FEATURE-BASED GLOBAL MOTION ESTIMATION USING THE HELMHOLTZ PRINCIPLE

Michael Tok, Alexander Glantz, Andreas Krutz, and Thomas Sikora

Communication Systems Group
Technische Universität Berlin
Berlin, Germany

ABSTRACT

Global motion estimation is an important task for various video processing techniques. The estimation itself has to be robust in presence of arbitrarily moving foreground objects. For that task, two different kinds of estimation methods exist. On the one hand, pixel-based approaches deliver more precise results and work more robust on video sequences with foreground objects. On the other hand, when working on encoded video streams, block-based methods can be used for a much faster but often less precise estimation. We propose a two step estimation method based on the determination and tracking of feature points of video frames and robust motion model estimation using the Helmholtz principle. Therefore, good trackable features are detected and tracked in video sequences. Subsequently, a perspective motion model is derived from the resulting correspondencies by removing feature pairs not belonging to global motion.

Index Terms— Global motion estimation, Helmholtz Tradeoff Estimator, feature tracking, robust regression

1. INTRODUCTION

Motion estimation is a fundamental problem in video analysis and coding tasks. The knowledge of local motion for example is used in video compression to remove temporal correlation. A more complex description of motion in video sequences is the global motion represented by a higher-order motion model.

In this paper, global motion means the background transformation between two adjacent frames of a video sequence. With the knowledge of such a transformation model, various problems concerning video processing, for example background sprite generation for video summarization or camera motion characterization for classification can be solved. Irani et al. describe in [1] how background sprites generated by global motion estimation (GME) can be used to get an overview of a video sequence.

Approaches for improving video coding, based on or assisted by global motion models also already exist. Glantz et al. for example describe a post filter based on global motion temporal filtering [2]. This filter uses temporal correlation of adjacent frames for blocking artifact reduction. Therefore, highly precise and robust GME is required.

Generally, GME techniques are separated into two classes. Pixel-based approaches, working directly on pixel data are said to deliver more accurate results than motion-vector-based ones. However, heavy computation load is their main drawback in most cases. A well-known pixel-based approach works on error minimization by gradient descent. Dufaux and Konrad use this method in [5] to get a perspective motion model directly in the pixel domain. In block-based approaches, working on motion vector fields, detection and removal of outliers not belonging to global motion (local motion models and misestimations) are the key tasks of robust regression methods. Smolić et al. for example use a robust M-estimator for such GME [3]. In [4], we presented a GME method that estimates on regular block structures with motion vectors as existing in H.264/AVC video streams.

We present a two step hybrid approach which first estimates local motion models on good trackable features of a video frame and then estimates a global motion model out of these with the use of a highly robust regression method based on the Helmholtz principle.

The rest of this paper is organized as follows. Section 2 shortly describes how feature correspondencies for the GME are generated. In Section 3 the outlier rejection method and the final motion model calculation is described. Section 4 presents and discusses the results. Finally, Section 5 summarizes this paper.

2. FEATURE TRACKING

Motion vectors in regular block structures are generated for every block, whether they contribute to the global motion or not. When used in GME, this severely affects the quality of the estimation. Another quality limiting factor of block motion data is the motion information resolution which is often limited to half or quarter-pel. Hence, when choosing only good trackable features in a video frame and tracking them with an accuracy much higher than quarter-pel, better GME

results are possible. Therefore, for motion vector field generation a KLT feature tracker as described in [6] is used.

Feature tracking means the search for the correspondence of a feature point $I(\mathbf{x})$ of a given frame I in a consecutive frame J. The result is a displacement (or motion) vector $\mathbf{d}$ for each feature point, describing its linear translational motion. Such a relationship of two corresponding features can be defined as

$$J(\mathbf{x}) \approx I(\mathbf{x} - \mathbf{d}). \tag{1}$$

Tracking a feature can be done by gradient descent. For a feature window W of a feature point, the mean squared error between two frames is minimized by finding an optimal displacement vector $\mathbf{d}$:

$$\epsilon = \int_W \left[I(\mathbf{x} - \mathbf{d}) - J(\mathbf{x})\right]^2 \, d\mathbf{x}. \tag{2}$$

As the displacement is expected to be relatively small, $I(\mathbf{x} - \mathbf{d})$ can be approximated by a Taylor series of first degree with the two dimensional gradient vector $\mathbf{g}$:

$$I(\mathbf{x} - \mathbf{d}) = I(\mathbf{x}) - \mathbf{g} \cdot \mathbf{d}. \tag{3}$$

Then, by setting $h = I(\mathbf{x}) - J(\mathbf{x})$, (2) can be written as

$$\epsilon = \int_W \left(h - \mathbf{g} \cdot \mathbf{d}\right)^2 \, d\mathbf{x}. \tag{4}$$

This equation can be interpreted as a quadratic error function $\epsilon(\mathbf{d})$. The minimum of this function can be found by differentiation with respect to $\mathbf{d}$ and setting the result to zero:

$$\int_W (h - \mathbf{g} \cdot \mathbf{d})\mathbf{g} \; dA = 0 \tag{5}$$

which leads to an easy to solve equation system of two scalar equations with two unknowns.

3. MODEL ESTIMATION

A lot of features that are selected and tracked belong to foreground objects. These features lead to misestimation when they are not rejected. Rejecting such outliers is the task of robust estimators. We use a modified version of the Helmholtz Tradeoff Estimator first described in [7]. It evaluates randomly selected subsets from a dataset. The amount m of needed subsets can be calculated by

$$m = \frac{\log(1 - P)}{\log(1 - (1 - \epsilon)^p)}, \tag{6}$$

where P is the desired probability of finding a good estimation in an environment with an outlier percentage of at most ϵ for a model with p parameters. This means that for $P = 95\%$ and $\epsilon = 80\%$, more than 1,170,000 subsets have to be evaluated, when a perspective eight parameter model is to be estimated out of diplacement vectors. Thus, a direct usage of the original Helmholtz Tradeoff Estimator is too complex.

When having n feature positions and their motion vectors, for each subset s two correspondencies $(x, y) \leftrightarrow (x', y')$ (meaning $\mathbf{x} \leftrightarrow (\mathbf{x} + \mathbf{d})$) are taken and a four parameter motion model only containing translation, rotation and scaling is calculated:

$$\begin{pmatrix} x_{1,s} & y_{1,s} & 1 & 0 \\ y_{1,s} & -x_{1,s} & 0 & 1 \\ x_{2,s} & y_{2,s} & 1 & 0 \\ y_{2,s} & -x_{2,s} & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} m_{0,s} \\ m_{1,s} \\ m_{2,s} \\ m_{3,s} \end{pmatrix} = \begin{pmatrix} x'_{1,s} \\ y'_{1,s} \\ x'_{2,s} \\ y'_{2,s} \end{pmatrix}, \tag{7}$$

so that each subset s gets a homography

$$H_s = \begin{pmatrix} m_{0,s} & m_{1,s} & m_{2,s} \\ -m_{1,s} & m_{0,s} & m_{3,s} \\ 0 & 0 & 1 \end{pmatrix}. \tag{8}$$

That way, the parameter complexity p is reduced from 8 to 4 which means that only 1,874 subsets are needed for the case described above. Now, for every subset all feature positions $\mathbf{x}_i = (x_i, y_i)$ selected and tracked by KLT are transformed by H_s to $\tilde{\mathbf{x}}_{i,s} = (\tilde{x}_{i,s}, \tilde{y}_{i,s})$ and the λ-th percentile $v_{\lambda,s}$ (with $\lambda = 1 - \epsilon$) of the squared error distances

$$r_{i,s}^2 = \|\mathbf{x}_i + \mathbf{d}_i - \tilde{\mathbf{x}}_{i,s}\|^2 \tag{9}$$

is taken to estimate a subset standard deviation

$$\hat{\sigma}_s \;\; = \;\; \frac{1}{\Phi^{-1}(0.75)} \cdot \left(1 + \frac{5}{n - p}\right) \cdot v_{\lambda,s}. \tag{10}$$

Thus, every feature correspondency can be classified by its estimation error related to H_s:

$$w_{i,s} \;\; = \;\; \begin{cases} 1, & \text{if } |r_{i,s}/\hat{\sigma}_s| \leq 2.5 \\ 0, & \text{else} \end{cases}, \tag{11}$$

where $w_{i,s} = 1$ classifies a correspondence as inlier. Having the amount of inliers

$$I_s = \sum_{i=1}^{n} w_{i,s} \tag{12}$$

and their estimation error standard deviation

$$\sigma'_s = \sqrt{\sum_{k \in \text{Inliers}} \frac{(r_{k,s} - \mu_s)^2}{I_s}} \tag{13}$$

with μ_s as inlier mean error of a set, a rating value

$$\Phi_s = \frac{I_s}{\sigma'_s} \tag{14}$$

can be defined for every subset. The selection of $k = I_s$ remaining inliers corresponding with the highest rating Φ_s should describe a global motion model best and is taken to

get a final perspective motion model with eight parameters by solving an overdetermined equation system of the form

$$A \cdot \begin{pmatrix} m_0 \\ m_1 \\ \vdots \\ m_6 \\ m_7 \end{pmatrix} = \begin{pmatrix} x_1^{'} \\ y_1^{'} \\ \vdots \\ x_k^{'} \\ y_k^{'} \end{pmatrix}, \tag{15}$$

with the perspective design matrix

$$A = \begin{pmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1 x_1^{'} & -y_1 x_1^{'} \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -x_1 y_1^{'} & -y_1 y_1^{'} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n & y_n & 1 & 0 & 0 & 0 & -x_n x_n^{'} & -y_n x_n^{'} \\ 0 & 0 & 0 & x_n & y_n & 1 & -x_n y_n^{'} & -y_n y_n^{'} \end{pmatrix} \tag{16}$$

by calculating the pseudo-inverse of A. The final result is a precise perspective motion model. Figure 1 illustrates the process of estimating a global motion model from feature correspondencies.

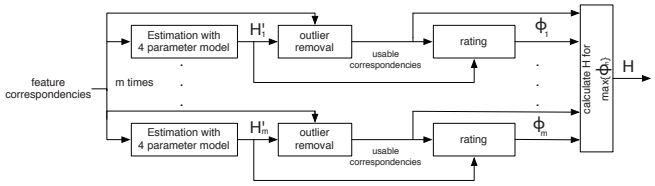


Fig. 1. Global motion estimation on feature correspondencies

4. EXPERIMENTAL EVALUATION

Twelve test sequences with varying properties as resolution, frame rate and existence and size of foreground objects have been selected for evaluation of the proposed method. Table 1 shows the used sequences. For comparison with the method discussed in this paper, we used the algorithm proposed in [4] on motion vector fields created by encoding all sequences with a GOP structure of IPPP... and a QP of 4 with KTA 2.4 reference software. For the method described in this paper we always selected 400 features with a minimum euclidean distance of 10 pixels and a feature window size of 7×7. To measure the motion estimation quality we warped each frame of a given test sequence onto its successive one with the use of the estimated parameters. The frame warping is done with bicubic spline interpolation of degree three. Background PSNR (BPSNR) values between the warped frames and their correspondencies have been calculated using manually segmented ground truth masks of the background regions. Table 2 shows BPSNR values for the uncompensated case (no motion compensation), for the method on motion vector fields and the method described in this paper[1]. The last column shows the gain of the feature-based method for each sequence in comparison to the motion-vector-based one.

The results show that for almost all resolutions, quality improvements are possible, irrespective of existing foreground objects. As the camera motion in the *Monaco* sequence is much smaller than quarter-pel, the gain resulting from the highly precise feature motion estimation is even about 1.66dB. The rotation in the *Blue Sky* sequence, being much faster than sub-pel is estimated exactly by both approaches (motion-vector-based and feature-based) so that there is no quality gain achievable. But this result also shows that the new method does not perform less accurate. A gain of 0.96dB for the *Mountain* sequence proves, that even in video sequences with moving foreground objects the new feature-based method can outperform the former block-based one. Figure 2 shows exemplarily BPSNR-curves for the *Monaco* and *Mountain* sequence. For comparison, uncompensated values are shown as well as results for block-based and feature-based GME. As the motion-vector-based method can use existing vector fields, while the new one needs to calculate feature correspondencies first, our new method needs about 69% more runtime in average. Notice, however, that both methods are extremely fast in comparison to the method proposed in [5].

Sequence name	Size			fps	Frames
Mountain	352	x	192	25	130
Stefan	352	x	240	30	300
Allstars (small)	352	x	288	25	250
Biathlon	352	x	288	25	200
Monaco	352	x	288	25	150
Race	544	x	336	25	100
Flowervase	832	x	480	30	300
Allstars (big)	704	x	576	25	250
Room 3D	720	x	576	25	60
Schloss	720	x	576	25	120
Penguins	1280	x	720	25	349
Blue Sky	1920	x	1080	25	217

Table 1. Overview of the used test sequences

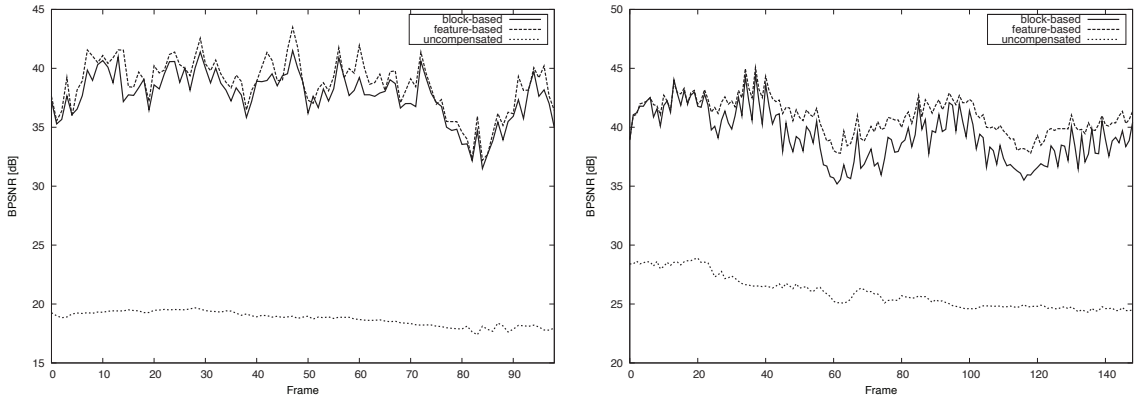
5. SUMMARY

We presented a new method for pixel-based global motion estimation working on features that are selected by their trackability and then tracked with sub-pixel accuracy using a local gradient descent approach as done by the KLT feature tracker. The outlier rejection process for getting a perspective model that describes the background transformation of two adjacent frames has been introduced. The comparison to the block-based method shows quality improvements up to 1.66dB.

[1]Estimation error videos and further BPSNR-curves can be found at http://www.nue.tu-berlin.de/research/featgme

Sequence	uncompensated [dB]	block-based[1)] [dB]	feature-based[2)] [dB]	$\Delta_{1) \text{ to } 2)}$ [dB]
Mountain	18.81	37.74	38.70	+0.96
Stefan	17.74	30.39	30.75	+0.36
Allstars (small)	30.71	41.95	42.42	+0.47
Biathlon	24.21	39.05	39.22	+0.17
Monaco	26.03	39.28	40.94	**+1.66**
Race	20.66	37.09	37.28	+0.19
Flowervase	32.77	36.57	36.65	+0.08
Allstars (big)	29.27	39.70	40.41	+0.71
Room 3D	17.12	35.50	36.27	+0.77
Schloss	21.89	37.65	37.66	+0.01
Penguins	21.82	32.40	32.63	+0.23
Blue Sky	17.67	39.43	39.43	±0.00

Table 2. This table shows the results in terms of BPSNR-values [dB] for the uncompensated case, the block-based approach described in [4], the method described in this paper and the difference between these two methods.



(a) Mountain (b) Monaco

Fig. 2. BPSNR-values for two selected test sequence comparing the block-based method with the feature-based one

6. REFERENCES

[1] M. Irani and P. Anandan, "Video indexing based on mosaic representations," *Proceedings of the IEEE*, vol. 86, no. 5, pp. 905 –921, may. 1998.

[2] A. Glantz, A. Krutz, M. Haller, and T. Sikora, "Video coding using global motion temporal filtering," in *Image Processing (ICIP), 2009 16th IEEE International Conference on*, nov. 2009, pp. 1053 –1056.

[3] A. Smolic, M. Hoeynck, and J.-R. Ohm, "Low-complexity global motion estimation from P-frame motion vectors for MPEG-7 applications," *Image Processing, 2000. Proceedings. 2000 International Conference on*, vol. 2, pp. 271 –274 vol.2, Sept. 2000.

[4] M. Tok, A. Glantz, M. G. Arvanitidou, A. Krutz, and T. Sikora, "Compressed Domain Global Motion Estimation using the Helmholtz Tradeoff Estimator," in *Proceedings of the 17th IEEE International Conference on Image Processing (ICIP 2010)*, Hong Kong, Sept. 2010.

[5] F. Dufaux and J. Konrad, "Efficient, robust, and fast global motion estimation for video coding," *Image Processing, IEEE Transactions on*, vol. 9, no. 3, pp. 497 –501, mar 2000.

[6] Jianbo Shi and C. Tomasi, "Good features to track," *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR '94., 1994 IEEE Computer Society Conference on*, pp. 593 –600, jun. 1994.

[7] R. L. Felip, X. Binefa, and J. Diaz-Caro, "A new parameter estimator based on the Helmholtz principle," in *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, Sept. 2005, vol. 2, pp. II–306–9.