Selective Object Stabilization for Home Video Consumers

Zailiang Pan and Chong-Wah Ngo

Abstract — *This paper describes a unified approach for video stabilization. The essential goal is to stabilize image sequences that consist of moving foreground objects, which appear frequently in today's home videos captured by hand-held consumer cameras. Our proposed techniques mainly rely on the analysis of motion content. Three major components are: initialization, segmentation and stabilization. In motion initialization, we propose a novel algorithm to efficiently search for the best possible frame in a sequence to start segmentation. Our segmentation algorithm is based on Expectation-Maximization (EM) framework which provides the mechanism for simultaneous estimation of motion models and their layers of support. Based on the framework of Kalman filter and EM motion estimation, our proposed algorithm has the flexibility of allowing selective stabilization with respect to background or/and foreground objects, subject to the preferences of customers*[1].

Index Terms — Digital Image Stabilization, Motion Segmentation, Kalman Filter.

I. INTRODUCTION

Nowadays, digital cameras are widely used to capture personal and family lives. However many valuable videos are simply discarded due to their shaking artifacts. These artifacts are commonly seen in consumer videos due to the amateurish operations of cameras, particularly when the holders are in motion or in moving vehicles. A digital image stabilization (DIS) system is needed to produce compensated video sequence so that the inevitable and undesirable camera motions can be removed. Moreover, it would be worthwhile to design a DIS scheme that can further stabilize the video visual effects, not only restricted to the camera motion, but also other visual components, such as the foreground objects.

In general, digital image stabilization system consists of two parts: motion estimation and motion correction subsystems. The motion estimation system aims to estimate the global inter-frame motion. The global motion is usually represented by 2D or 3D geometry transformation models of the scene. 3D model, although desirable, is generally an ill-posed problem. The typical 2D models include 2-parameter translation model [11, 14], 4-parameter rigid model [12, 17] and 6-paramter affine model [5, 18]. The affine model can precisely describe purely rotation, panning, zooming and translation. For most of indoor and outdoor scenes, the affine model is enough for estimation. A variety of approaches for motion model estimation has also been proposed. For instance, block matching algorithms, which search for the best matching of blocks between two images, are popularly used [13, 15, 16], due to its simplicity. More precise approaches are also proposed by minimizing the cost functions based on salient features [2, 3, 5] or image intensities [18].

The *motion estimation* part forms the basis of a DIS system. Accurate motion estimation is critical. To date, most existing works assume non-foreground-objects or static foreground objects. This assumption, nevertheless, does not hold in most home videos. The frequent presence of moving foreground objects in home videos indeed causes DIS a challenging task. To tackle this problem, the approaches in [15, 16] propose to estimate the background and foreground motions respectively based on the predefined foreground and background regions. However, these regions are usually small and fixed, and prohibit the precise estimation of motion due to either foreground or background objects.

In order to remove undesirable motion in home videos, the intentional camera motion should be estimated from the global inter-frame motion. One way is to smooth the camera motion by low-pass filtering [19], regardless of the spatial dynamics of camera motion. A flexible approach is motion vector integration (MVI) [13, 15, 16], which uses a damping factor to compress the motion fluctuation in spatial domain. This can somewhat approximate, although not precise enough, the intentional motions. More accurate motion estimation algorithm based on Kalman filtering (KF) is proposed in [14, 18]. In KF framework, intentional motion is represented by a physically meaningful state-space model, while undesirable motion is described by random noise. However, the estimated motion trajectory by KF is usually delayed with respect to the actual intentional movement. This is because the traditional KF only utilizes the observations in the past.

Our DIS system described in this paper is also composed of two parts: motion estimation and motion correction subsystems. For motion estimation, we propose to automatically and simultaneously estimate the motions of foreground and background objects, and their support layers by expectation-maximization (EM) segmentation algorithm [10]. The EM-based motion estimation algorithm can be regarded as minimizing cost function on multiple image layers each with an affine motion model. This indeed leads to accurate motion estimation of video objects. Here we regard

Please contact Dr. Chong-Wah Ngo for any enquiry. Email: cwngo@cs.cityu.edu,hk.

Z. Pan and C. W. Ngo are with the Department of Computer Science, City University of Hong Kong, 83 Tat Chee Avenue, Kowloon Tong, Hong Kong. Tel: (852)2784-4390. Fax: (852)2788-8614. Email: {zerin, cwngo}@cs.cityu.edu.hk.

background as a specific object. Then an image sequence can be decomposed into several object layer sequences and their associated motion trajectories. For motion correction, we propose a dual Kalman filtering (DKF) in both forward and backward temporal directions for video stabilization. The delay of traditional KF algorithm can be counteracted in the dual Kalman filter stabilizer by the combination of forward and backward Kalman filtering. Ultimately, motion correction can be achieved by warping the current frame to the desirable intentional trajectory. Undefined regions of warped frame can be filled by dynamic mosaics.

Traditional DIS systems only stabilize video camera since physically shaking artifacts are the cause of irregular movement. However, stabilization with respect to camera motion only may not be enough and appropriate. For instance, in many circumstances, camera stabilization can lead to awful foreground movement and make the visual quality even worse. Thus a good DIS system should have the flexibility of allowing the selective stabilization of video components. In our approach, by exploiting EM algorithm for compact video decomposition, we develop a stabilization algorithm with object selectivity function. It permits the stabilization of background, foreground, or a combination of arbitrary objects, subject to the choices of consumers.

One significant component in our selective object stabilization algorithm is EM motion estimation. However, EM-based algorithm is usually sensitive to initial conditions, and thus cannot be directly applied to home videos. When bad initial conditions are intertwined with poor quality frames caused by shaking artifacts or low visual resolution, the results of estimation are unreliable. One prominent solution to this problem is to look for the best possible frame, in a bunch of jerky frames, to estimate initial conditions, and then initialize others frames with the estimated conditions. Motivated by this idea, we propose a novel initialization technique to rapidly locate the best possible frame in an image sequence to start initialization. Our technique is developed based upon 3D tensor representation and robust clustering algorithm. The best possible initial frames are usually the video frames where their optical flows can be unambiguously clustered into few distinct moving groups.

By the proposed motion initialization, the EM algorithm can start temporally forward and backward at the selected frame for simultaneous segmentation and estimation. The clusters of objects, found by motion initialization, are utilized to initialize the proposed EM algorithm. Since the initial conditions are estimated based on the best possible frame, more trust is given to the initial conditions. Taking this into account, we modify and improve the existing EM algorithm in [10]. The modification leads to more reliable estimation, particularly in homogenous image regions. Most importantly, it is found to be appropriate for home videos that are suffered from low visual quality.

The remaining part of this paper is organized as follows. Section II presents the overview of our approach. Section III proposes the estimation of initial condition based on 3D tensors and robust clustering. Section IV describes the modified EM motion estimation algorithm. Section V presents our proposed algorithm for supporting selective object stabilization. A dual Kalman filtering and combinational stabilizer is proposed to stabilize arbitrary objects selected by consumers. Finally Section VI presents the experiment results and Section VII concludes this paper.

II. Overview of our Approach

Fig. 1 illustrates the proposed framework for DIS. Three major components are: motion *initialization*, *estimation* and *stabilization*. The goal of motion initialization is to temporally locate the best possible initial positions in a sequence for segmentation. Basically, the good starting frames that have good initial conditions are obtained by analyzing the optical flows in a 3D image volume. We adopt 3D structural tensor representation for motion estimation due to its robustness and efficiency. This representation allows each optical flow vector, computed over a 3D spatio-temporal block, associated with a saliency measure. By incorporating the saliency measures into motion clustering, the initial frame for motion estimation is selected based on the quality of clustering.

Motion estimation started at the selected initial frames is carried out progressively in a bi-directional fashion along the temporal dimension, by giving the initial layers of support obtained from motion clustering. To accurately estimate the motions with multiple foreground objects, EM algorithm is used for the simultaneous estimation of multiple parametric motion models and their layers of support. The final segmented layers are then utilized for video stabilization. In our framework, stabilization is viewed as a process of estimating the intentional trajectories of video objects. Our stabilization algorithm allows the selective stabilization with respect to objects specified by consumers.

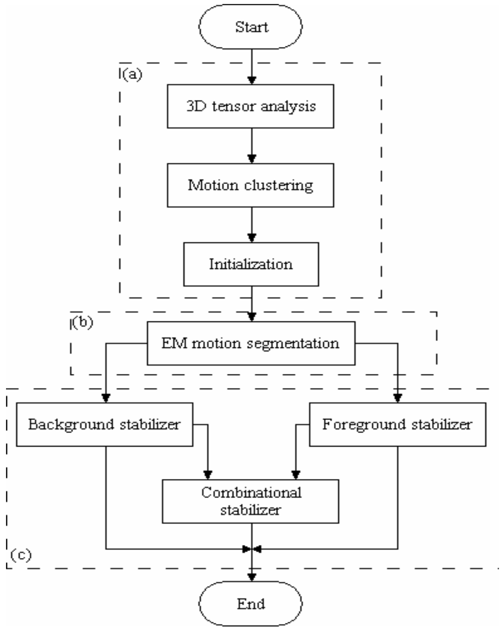


Fig. 1. The overview of our approach. (a) Motion initialization. (b) Motion estimation. (c) Motion stabilization.

III. MOTION INITIALIZATION

In this section, we start with 3D structure tensor computation and representation. The computed optical flows and their associated saliency (or fidelity) measures are utilized directly for clustering.

A. 3D Tensor Representation

Let $I(x, y, t)$ be the space-time intensity of a point in a 3D image volume. Assume $I(x, y, t)$ remains constant along a motion trajectory, a constraint condition of optical flow can be derived as

$$\frac{dI}{dt} = \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \frac{\partial I}{\partial t} = \varepsilon \qquad (1)$$

where u and v represent the components of local spatial velocity, namely optical flow, and ε is a noise variable assumed to be independent, white and zero-mean Gaussian. Eqn (1), more precisely, is the inner product of a homogeneous velocity vector V and a spatio-temporal gradient ∇I , i.e.

$$(\nabla I)^T V = \varepsilon \qquad (2)$$

where $V = [u, v, 1]^T$ and $\nabla I = \left[\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}, \frac{\partial I}{\partial t}\right]^T$. Eqn (2) has been widely applied in motion estimation algorithms. In particular, the noise term ε^2 is frequently used as a fidelity measure of the optical flow. Nevertheless, any fidelity measure that involves only ε cannot fully exploit the fact that the estimated local velocity in a region with high intensity variability is more reliable than in a region with low variability. To tackle this deficiency, we introduce a fidelity term based on 3D tensor representation for robust estimation.

Under the assumption that the flows are constant over a 3D volume R , the total sum of ε^2 in R can be derived as

$$E = \sum \varepsilon^2 = V^T \left(\sum_{x,y,t \in R} (\nabla I)(\nabla I)^T \right) V \qquad (3)$$

The central term is a symmetric tensor which represents the local structure of R in space-time dimension. The tensor has the form

$$\Gamma = \begin{bmatrix} J_{xx} & J_{xy} & J_{xt} \\ J_{yx} & J_{yy} & J_{yt} \\ J_{tx} & J_{ty} & J_{tt} \end{bmatrix} \qquad (4)$$

where

$$J_{mn} = \sum_{x,y,t \in R} \frac{\partial I}{\partial m} \frac{\partial I}{\partial n} \quad m, n = x, y, t .$$

Given the tensor representation in Eqn (4), the optical flow can be estimated by minimizing the cost function E in Eqn (3). The diagonal components of a tensor which represent the intensity variation in spatio-temporal coordinate can be exploited for fidelity measure. Thus, our proposed fidelity term λ , which depicts the certainty of estimated optical flow in R , is defined as

$$\lambda = 1 - \frac{E}{E + J_{xx} + J_{yy}} .$$

The fidelity term has following favorable properties: i) It is maximal for ideal flows, i.e. $E = 0$. ii）It is minimal if no spatial intensity variation, i.e. $J_{xx} + J_{yy} = 0$. iii) Its value is normalized between [0, 1].

B. Initialization

Given the flows $\{\mathbf{v}_i\}$ and their fidelities $\{\lambda_i\}$ at time t as described in Section II-*A*, we employ k-means algorithm to cluster optical flows in each frame. The number of clusters, g , is initially set to a reasonably large value. Then the clusters are merged one by one according to the distance between each two clusters. The algorithm is described in Algorithm 1. The results are then used to define motion saliency based on scattering of both inter and intra classes as follows,

$$\begin{aligned} \eta &= tr(\eta_w^{-1}\eta_b) \\ \eta_w &= \sum_{j=1}^{g} p_j C_j \\ \eta_b &= \sum_{j=1}^{g} p_j (M_j - \sum_{k=1}^{g} p_k M_k)(M_j - \sum_{k=1}^{g} p_k M_k)^T \end{aligned} \qquad (5)$$

where M_j, C_j and p_j are defined in Algorithm 1, η_w and η_b are the expected intra and inter distances respectively. The more distinct the motions are, the larger the saliency η would be.

Therefore the motion saliency is an appropriate criterion to measure the frame's motion quality. We actually select the frame with largest saliency value as the best frame t^* ,

$$t^* = \arg\max_t (\eta_t) . \qquad (6)$$

An apparent advantage of this approach is that the g and $\mathbf{v}_j$ in the selected initial frame can be passed as initial parameters for motion segmentation described in the next section.

1. Given the cluster number g , initial classification $\{u_{ij}\}$ is calculated by k-means algorithm on N optical flows $\{\mathbf{v}_i\}$ incorporating with fidelities $\{\lambda_i\}$, where $u_{ij} = 1$ if $\mathbf{v}_i$ belongs to the j^{th} cluster and $u_{ij} = 0$ otherwise.
2. Calculate the cluster probability $\{p_j\}$,
$$p_j = \frac{\sum_i u_{ij}}{\sum_j \sum_i u_{ij}}$$
3. Compute the cluster means $\{M_j\}$ and covariance matrices $\{C_j\}$ by using the robust estimator, minimum volume ellipsoid (MVE) algorithm [9].
4. Calculate the distance d_{kl} between each two cluster k and l as follows
$$d_{kl} = s(1-s)(M_k - M_l)^T$$
$$[sC_k + (1-s)C_l]^{-1}(M_k - M_l)$$
where $k, l \in 1\cdots g$ and $s = p_k / (p_k + p_l)$.
5. Select the closest two clusters k^* and l^* , $k^* < l^*$, such that $d_{k^*l^*} = \min_{ij}\{d_{ij}\}$
6. If $d_{k^*l^*} < \tau$, a threshold, merge the two clusters. That is to set
$$u_{ik^*} = 1 \quad \forall\{i \in 1\cdots N \wedge u_{il^*} = 1\},$$
$$u_{i(j-1)} = u_{ij} \quad \forall\{i \in 1\cdots N \wedge j \in l^* + 1, \cdots, g\},$$
$$g = g - 1 .$$
7. Go to step 2, until no clusters are merged.

Algorithm 1. Motion clustering.

IV. MOTION ESTIMATION

The shaky artifacts in home videos are mostly due to irregular camera motion. To remove these artifacts, an essential step is to estimate the trajectory of the camera motion. The camera motion estimation, nevertheless, is not straightforward due to the presence of moving foreground objects. Direct estimation without foreground and background segmentation can normally lead to bias computation. In this section, we adopt an EM framework, similar to the excellent works of Sawhney & Ayer in [10], for simultaneous motion segmentation and parameter estimation.

A. The Motion Model

We use a common approach to describe the motion by a set of parameters with an assumption that the observed scene undergo a geometry transformation. The 6-parameter affine model is adopted as the trade-off between model stability and representative capacity. The affine transformation of a pixel position $\mathbf{p}$ between frames I_t and I_{t+1} is given by

$$\mathbf{p}^{t+1} = \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix} \mathbf{p}^t + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = A_t \mathbf{p}^t + \mathbf{b}_t . \qquad (7)$$

With the assumption of no moving objects, the affine parameters $\theta_t = [A_t, \mathbf{b}_t]$ can be estimated by robust M-estimator as in [10].

B. Expectation-Maximization Motion Segmentation

Only one affine motion model is not enough to describe the scene changes with multiple moving objects. However we can regard the reference frame at time t as being generated by the frame at time $t-1$ with multiple affine models corresponding to the moving camera and objects. Thus each intensity $I_t(\mathbf{p})$ at pixel $\mathbf{p}$ of reference frame can be viewed as arising from the mixture of a finite number of images $\widetilde{I}_1, \cdots, \widetilde{I}_g$ with some unknown proportions $\pi_1, \cdots, \pi_g$, respectively, where

$$\sum_{i=1}^{g} \pi_i = 1, \quad \pi_i \geq 0, \quad i = \{1, \cdots, g\} .$$

The images $\widetilde{I}_{i=1:g}$ are derived from frame I_{t-1} with different motion parameters $\theta_{i=1:g}$, that is

$$\widetilde{I}_i(\mathbf{p}) = I_{t-1}(\mathbf{p}, \theta_i) = I_{t-1}(A_i \mathbf{p} + \mathbf{b}_i) \qquad i = 1, \ldots, g .$$

Given the i^{th} population $\widetilde{I}_i$, the conditional probability density function of intensity $I_t(\mathbf{p})$ is assumed to be a normal distribution $N(\widetilde{I}_i(\mathbf{p}),\sigma_i^2)$ with the variance σ_i^2. That is

$$p(I(\mathbf{p});\theta_i,\sigma_i)=\frac{1}{\sqrt{2\pi}\sigma_i}\exp-\frac{r_i^2(\mathbf{p})}{2\sigma_i^2} \qquad (8)$$

where $r_i(\mathbf{p})=I(\mathbf{p})-\widetilde{I}_i(\mathbf{p})$.

Let vector $\Psi=[\Pi^T,\Sigma^T,\Theta^T]^T$ denote all the unknown parameters, proportions $\Pi=[\pi_1,\cdots,\pi_g]^T$, variances $\Sigma=[\sigma_1,\cdots,\sigma_g]^T$ and motion parameters $\Theta=[\theta_i,\cdots,\theta_g]^T$. The log likelihood function can now be written as

$$L(\Psi)=\sum_{j=1}^{n}\log\left(\sum_{i=1}^{g}\pi_i\,p(I(\mathbf{p}_j);\theta_i,\sigma_i)\right) \qquad (9)$$

where n is total number of pixels in current frame. The solution to Eqn (9) is usually obtained by maximizing through EM algorithm by introducing the hidden variable, ownership indicator $Z=\{z_{ij}\}$ that,

$$z_{ij}=\begin{cases}1 & I(\mathbf{p}_j)\in\widetilde{I}_i\\ 0 & I(\mathbf{p}_j)\notin\widetilde{I}_i\end{cases}$$

B. *Extending E-step*

In the case where initial conditions are absent that randomly initialized conditions are used, the update of ownership, calculated in a way as [10], is reasonable. In our case, however, the ownership of the starting frame, obtained from Section III-*B*, is known. Since the initial conditions are estimated in the best frame, we have the confidence that they are close to the segmentation result of EM algorithm. This prior information would make the EM algorithm more robust if it can be considered in EM iteration. Therefore we extend the E-step of EM motion segmentation algorithm in [10] by assuming ownership Z is propagated by a given transit matrix M. Then in the l^{th} iteration, the expectation τ_{ij} of current ownership z_{ij} at j^{th} pixel for i^{th} population, is given by,

$$\begin{aligned}\tau_{ij}&=E(z_{ij}^{(l)}\mid I_t(x_j),\Psi,Z^{(l-1)})\\&=p(z_{ij}^{(l)}=1\mid I_t(x_j),\Psi,Z^{(l-1)})\\&=\frac{p(I_t(x_j)\mid z_{ij}^{(l)}=1,\Psi)\,p(z_{ij}^{(l)}=1\mid Z^{(l-1)})}{\sum_{k=1}^{g}p(I_t(x_j)\mid z_{kj}^{(l)}=1,\Psi)\,p(z_{kj}^{(l)}=1\mid Z^{(l-1)})}\\&=\frac{p(I_t(x_j)\mid\theta_i,\sigma_i)\,p(z_{ij}^{(l)}=1\mid Z^{(l-1)})}{\sum_{k=1}^{g}p(I_t(x_j)\mid\theta_k,\sigma_k)\,p(z_{kj}^{(l)}=1\mid Z^{(l-1)})}\end{aligned}$$

where

$$p(z_{kj}^{(l)}=1\mid Z^{(l-1)})=\sum_{i=1}^{g}m_{ki}z_{ij}^{(l-1)}\quad\forall k\in 1\ :\ g$$

Here $\mathrm{M}=\{m_{ij}\}_{ij=1:g}$ is the transit probability matrix, which describes our confidence on the initial segment results given by the clustering algorithm in Section III-*B*. One advantage of extended E-step, from the practical point of view, is that the EM framework, in contrast to typical ownership update, is more robust in homogeneous segmentation. In a typical EM solution like [10], a population $\widetilde{I}_i$ whose proportion is relatively small compared with other populations $\widetilde{I}_{j\neq i}$, is more likely to vanish after update. This is a common scenario particularly for some foreground objects which occupy small regions in home videos.

C. *M-step*

As shown in [10], given the current population ownership expectation τ_{ij}, the maximum likelihood estimate of parameter Ψ, $\hat{\Psi}$, satisfies the following equations:

$$\hat{\pi}_i=\sum_{j=1}^{n}\tau_{ij}/n\quad i=\{1,\cdots,g\}, \qquad (10)$$

$$\sum_{i=1}^{g}\sum_{j=1}^{n}\tau_{ij}\frac{\partial\log p(I(\mathbf{p}_j);\theta_i,\sigma_i)}{\partial\hat{\sigma}_i}=0\ , \qquad (11)$$

$$\sum_{i=1}^{g}\sum_{j=1}^{n}\tau_{ij}\frac{\partial\log p(I(\mathbf{p}_j);\theta_i,\sigma_i)}{\partial\hat{\theta}_i}=0\ . \qquad (12)$$

In our approach, Gaussian-Newton algorithm with robust M-estimator is used to solve Eqn (12) as in [10].

The E-step and M-step are iterated until some conditions such as the predefined change of successive motion model or number of iteration are reached. At last, the segmentation divides the current frame into g layers $\{L_i\}$. Each layer is given by

$$L_i=\{\mathbf{p}_j\mid\tau_{ij}>\tau_{kj},\ k\neq i\}.$$

The corresponding affine model parameter θ_i is estimated by Eqn. (12). Starting from the selected frame, this procedure is carried on in both forward and backward temporal directions. In a new frame, the initial ownership indicators are transformed from the ownership and motions estimated in the previous frame. In this manner, we have g sequence of $\{L_t,\theta_t\}^j$ for the whole image sequence.

V. SELECTIVE OBJECT STABLIZATION

In a typical home video, there could be many annoying perturbations caused by irregularities in camera motion. If the irregularities are regarded as the noise of a motion trajectory, video stabilization can be viewed as a filter that estimates the intentional camera motion trajectory from the observed trajectory corrupted by unwanted motions. Rather than using either overly simple assumption of camera motions or smooth filtering in frequency domain without care of physical movement of the camera, we use Kalman filtering from recursive estimation theory. In the framework of Kalman filtering, we propose dynamic motion model that can physically describe the intentional camera movement, on the other hand, the undesirable motions can be removed by regarding as the measurement noise. To overcome the delay problem of intentional trajectory estimation by a single forward Kalman, we propose a dual Kalman filtering in both forward and backward directions. The delay can be counteracted in this way.

More importantly, since the image sequence is decomposed into several layer sequences by EM segmentation algorithm (Section IV), we are not restricted only on the stabilization of the camera. All the moving objects including the camera and foreground objects can be subjectively selected and stabilized. This selective object stabilization provides flexibility and user-friendly interface for the consumers.

A. *Motion Feature Extraction*

To establish the dynamics of motion, we need to know the physical meanings. The affine parameter A (Eqn (7)) is lack of this attribute. We decompose A to extract the motion features by QR decomposition,

$$\mathbf{A} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} k_1 - 1 & (k_1 - 1)\tan(\phi) \\ 0 & (k_2 - 1)/\cos(\phi) \end{bmatrix} \quad (13)$$

where θ and ϕ are the rotation and skew angles respectively. The k_1 and k_2 are zoom factors in horizontal and vertical directions respectively. Together with the translation $\mathbf{b}$, we have a six-motion-feature vector $\mathbf{x} = [b_1, b_2, \theta, \phi, k_1, k_2]$ associated with each two successive frames.

B. *Intentional Motion Estimation*

To estimate the intentional trajectory, we should carefully establish the dynamics of the motion. Based on the physical meaning of $\mathbf{x}$, we assume that each of the cumulative of b_1, b_2 and θ, denoted by $\widehat{b}_1$, $\widehat{b}_2$ and $\widehat{\theta}$, changes with a constant velocity, denoted by b_1^v, b_2^v and θ^v, which is subject to random noise. The dynamics of, say the horizontal translation, is given by

$$\begin{bmatrix} \widehat{b}_1 \\ b_1^v \end{bmatrix}^{t+1} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \widehat{b}_1 \\ b_1^v \end{bmatrix}^{t} + \begin{bmatrix} 0 \\ N(0, \sigma_b) \end{bmatrix},$$

where $N(0, \sigma_b)$ is Gaussian distribution with variance σ_b.

The remaining motion features are assumed to be constant with random noise. That is, for example,

$$(k_1)^{t+1} = (k_1)^t + N(0, \sigma_k).$$

Then the whole dynamics of motion is given by,

$$\hat{\mathbf{x}}_{t+1} = \Phi \hat{\mathbf{x}}_t + Q \equiv$$

$$\begin{bmatrix} \widehat{b}_1 \\ b_1^v \\ \widehat{b}_2 \\ b_2^v \\ \widehat{\theta} \\ \theta^v \\ \phi \\ k_1 \\ k_2 \end{bmatrix}^{t+1} = \begin{bmatrix} 1 & 1 & & & \dots & & & & 0 \\ & 1 & & & & & & & \\ & & 1 & 1 & & & & & \\ & & & 1 & & & & & \\ \vdots & & & & 1 & 1 & & & \vdots \\ & & & & & 1 & & & \\ & & & & & & 1 & & \\ & & & & & & & 1 & \\ 0 & & & & \dots & & & & 1 \end{bmatrix} \begin{bmatrix} \widehat{b}_1 \\ b_1^v \\ \widehat{b}_2 \\ b_2^v \\ \widehat{\theta} \\ \theta^v \\ \phi \\ k_1 \\ k_2 \end{bmatrix}^{t} + \begin{bmatrix} 0 \\ N(0,\sigma_b) \\ 0 \\ N(0,\sigma_b) \\ 0 \\ N(0,\sigma_\theta) \\ N(0,\sigma_\phi) \\ N(0,\sigma_k) \\ N(0,\sigma_k) \end{bmatrix} \quad (14)$$

where $\hat{\mathbf{x}} = [\widehat{b}_1, b_1^v, \widehat{b}_2, b_2^v, \widehat{\theta}, \theta^v \phi, k_1, k_2]$ is the state vector, Φ is the transition matrix and Q is process noise. The measurement model is given by

$$\mathbf{x}_t = H \hat{\mathbf{x}}_t + R \equiv$$

$$\begin{bmatrix} \widehat{b}_1 \\ \widehat{b}_2 \\ \widehat{\theta} \\ \phi \\ k_1 \\ k_2 \end{bmatrix}^{t} = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & & & & 0 \\ & 0 & 1 & 0 & 0 & & & & \\ \vdots & & 0 & 0 & 1 & 0 & & & \vdots \\ & & & 0 & 0 & 0 & 1 & & \\ & & & & 0 & 0 & 0 & 1 & \\ 0 & & & & \cdots & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \widehat{b}_1 \\ b_1^v \\ \widehat{b}_2 \\ b_2^v \\ \widehat{\theta} \\ \theta^v \\ \phi \\ k_1 \\ k_2 \end{bmatrix}^{t} + \begin{bmatrix} N(0,\sigma_b) \\ N(0,\sigma_b) \\ N(0,\sigma_\theta) \\ N(0,\sigma_\phi) \\ N(0,\sigma_k) \\ N(0,\sigma_k) \end{bmatrix} \quad (15)$$

where $\mathbf{z} = [\widehat{b}_1, \widehat{b}_2, \widehat{\theta}, \phi, k_1, k_2]$ is the observation, H is measurement matrix and R is the measurement noise. By Kalman filtering, the time update and measurement update are given in the following equations:

- Time update equations

$$\hat{\mathbf{x}}_t^- = \Phi \hat{\mathbf{x}}_{t-1}$$

$$\hat{P}_t^- = \Phi \hat{P}_{t-1} \Phi^T + Q$$

- Measurement update equations

$$K_t = \hat{P}_t^- H^T (H \hat{P}_t^- H^T + R)^{-1}$$

$$\hat{\mathbf{x}}_t = \hat{\mathbf{x}}_t^- + K_t (\mathbf{z}_t - H \hat{\mathbf{x}}_t^-)$$

$$\hat{P}_t = (I - K_t H) \hat{P}_t^-$$

where $\hat{P}_t^-$ ($\hat{P}_t$) is the *priori* (*posteriori*) estimate state error covariance, $\hat{\mathbf{x}}_t^-$ ($\hat{\mathbf{x}}_t$) is the *priori* (*posteriori*) state estimate, and K_t is Kalman gain matrix.

If carried out only in forward direction, Kalman filtering usually results in motion trajectory that is delayed to intentional movement, since only past data are used. To take into account the future data for motion smoothing, we employ a dual Kalman filtering that consists of two Kalman filtering in both forward and backward directions. Given the trajectory observations $\{\mathbf{z}\}$, we first use a forward Kalman filtering (described in Section V-*A*) to estimate the trajectory $\hat{\mathbf{x}}_f$ and covariance $\hat{P}_f$. Then we employ a backward Kalman filtering started with the last observation and carried on temporally backward, and have another estimated trajectory $\hat{\mathbf{x}}_b$ and covariance $\hat{P}_b$. These two Kalman filters are combined together by,

$$\begin{aligned} \hat{P} &= (\hat{P}_f^{-1} + \hat{P}_b^{-1})^{-1} \\ \hat{\mathbf{x}} &= \hat{P}(\hat{P}_f^{-1}\hat{\mathbf{x}}_f + \hat{P}_b^{-1}\hat{\mathbf{x}}_b) \end{aligned} . \qquad (16)$$

C. Combinational Stabilization of Multiple Objects

Kalman filtering, intuitively, should only be carried out for the segmented camera motion. Foreground motion can be discarded as useless visual cues. Nevertheless, the consumers may like to stabilize the foreground objects in some cases. For instance, stabilize with respect to a walking person or a moving car − the focus of video capture. In contrast to traditional DIS system, our approach has the capability of stabilizing multiple video objects selected by a consumer. Given two selected objects, we have trajectory observations $\{\mathbf{z}^a\}$ and $\{\mathbf{z}^b\}$ for object A and B respectively by the EM segmentation algorithm (Section IV). The combinational stabilizer is carried out as follows:

1. Estimate the intentional trajectory of $\{\mathbf{z}^a\}$ and $\{\mathbf{z}^b\}$ independently using the dual Kalman filtering as described in Section V-*B*. Then we have $\{\hat{\mathbf{x}}^a\}$, $\{\hat{\mathbf{x}}^b\}$, and corresponding covariance matrices $\{\hat{P}^a\}$ and $\{\hat{P}^b\}$.

2. Combine the two trajectory by transforming $\hat{\mathbf{x}}^b$ to $\hat{\mathbf{x}}^a$, then the new intentional trajectories of object A and B, $\overline{\mathbf{x}}^a$ and $\overline{\mathbf{x}}^b$, is given by,

$$\begin{aligned} \hat{P} &= ((c\hat{P}^a)^{-1} + ((1-c)\hat{P}^b)^{-1})^{-1} \\ \overline{\mathbf{x}}^a &= \hat{P}((c\hat{P}^a)^{-1}\hat{\mathbf{x}}^a + ((1-c)\hat{P}^b)^{-1}(\hat{\mathbf{x}}^b - \mathbf{z}^b + \mathbf{z}^a)) \\ \overline{\mathbf{x}}^b &= \overline{\mathbf{x}}^a - \mathbf{z}^a + \mathbf{z}^b \end{aligned} \qquad (17)$$

where $0 \le c \le 1$, c is a control parameter of the combinational stabilizer. If c is smaller, the object A would be more stable. While if c is larger, the object B will be more stable. The above procedure is carried out in a similar way if more than two objects are selected.

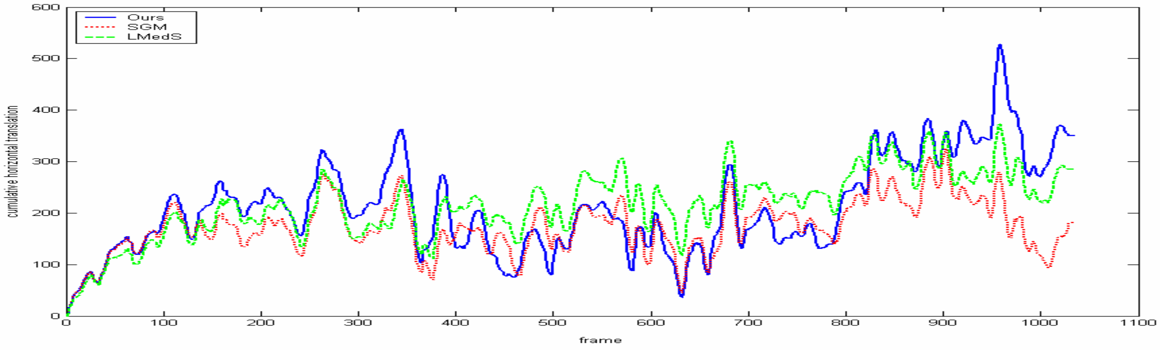


Fig. 2. The cumulative horizontal translations ($\widehat{b}_1$) estimated by ours, SGM and LMedS methods.

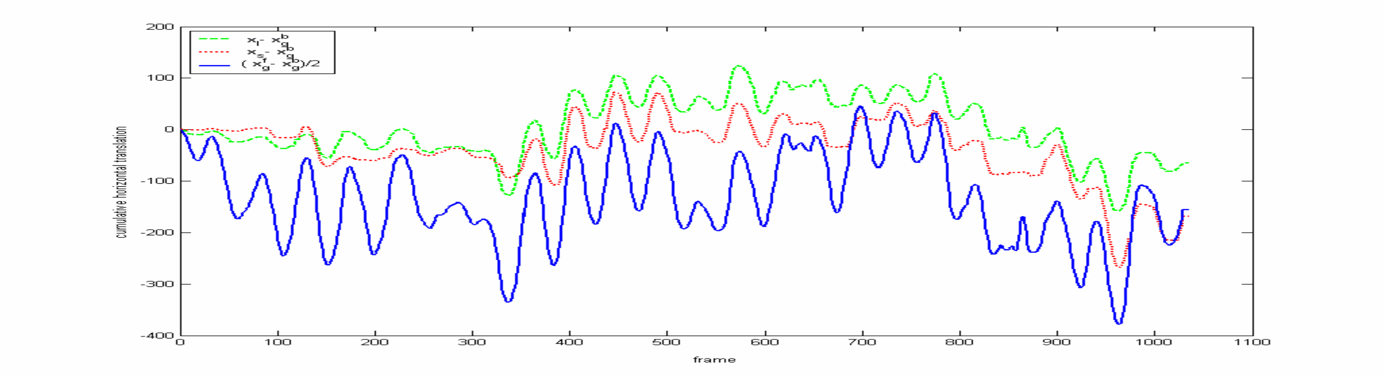


Fig. 3. The cumulative horizontal movements of EM foreground, SGM and LMedS background with respect to EM camera motion.

VI. Experiments

We test our selective object stabilizer on two video sequences: *Walking* and *Beach*. *Walking* captures a walking person in close distance while *Beach* is a shaky video captured by a person walking along the beach. At the beginning of *Walking*, the size of foreground object is relatively small, about half of the background regions. As the person moves closer and closer to the camera, the size of object gradually increases until about two times larger than the background. This video, technically presents intense challenge for traditional DIS due to large and unstable foreground object motion. In the remaining part, we will demonstrate, mainly with *Walking*, the advantages of our approaches in handling large moving objects, and in stabilizing with respect to different objects.

Generally speaking, the most crucial part of DIS system is motion estimation. The global motion between two successive frames must be estimated accurately, otherwise annoying discontinuities would be introduced in the stabilized videos. Since no ground-truth motions are available for real sequence and the testing by synthetic video is limited, we compare our approach with two other well-known motion estimation algorithms. The first approach is similar to our EM motion estimation, but assumes only single Gaussian model (SGM). In other words, only camera motion is assumed to be present. The motion parameters are estimated by maximizing the likelihood of two frames. The M-Estimator and Gaussian-Newton methods are used to solve the ML problem [10]. The second approach is based on RANSAC-like robust estimator least-median-of-square (LMedS) method. The feature matching of two frames is first estimated by optical flows (Sec. III), then LMedS is used to estimate the dominant motion model based on feature matching [20].

Fig. 2 shows the cumulative horizontal background translations estimated by SGM, LMedS and our methods. Although the fluctuations of the three trajectories look similar, the amplitudes are different. We observe that the differences between ours and other two methods are caused by the disturbance of foreground object movement. Thus the differences should follow the same fluctuation as the foreground object, that is

$$\begin{aligned} \mathbf{x}_s &\approx (\mathbf{x}_g^b + \mathbf{x}_g^f)/2 \\ \mathbf{x}_l &\approx (\mathbf{x}_g^b + \mathbf{x}_g^f)/2 \end{aligned} \tag{18}$$

where $\mathbf{x}_g^b$, $\mathbf{x}_s$ and $\mathbf{x}_l$ are background motions estimated by ours, SGM and LMedS methods respectively. $\mathbf{x}_g^f$ is the foreground motion by ours method. To verify this observation, we remove the influence of background by subtracting $\mathbf{x}_g^b$ from both sides of Eqn. (18). Fig. 3 shows the cumulative of $(\mathbf{x}_g^f - \mathbf{x}_g^b)/2$ (blue solid), $(\mathbf{x}_s - \mathbf{x}_g^b)$ (red dotted) and $(\mathbf{x}_l - \mathbf{x}_g^b)$ (green dashed). Note that the trajectories have similar fluctuations, which means the results of SGM and LMedS are impacted by the foreground objects. Fig. 3 also illustrates the influence of foreground object size. In the initial part of *Walking*, the object size is relatively small, and less influence is introduced. However, when the size of object gets larger, significant influence is observed.

The accuracy of EM motion estimation depends on the quality of the object layer segmentation. The well-known EM motion segmentation algorithm [10] is prejudicially in favor of the objects with relative large image region. Our modified EM segmentation algorithm, in contrast, can robustly segment the objects by taking advantage of good motion initialization (Sec. IV). Fig. 4 shows the segmentation results in two frames (first column). The first row shows the results with small foreground object, while the second row shows the results with a relatively large foreground object. The foreground object detections (in white color) by traditional EM [10] are shown in the second column, where homogenous regions in the relatively small object have incorrect assignment. In contrast, the results by our approach, shown in the third column, are more accurate.

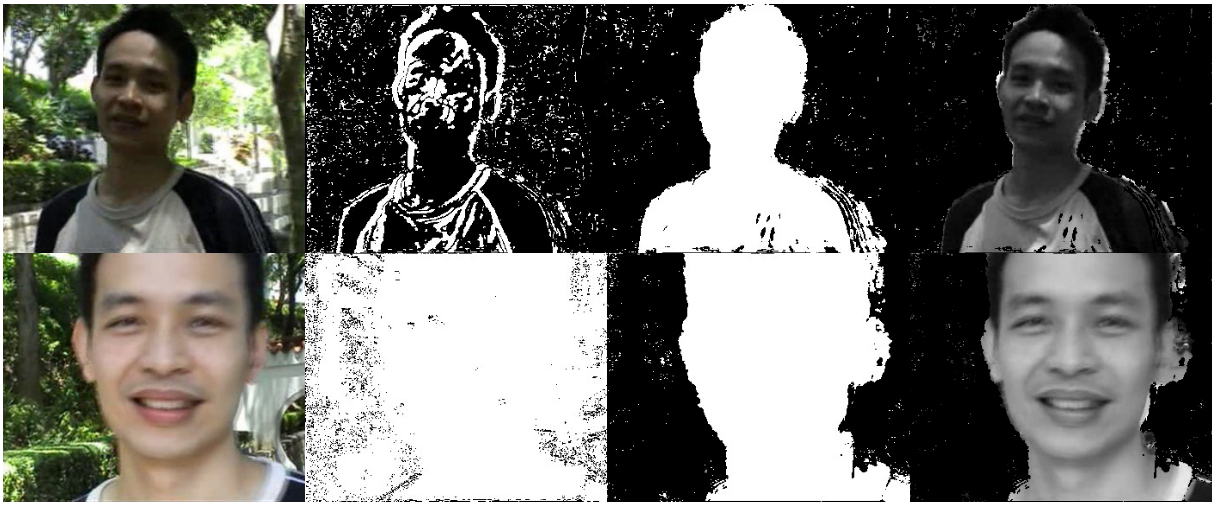


Fig. 4. Results of EM motion segmentation. Column 1: original frame. Column 2: the segmentation by traditional EM [10]. Column 3: the segmentation by our modified EM algorithm. Column 4: the segmented objects by our approach.

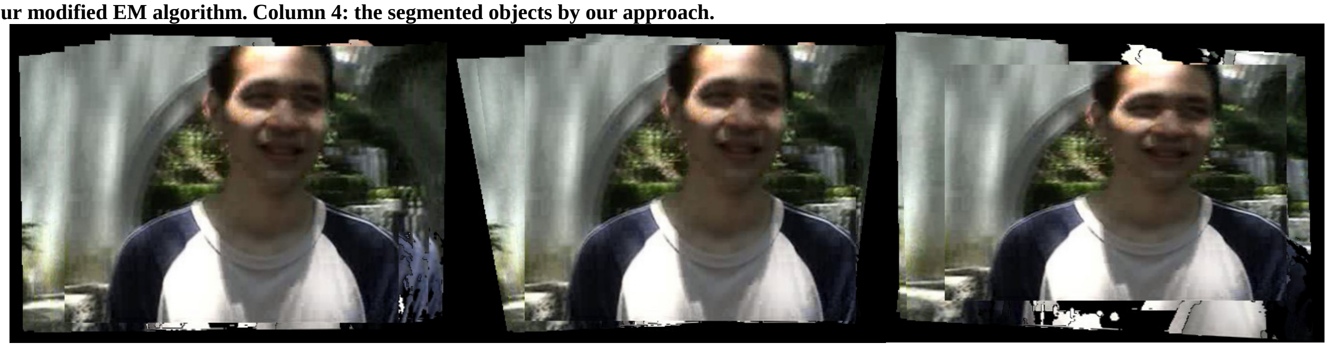


Fig. 5. From left to right: the image alignment results of ours, SGM and LMedS methods.

More visual comparison is illustrated in Fig. 5, which are the alignment results of multiple frames: 346-352, using ours, SGM and LMedS methods respectively. In these frames, the upper body swings in horizontal direction. The V-shape alignment of SGM (middle) obviously shows the impact of the body's movement. The right one shows the results of LMedS. In this mosaic, the background is discontinuous whereas the foreground is aligned. This is because relatively more features are detected in the body than in the white wall. Among the three methods, the best alignment result is achieved by our proposed motion estimation algorithm after taking into account the moving foreground objects, although, in some frames, undesirable effect of edge cascade might appear due to inaccurate segmentation in the edges of the foreground objects, as shown in the first column of Fig.5. In practice, this effect can be compressed by applying a temporal median filter near the edges of object.

A. Dual Kalman Smoothing

Fig. 6 and Fig. 7 show the intentional motion trajectory estimated by forward, backward and dual Kalman filtering (see Section V). The blue line is the observed cumulative horizontal translation movement. The red dashed curve is the estimated intentional motion by applying Kalman filtering in forward temporal dimension, while the green dotted curve is the result of backward Kalman filtering. The black curve is the final result which is estimated by the dual Kalman, the combination of forward and backward Kalman filtering. In Fig. 6 and Fig. 7, we observe that the trajectories estimated by either forward or backward Kalman filtering alone, are delayed to the intentional variation of movement, but in opposite directions. Therefore the forward and backward delays can be counteracted and eliminated by dual Kalman filtering, which is a linear combination of forward and backward stabilizer.

B. Selective Object Stabilizer

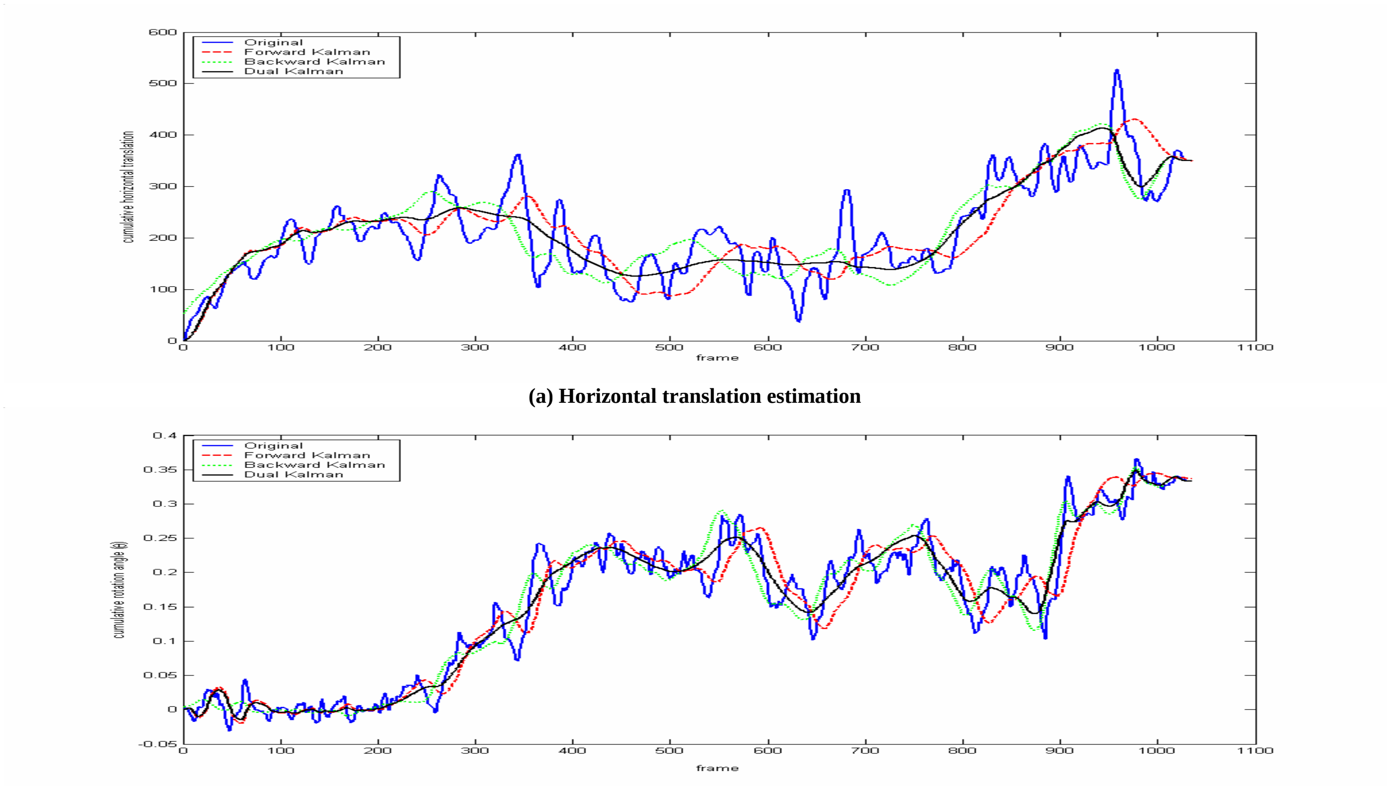
Allowing flexibility is a key feature from consumer point of view. Our approach has the flexibility of supporting selective object stabilization. A consumer can subjectively select and combine different objects for stabilization, and then pick the one which produces the best quality. In some cases, stabilizing background can lead to more shaking foreground. This is particularly worse if the ultimate goal of a consumer is to track moving foreground objects. Customers should be given priority to customize the stabilization effect as they wish. Fig. 8a (Fig. 8b) demonstrates the merit of our approach in supporting this facility. The blue curves in both subfigures are the original observed trajectory of horizontal translation, whereas the green dotted and red dashed curves are the estimated intentional trajectories by the dual Kalman stabilizer with respect to the foreground and background respectively. From the newly generated foreground trajectory in Fig 8b (background in Fig 8a) as a result of stabilizing background (foreground), we find that foreground object gets even unstable when stabilizing the background scene, and vice versa. . In this situation, our DIS system allows customized stabilization based on the consumer's selection.

For instance, we choose to stabilize both foreground and background by a combinational stabilizer (Eqn. (17)). By setting C =0.5, the stabilized trajectories are generated as depicted by black curves shown in Fig 8.

VII. CONCLUSION

We have presented a new approach for home video stabilization. Particularly, we take into account the effect of foreground moving objects. The novelty of our approach lies on the utilization of motion initialization in a sequence for the selection of initial frame in segmentation. This indeed leads to a robust EM based foreground and background segmentation algorithm that allows effective stabilization. Based on multiple motions and support layers estimated by EM algorithm, our selective home video stabilizer can stabilize the background as well as foreground objects. Furthermore, combinational stabilization algorithm has the flexibility of stabilizing arbitrary combination of objects selected by consumers.

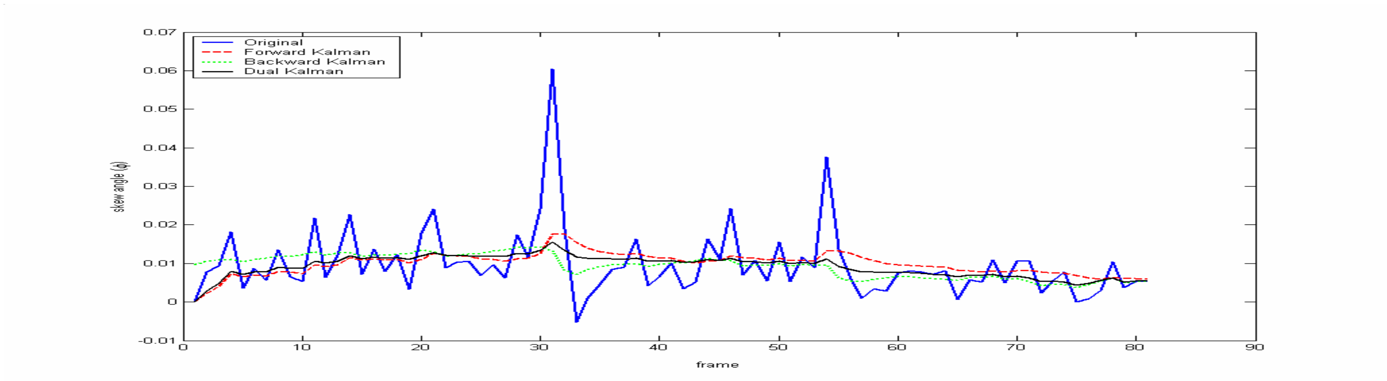
Currently, our approach suffers from the problem of "incomplete scene". A stabilized video frame may have holes especially at the image border due to the missing of scene information. Foreground moving objects may also encounter this problem if a camera swings rapidly. Part of the problems could probably be solved by image repairing techniques [8, 21]. Because home videos usually suffer from motion blur and lighting variation, effective construction of mosaics is another challenging issue for video stabilization.



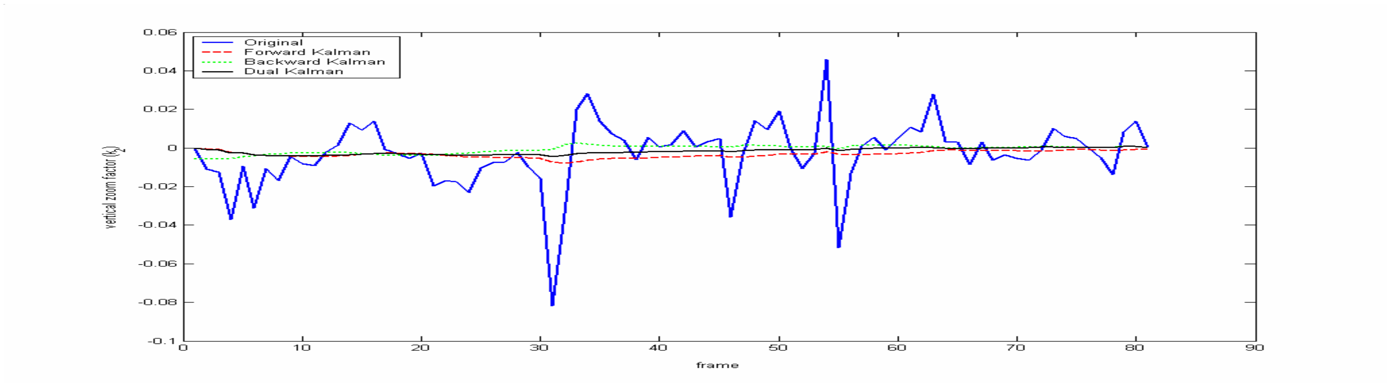
(a) Horizontal translation estimation

(b) Rotation angle estimation

Fig. 6. Intentional motion estimation of video *Walking* by forward, backward and dual Kalman filtering.

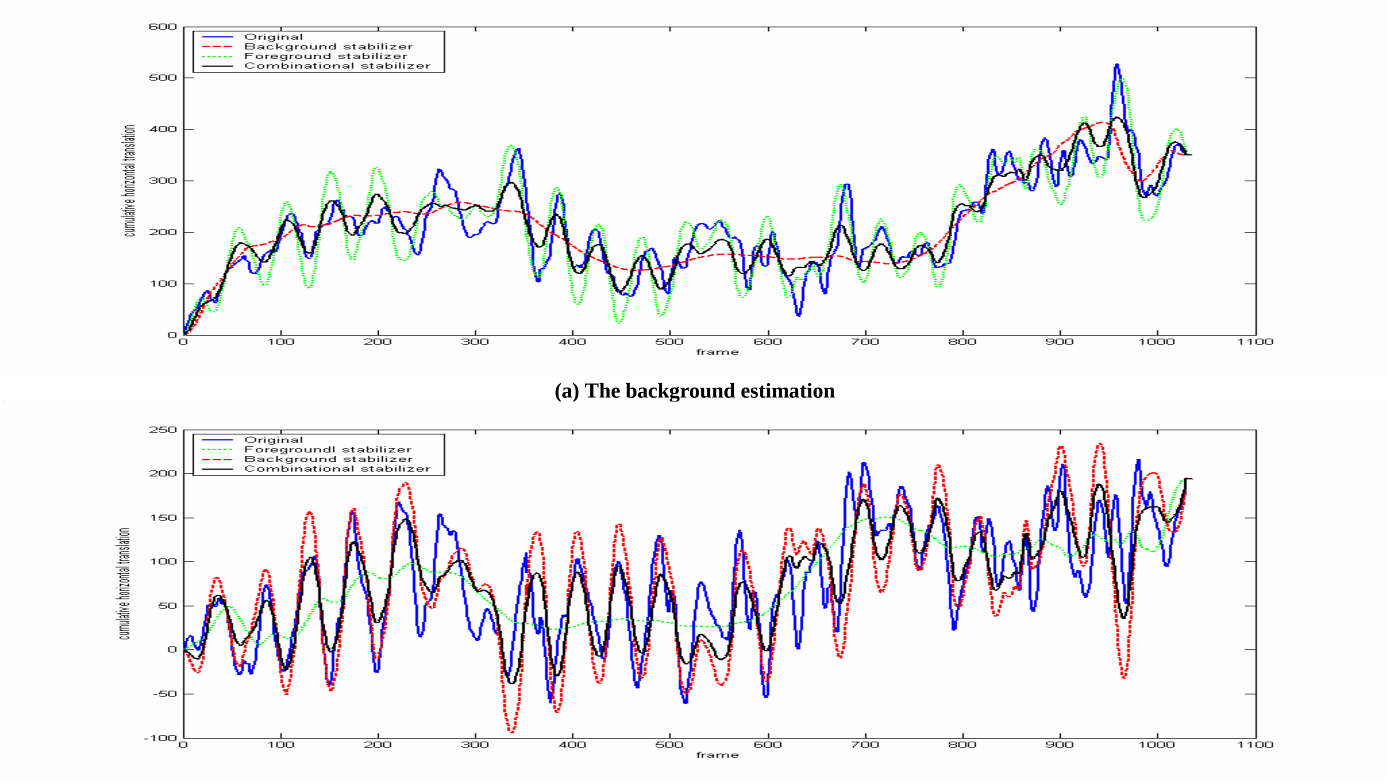


(a) Skew angle estimation



(b) Vertical zoom factor estimation

Fig.7 Intentional motion estimation of video *Beach* by forward, backward and dual Kalman filtering.



(a) The background estimation

(b) The foreground estimation

Fig. 8. The intentional motion estimation by dual and combinational Kalman filtering.

ACKNOWLEDGMENT

The work described in this paper was fully supported by a RGC grant CityU1072/02E (project No. 9040693) from the Research Grants Council of the Hong Kong SAR.

REFERENCES

[1] S. Ayer, P. Schroeter and J. Bigun, "Segmentation of moving objects by robust motion parameter estimation over multiple frames," *European Conf. on Computer Vision*, 1994

[2] A. Censi, A. Fusiello and V. Roberto, "Image stabilization by features tracking," *Int'l Conf. Image Analysis and Processing*, pp. 665-667, Sep. 1999.

[3] Z. Duric and A. Rosenfeld, "Shooting a smooth video with a shaky camera," *Machine Vision and Applications*, No. 5-6, pp. 303-313, 2003.

[4] A. Gelb et al. *Applied Optimal Estimation*. M.I.T. Press, 1974.

[5] M. Irani, B. Rousso and S. Peleg, "Recovery of ego-motion using image stabilization," *Int'l Conf. Computer Vision and Pattern Recognition*, pp. 454-460, 1994.

[6] C. Morimoto and R. Chellappa, "Fast 3D stabilization and mosaic construction," *Int'l Conf. Computer Vision and Pattern Recognition*, pp. 660-665, 1997.

[7] D. J. Lan, Y. F. Ma & H. J. Zhang, "A systematic framework for camera motion analysis for home video," *Int'l Conf. on Image Processing*, 2003.

[8] J. Jia, C. K. Tang, "Image repairing: robust image synthesis by adaptive ND tensor voting," *Int'l Conf. Computer Vision and Pattern Recognition*, 2003.

[9] P. Rousseeuw, *Robust Regression and Outlier Detection*. Wiley, New York, 1987.

[10] H. S. Sawhney and S. Ayer, "Compact representation of videos through dominant and multiple motion estimation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 18, No. 8, pp. 814-830, Aug. 1996.

[11] W. Q. Yan and M. S. Kankanhalli, "detection and removal of lighting & shaking artifacts in home videos," *ACM Multimedia*, 2002.

[12] C. Morimoto and R. Chellappa, “Fast electronic digital image stabilization,” *Int’l Conf. on Pattern Recognition*, vol. 3 pp. 284-288, 1996.

[13] Sung-Jea Ko, Sung-Hee Lee, Seung-Won Jeon and Eui-Sung Kang, “Fast digital image stabilizer based on gray-coded bit-plane matching, ” *IEEE Trans. Consumer Electronics*, vol. 45, no. 3, pp. 598 – 603, Aug. 1999.

[14] S. Erturk, “Digital image stabilization with sub-image phase correlation based global motion estimation,” *IEEE Trans. Consumer Electronics*, vol. 49, no. 4, pp. 1320-1325, Nov. 2003.

[15] F. Vella, A. Castorina. M. Mancuso and G. Messina “Digital image stabilization by adaptive block motion vectors filtering,” *IEEE Trans. Consumer Electronics*, vol. 48, no. 3, pp. 796 – 801, Aug. 2002.

[16] A. Engelsberg and G. Schmidt, “A comparative review of digital image stabilising algorithms for mobile video communications,” *IEEE Trans. Consumer Electronics*, vol. 45, no. 3, pp. 591-597, Aug. 1999.

[17] Jyh-Yeong Chang, Wen-Feng Hu, Mu-Huo Cheng and Bo-Sen Chang, “Digital image translational and rotational motion stabilization using optical flow technique,” *IEEE Trans. Consumer Electronics*, vol. 48, no. 1, pp. 108-115, Feb. 2002.

[18] 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, Jan. 2003.

[19] S. Erturk and T.J. Dennis, “Image sequence stabilisation based on DFT filtering,” *IEEE Proc. Vision, Image and Signal Processing*, vol. 147, no. 2, pp. 95-102, Apr. 2000.

[20] P. Meer, “Robust techniques for computer vision,” Emerging Topics in Computer Vision, G. Medioni and S. B. Kang (Eds.), Prentice Hall, 107-190, 2004.

[21] Y. Matsushita, E. Ofek, Tang Xiaoou and Shum Heung-Yeung, “Full-Frame Video Stabilization,” *Int’l Conf Computer Vision and Pattern Recognition* 2005.