International Conference on Pattern Recognition, ICPR08,
Tampa Convention Center, Florida, USA.

# Extended Kalman Filtering approach to stereo video stabilization

Kai Ki Lee[1], Kin Hong Wong[2], Michael Ming Yuen Chang[1], Ying Kin Yu[3] and Man Kin Leung[2]
*Dept. of Information Engineering, The Chinese University of Hong Kong, Hong Kong.[1]*
*Dept of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong[2]*
*Dept. of Computer Science, Konkuk University, South Korea[3]*
*E-mail: {kklee6, mchang}@ie.cuhk.edu {khwong, mkleung}@cse.cuhk.edu.hk, ykyu@kku.ac.kr.*

## Abstract

*Processing of stereo images has become more and more important in recent years because of the availability of various stereo displaying devices. In particular, stabilizing of stereo images is important and useful especially when the images are obtained from cameras held by inexperienced hands or placed on unstable platforms. In this paper, we propose a new frame warping method for such a problem. Most video stabilization methods to date use 2-D geometric transform to approximate the changes between frames. However, these methods fail when there is a large depth variation between the foreground and the background in the scene. We try to solve this by estimating the 3-D motion parameters of the cameras by tri-focal tensor and the Extended Kalman filter. And use the motion parameters to stabilize the images. We test the method using synthetic and real images and the results show that the performance of our proposed method is accurate even if the background contains large relative depth variations.*

## 1. Introduction

3-D video input and display, especially stereo videos, have received significant attention in research and becomes more and more important in recent years. It is largely because of the developments of new 3-D display devices. In this paper, we study the stabilization of such video clips from handheld cameras or on mobile platforms because they are easily degraded by parasitic motions. The common causes are camera mechanical vibrations or random motion from a person holding the camera. This is a well known problem and different video stabilization techniques have been developed to stabilize and reduce the annoyance of such undesirable motion effects.

Some hardware approaches use motion sensors such as accelerometers, gyroscopes, dampers and active optical system to detect and compensate for the parasitic camera motion. This solution is relatively more costly, whereas processing of the video sequence using computer vision provides a much cheaper way to handle the unstable videos. In this approach, we follow the assumptions of the general stabilization framework. The parasitic motions do not modify the individual frame content and blurring is not usually taking place. 2-D global motion model is commonly used [1, 2]. However, we use the 3-D motion model for the reasons mentioned later in this paper. Inter-frame motion parameters can be estimated by phase correlation, global matching cost function, or feature tracking in [4]. The Kalman filtering is used in [3] to filter out the parasitic motions.

## 2. Problem setting

### 2.1. Image System

In our system, the state vector $s_t$ is defined as a 6-dimensional vector, and the global motion model $M_t$ is modeled as 4*4 rigid transform twist motion model [8] which consists of a rotational matrix $R_t$ and a translational vector $T_t$. K is the intrinsic parameters and E is the 3*4 rigid transformation between 2 stereo cameras. Both K and E are found in the camera calibration. For simplicity, the cameras used in our system are assumed to be identical. $p_{m,t}$ and $p_{m,t}$' is the left and right image coordinates respectively.

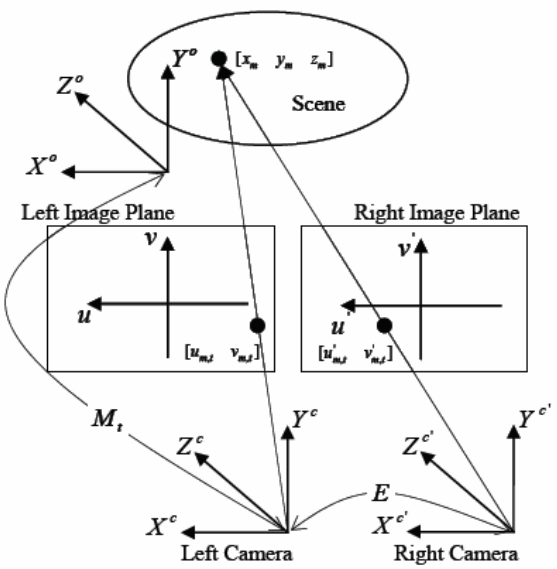

Fig. 1, The geometric model used in this study

The relationship between the 3-D world and its projection on both images are as below,

$$\begin{bmatrix}\tilde{u}_{m,t}\\ \tilde{v}_{m,t}\\ \tilde{w}_{m,t}\end{bmatrix} = K[I_{3x3} \quad 0_{3x1}]\begin{bmatrix}x_m\\ y_m\\ z_m\\ 1\end{bmatrix} \qquad \begin{bmatrix}\tilde{u}'_{m,t}\\ \tilde{v}'_{m,t}\\ \tilde{w}'_{m,t}\end{bmatrix} = KEM_t\begin{bmatrix}x_m\\ y_m\\ z_m\\ 1\end{bmatrix} \tag{1}$$

$$p_{m,t} = \begin{bmatrix}u_{m,t}\\ v_{m,t}\end{bmatrix} = \begin{bmatrix}\tilde{u}_{m,t} / \tilde{w}_{m,t}\\ \tilde{v}_{m,t} / \tilde{w}_{m,t}\end{bmatrix} \tag{2}$$

$$p'_{m,t} = \begin{bmatrix}u'_{m,t}\\ v'_{m,t}\end{bmatrix} = \begin{bmatrix}\tilde{u}'_{m,t} / \tilde{w}'_{m,t}\\ \tilde{v}'_{m,t} / \tilde{w}'_{m,t}\end{bmatrix} \tag{3}$$

### 2.2. Motion Model

When the background has large relative depth variation, the transformation between frames in the video is not able to be approximated by simple 2-D geometric transform, such as affine transformation applied in [4]. In this case, estimating a full 3-D model of the scene including depth is necessary. According to the study of Morimoto and Chellappa in [9], more complex models may perform worse than simple models due to their sensitivity to tracking errors, hence we would use the 3-D model with the help of stereo images to eliminate such errors and estimate the intentional motion model.

## 3. Methodology

### 3.1. Feature Tracking

In our study, we use the Kanade-Lucas-Tomasi (KLT) tracker in [5] to extract features from the left and right image sequences independently. We assume they are contaminated by Gaussian noise only. Outliers are filtered off in the next step.

### 3.2. Stereo Correspondences

First, we match the points putatively based on their normalized correlations. Then we use the eight-point algorithm in [6], the matched points and the Random Sample Consensus (RANSAC) robust estimator in [7] to compute the fundamental matrix F. Since we have the intrinsic parameters K, we could calculate the required extrinsic parameters E from F with the method mentioned in [6]. Stereo correspondences are then found by the guided search.

### 3.3. Motion estimations

The parameters are estimated by the smoothing Kalman filter which is mentioned in [8] under the idea of [3]. Trifocal tensor is used to constrain the 2-D positions of the feature points in every three views in the measurement model. Since trifocal tensor is used, hence the computation steps of 3-D models are eliminated. The base pair is the first stereo image sequence pair. The pair is the two views of the trifocal tensors. The left image at time t is used as the third view of the trifocal tensor for the left side. Similarly, the right image at t is used as the third view of another trifocal tensor. The illustration of the trifocal tensors in our stereo system is shown in Fig. 2.

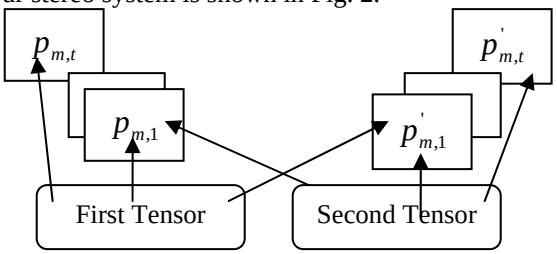

Fig. 2, Illustration of the trifocal tensors in our stereo system

$T_t(\cdot)$ is the trifocal tensor point transfer function. $v_t$ is the noise of the measurement $y_t$ which is defined as,

$$y_t = T_t(M_t) + v_t \tag{4}$$

The tensors can be expressed in tensor notation as,

$$T_i^{jk} = a_i^j b_4^k - a_4^j b_i^k \tag{5}$$

$$T'^{jk}_i = a'^j_i b_4^k - a'^j_4 b_i^k \tag{6}$$

where $b$ is the extrinsic parameters of the right camera

at base frame. *a* and *a'* are the extrinsic parameters of the left and the right camera at image sequence time t respectively. The epipole $e_{12}$ is estimated in the initialization step. $l_m$ is calculated from (9). It is the line passing through the m[th] feature point in the image sequence of the right camera in the base frame.

$$\bar{p}'_{m,t} = \begin{bmatrix} \bar{u}'_{m,t} \\ \bar{v}'_{m,t} \\ \bar{w}'_{m,t} \end{bmatrix} = \frac{1}{f} \begin{bmatrix} u'_{m,t} \\ v'_{m,t} \\ f \end{bmatrix} \tag{7}$$

$$l'_m = \begin{bmatrix} l_1 & l_2 & l_3 \end{bmatrix}^T = e_{12} \times \bar{p}'_{m,1} \tag{8}$$

$$l_m = \begin{bmatrix} l_2 & -l_1 & -\bar{u}'_{m,1} l_2 + \bar{v}'_{m,1} l_1 \end{bmatrix}^T \tag{9}$$

Then we estimate the state vector $s_t$ using the Kalman Filtering algorithm mentioned in [8]. The prediction equation of the state is shown in (10). $\mathbf{H}(\cdot)$ is the measurement. $\mathbf{K}_t$ is the Kalman gain. The update equation of the state is shown in (11),

$$\hat{s}_t^- = \hat{s}_{t-1} \tag{10}$$

$$\hat{s}_t = \hat{s}_t^- + \mathbf{K}_t \left( \mathbf{y}_t - \mathbf{H}(\hat{s}_t^-) \right) \tag{11}$$

### 3.4. Frame Warping

Resulting compensation parameters [3], which are used to compute the warped frames, are given by,

$$\bar{R} = \begin{bmatrix} \bar{r}_{11} & \bar{r}_{12} & \bar{r}_{13} \\ \bar{r}_{21} & \bar{r}_{22} & \bar{r}_{23} \\ \bar{r}_{31} & \bar{r}_{32} & \bar{r}_{33} \end{bmatrix} = \hat{R}\left(\tilde{R}\right)^{-1} \tag{12}$$

$$\bar{T} = \begin{bmatrix} \bar{l}_x \\ \bar{l}_y \\ \bar{l}_z \end{bmatrix} = -\hat{R}\left(\tilde{R}\right)^{-1} \tilde{T} + \hat{T} \tag{13}$$

The transformed coordinates in frame can be calculated as,

$$\begin{bmatrix} \bar{u} \\ \bar{v} \end{bmatrix} = \frac{z}{z} \begin{bmatrix} \frac{f\bar{x}}{\bar{z}} \\ \frac{f\bar{y}}{\bar{z}} \end{bmatrix} = \frac{z}{\bar{z}} \begin{bmatrix} \bar{r}_{11}u + \bar{r}_{12}v + \bar{r}_{13}f + \frac{f\bar{l}_x}{z} \\ \bar{r}_{21}u + \bar{r}_{22}v + \bar{r}_{23}f + \frac{f\bar{l}_y}{z} \end{bmatrix} \tag{14}$$

Letting $s = \bar{r}_{33}f + \bar{r}_{31}u + \bar{r}_{32}v$, we have,

$$\begin{bmatrix} \bar{u} \\ \bar{v} \end{bmatrix} = \frac{f}{s}\left(1 - \frac{\bar{l}_z}{\bar{z}}\right) \begin{bmatrix} \bar{r}_{11}u + \bar{r}_{12}v + \bar{r}_{13}f \\ \bar{r}_{21}u + \bar{r}_{22}v + \bar{r}_{23}f \end{bmatrix} + \frac{f}{\bar{z}} \begin{bmatrix} \bar{l}_x \\ \bar{l}_y \end{bmatrix} \tag{15}$$

s could also be expressed as the following,

$$s = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \bar{R} \begin{bmatrix} u \\ v \\ f \end{bmatrix} \tag{16}$$

When the background of the stereo images is far away from the camera, we could assume $\bar{z}$ to be very large. Therefore, we could approximate the value of the transformed coordinates as,

$$\begin{bmatrix} \bar{u} \\ \bar{v} \end{bmatrix} = \frac{f}{s}\left(1 - \frac{\bar{l}_z}{\bar{z}}\right) \begin{bmatrix} \bar{r}_{11}u + \bar{r}_{12}v + \bar{r}_{13}f \\ \bar{r}_{21}u + \bar{r}_{22}v + \bar{r}_{23}f \end{bmatrix} = \frac{f}{s} \begin{bmatrix} \bar{r}_{11}u + \bar{r}_{12}v + \bar{r}_{13}f \\ \bar{r}_{21}u + \bar{r}_{22}v + \bar{r}_{23}f \end{bmatrix}$$

$$\begin{bmatrix} \bar{u} \\ \bar{v} \end{bmatrix} = \frac{f}{s} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \bar{R} \begin{bmatrix} u \\ v \\ f \end{bmatrix} \tag{17}$$

If we have the background with small depth variations, we could approximate the value by letting $\bar{z}$ to be the estimate of all $\bar{z}$ in current image pair.

$$\begin{bmatrix} \bar{u} \\ \bar{v} \end{bmatrix} = \frac{f}{s}\left(1 - \frac{\bar{l}_z}{E[\bar{z}]}\right) \begin{bmatrix} \bar{r}_{11}u + \bar{r}_{12}v + \bar{r}_{13}f \\ \bar{r}_{21}u + \bar{r}_{22}v + \bar{r}_{23}f \end{bmatrix} + \frac{f}{E[\bar{z}]} \begin{bmatrix} \bar{l}_x \\ \bar{l}_y \end{bmatrix} \tag{18}$$

A better performance could be achieved by applying dense depth map, which is also known as dense disparity map. We get the value of each pixel in the warp image by bi-linear interpolation.

## 4. Experiment Results

We have tested our algorithm on two kinds of video sequences. The first kind of sequences is generated by reprojecting a synthetic 3-D object onto images with added noise simulating the parasitic motion. The object is inside sphere of 23985 pixels in diameter and is 92251 pixels away from the camera of focal length 848 pixels. The standard deviation of noise for the parasitic translation and rotation are 100 pixels and 0.001 radians respectively per frame. We have done 100 tests on only translational noises, 100 tests on only rotational noises, and 100 tests on mixed noises. The second kind of sequences, sequence B, is taken from a stereo camera.

Fig. 3, fig. 4, fig. 5 and fig. 6 show the stabilization results of synthetic image sequence. The dots are features of the 3-D projected image background. The patterns appear to be more stable in the scene after the stabilization process. Fig. 7 shows frame 5, frame 18 and frame 40 of a real indoor scene sequence B. Fig. 8 shows that our tester tests the performance of the stereo videos.

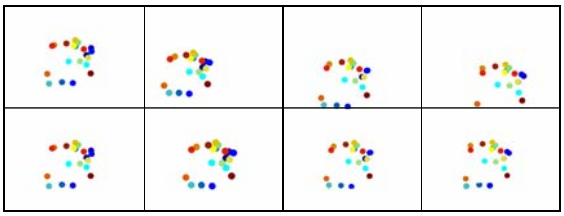

Fig. 3, Test on synthetic camera 3-D parasitic translation motion. The upper row contains the original left images. The lower row contains the stabilized left images. They are relatively stable.

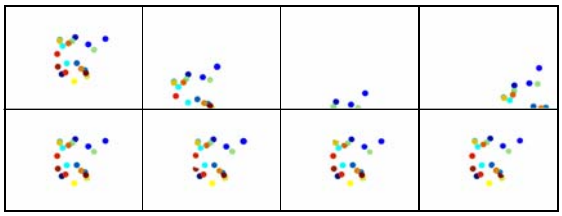

Fig. 4, Test on synthetic camera 3-D parasitic rotational motion. The upper row contains the original left images. The lower row contains the stabilized left images. They are relatively stable.

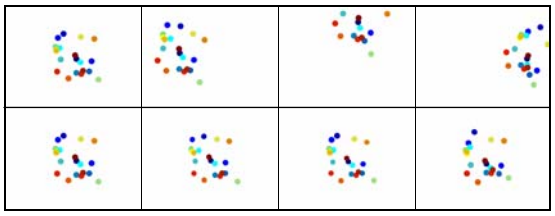

Fig. 5, Test on synthetic camera 3-D parasitic mixed motion. The upper row contains the original left images. The lower row contains the stabilized left images. They are relatively stable.

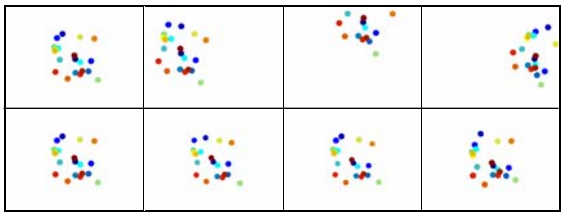

Fig. 6, Test on synthetic camera 3-D parasitic mixed motion. The upper row contains the original right images. The lower row contains the stabilized right images. They are relatively stable.

Fig. 7, Three sequence frames (5, 18, 40) after (left) and before (right) stabilization. The checkerboard pattern wall appears to be motionless in stabilized sequence while it is vibrating in the original sequence.

Fig. 8, One of our testers is watching the stereo video.

## 5. Conclusion

In our system, we have used a 3-D motion model combined with trifocal tensor and Kalman filter for our parameter estimations. Then we use the result and the frame warping method to stabilize our image sequences. We do not need to retrieve the disparity map from the stereo video because it is a complex task, but we estimate the approximation of the depth information for frame warping as in equation (18). We observed that the system could stabilize the video quite well even if this approximation is being used. However, we believe that the performance would be improved if we use the disparity map in the frame warping. This may be explored in the future.

## References


[1] C.H. Morimoto and R. Chellappa. Automatic digital image stabilization, ICPR96, Austria, Aug. 1996

[2] A. Litvin, J. Konrad, and W. Karl, Probabilistic video stabilization using kalman filtering and mosaicking, IS&T/SPIE symposium on Electronic Imaging, Image and Video Communication and Proc., jan 20-24, 2003

[3] A. Litvin, J. Konrad, and W.Karl, Probabilistic video stabilization using kalman filtering and mosaicking, Proc. SPIE Image and Video Communications and Process., vol. 5022, pp. 663-674, 2003

[4] J. Yang, D. Schonfeld, and C. Chen, Online Video Stabilization Based on Particle Filters, ICIP06, 2006.

[5] C.Tomasi and T.Kanade, Detection and tracking of point features, Carnegie Mellon University, Pittsburgh, PA, Tech. Rep. CMU-CS-91-132, Apr. 1991

[6] R.Hartley and A.Zisserman, Multiple View Geometry in Computer Vision, Cambridge University Press, 2000

[7] M.A.Fischler and R.C.Bolles, Random sample concensus: A paradigm for model fitting with applications to image analysis and automated cartography, in Commun. ACM, vol. 24, no. 6, pp. 381-395, Jun. 1981

[8] Y. K. Yu, K. H. Wong, S.H. Or., and M. Y. Y. Chang, Recursive recovery of position and orientation from stereo image sequences without three-dimensional structures, CVPR06, New York, Jun 2006

[9] C. Morimoto and R. Chellappa, Evaluation of image stabilization algorithms, ICASSP98, vol. 5, pp. 2789–2792, 1998