video conference," in *Proc. Nat. Telecommun., Conf.*, New Orleans, LA, Dec. 1981, pp.G5.3.1-G.5.3.5.

[7] Rules for annotating camera motion,
[Online] http://cio.nist.gov/esd/emaildir/lists/trecvid2005/msg00095.html.