Video Stabilization with Reinitialization on Sudden Scene Change

Sunglok Choi and Wonpil Yu
Robot Research Department, ETRI, Daejeon, Republic of Korea
{sunglok, ywp}@etri.re.kr

Abstract—The core step of video stabilization is to estimate global motion from locally extracted motion clues. Optical flow and feature matching have been utilized as local motion clues between adjacent image sequence. However, sudden scene change causes the motion clues totally wrong, which entails wrong global motion. Adaptive RANSAC, authors’ previous work, is applied to solve this problem with reinitialization. The proposed method resets its states when motion estimation is difficult. The difficulty is quantified as the ratio of probably correct motion clues among overall clues. An experiment using real image sequence verifies effectiveness of the proposed method.

Keywords—Video Stabilization, Motion Estimation, RANSAC, MLESAC, uMLESAC

1. Introduction

Video stabilization is the process to generate a compensated video without undesired motion (e.g. vibration) [1]. It has received attention due to widely used video devices. It can improve quality of video captured by cameras on a shaking hand or moving mobile robot. Video stabilization generally comprises four steps: motion estimation, motion filtering, image warping, and image enhancement. Motion estimation is the process to calculate camera motion between two adjacent images. Recently, corresponding points, which are extracted by KLT tracking or SIFT matching, are preferred as motion clues. Wrongly matched points, which are called outliers, are rejected by RANSAC, Hough transform, and others. Motion filtering is the process to separates intended motion and undesired vibration. Kalman filter and Gaussian smoothing give low frequency motion, which is regarded as intended motion and the other high frequency part is regarded as undesired vibration. Image warping is the process to compensate undesired motion in the given images, which is simple image transformation. Image enhancement is additional post-processes, which includes deblurring and recovering missing parts.

The most important step among them is motion estimation because it provides the essential information to the next steps. Even if RANSAC and other robust estimators avoid totally wrong motion clues, they give incorrect estimation due to suddenly changed image sequences. This paper investigates how to detect and overcome such situation. It is based on the authors’ previous work [1], which is briefly explained in Section 2. It quantifies the difficulty of estimation as the ratio of probably correct clues, in the other words, the lower ratio means more danger in estimating incorrect estimation. A threshold on the ratio can detect such severe situation for estimation. In the severe situation, the reinitialization is performed not to have an effect of wrong estimation, which is described in Section 3. Experiments in Section 4 confirms effectiveness of the proposed reinitialization.

2. Video Stabilization using Adaptive RANSAC

2.1 Motion Estimation

Three items are important in motion estimation: motion model, motion clue, and estimation algorithm. The authors approximates instantaneous camera motion as 2D affine motion model, which has 6 DOF as follows:

\[ M = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ 0 & 0 & 1 \end{bmatrix}. \]  

(1)

It can describe translation, rotation, scaling, and shearing of an image plane except perspective projection. KLT tracker [?] extracts corresponding points between adjacent images. The pairs of points are used as data (or clues) in estimating motion between two images. Common least squares provides motion from more than three pairs of points. Figure 5(a) shows a slice of video with KLT tracker, which contains two kinds of motion caused by static background and moving cars. KLT trackers by moving cars disturbs least squares because they includes motion not only by the camera, but also by the cars. Such outliers need to be excluded in least squares.

2.2 uMLESAC, Adaptive RANSAC

uMLESAC [1], [2] is applied to the outlier problem in motion estimation. It is based on Torr and Zisserman’s MLESAC [3] and their error probability density function (pdf). The error pdf models outlier error as uniform distribution and inlier (data except outliers) error as unbiased Gaussian distribution as follows:

\[ p(e|M) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left( -\frac{e^2}{2\sigma^2} \right) + (1 - \gamma) \frac{1}{\nu}, \]  

(2)

where \( M \) is the model to estimate (e.g. affine motion) and \( \nu \) is the size of error space. The model has two parameters \( \gamma \) and \( \sigma^2 \), where \( \gamma \) is prior probability of being an inlier and \( \sigma^2 \) is variance of Gaussian noise (Figure 1). The parameter \( \gamma \) means the ratio of inliers to whole data and \( \sigma^2 \) means the magnitude of noise which contaminate inliers. The error
model assumes that every datum has the same inlier prior probability. It leads posterior probability of being an inlier as follows:

\[ \pi_i = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-x_i^2}{2\sigma^2}\right) \]

Expectation maximization (EM) gives iterative solution of \( \gamma \) with respect to the given data as follows:

\[ \gamma = 1 - \frac{1}{N} \sum_{i=1}^{N} \pi_i, \]

where \( N \) is the number of data. More detail procedure of uMLESAC is described in the authors’ previous works [1], [2].

2.3 Motion Filtering and Image Warping

Kalman filter is employed for motion filtering, which relieves high frequency motion. Constant velocity model is utilized under assumption that each parameter is independent [4]. Motion in each frame needs to be accumulated to compensate motion from the initial frame as follows:

\[ T_0 = T_0^{-1} M_t \quad \text{and} \quad \tilde{T}_0 = \tilde{T}_0^{-1} \tilde{M}_t, \]

where \( \tilde{M}_t \) is filtered motion of \( M_t \) via Kalman filter, and \( T_0' \) and \( \tilde{T}_0' \) are raw and filtered accumulated motion, respectively. The stabilized image \( \tilde{I}_t \) is simply generated from the original image \( I_t \) through

\[ \tilde{I}_t = \tilde{T}_0'(T_0')^{-1} I_t. \]

Figure 2 shows overall procedure of video stabilization.

3. Reinitialization on Sudden Scene Change

Sudden scene change generates wrong KLT trackers as like Figure 5(b), which causes incorrect motion estimation \( M_t \). The wrong estimation \( M_t \) corrupts \( \tilde{M}_t, T_t, \) and \( \tilde{T}_t \). Finally, the stabilized images becomes totally fail after time \( t \) as like Figure 6(a).

The proposed stabilization includes an additional step before Image Warping in Figure 2, which detects such failure estimation and reinitializes its states, \( T_t \) and \( \tilde{T}_t \). The proposed method generates the stabilization image via Equation 6 in normal situation, but it just assign its input image when it detect the failure. The detection is performed using the estimated inlier ratio \( \gamma \) because it implies how many wrong KLT trackers is in the given image. The proposed step is described in Figure 3.

4. Experiment

An experiment was performed in real image sequences, which contained several urgent movements. Its 388–415-th frames are presented in Figure 5(b). Two parameters, \( \tau_p \) and \( \tau_r \), were assigned as 10 and 0.5, respectively. The inlier ratio \( \gamma \) was estimated as Figure 4 during overall image sequence. This result verifies that the inlier ratio is a enough measure to represent difficulty of motion estimation. The inlier ratio less than 0.5 was observed around 400, 460, and 540 frames, which were exactly in sudden image changes. Figure 6 shows stabilized images of Figure 5(b) with/-out the proposed method. The result with the reinitialization seems better, because its accumulated motion does not diverge.

5. Conclusion and Further Works

The proposed method detects failure motion estimation and reinitializes its states. The experiment with real images confirms its effectiveness.

Many meaningful works have been performed on video stabilization, but the video stabilization needs to be investigated more. The 3D motion model or multiple 2D models
The single 2D model cannot describe motion in complex 3D environment which has various depth. A drift error, which is originated from Equation 5, should be solved for long-term operation.

References


