http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet

Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 10 (2): 335-347, 2011
ISSN 1812-5638 / DOI: 10.3923/itj.2011.335.347
© 2011 Asian Network for Scientific Information

Robust Digital Image Stabilization Technique for Car Camera

Yuefei Zhang and Mei Xie
School of Electronic Engineering,
University of Electronic Science and Technology of China, Chengdu, Sichuan, 611731, China

Abstract: This research studies the digital image stabilization technique for the In-Car videos which are acquired from a car camera. Firstly, the relationship is established between the lane-line positions in a camera coordinate and an image plane. Then an analysis is performed to reveal the positions of the lane-lines in In-Car videos. Next, a digital image stabilization method for car cameras is proposed based on lane-line matching. This method begins with extracting the lane-lines from an In-Car video. Then, feature triangles are constructed to estimate the global inter-frame motions of the input video and a series of compensating motion vectors are yielded by using Kalman filter based algorithm with the inter-frame motions. Finally, repositioning the frames of the input video, according to the compensating motion vectors, can produce a stabilized In-Car video. The proposed method is resistant to the scene changes of In-Car videos. The experimental results, both for the simulated In-Car videos and the real ones, have demonstrated that the proposed method can robustly reduce the effects of undesired car camera motions on In-Car videos.

Key words: Digital image stabilization, global motion estimation, motion compensation, car camera, lane-line, feature triangle

INTRODUCTION

Intelligent vehicle techniques such as road detection technology (Yanqing *et al*., 2010) or face recognition technology (Ishak *et al*., 2006) have been applied, in recent years, to assist derivers in controlling vehicles. In particular, vision system plays an important role in an intelligent vehicle system, in which visual sensors or car cameras are fixed in front of a car to detect traffic condition. Unfortunately, the car cameras have to suffer from image or video instability due to the undesired camera motion or jitter caused by vehicle movements. Accordingly, such instability will degrade the quality of the acquired videos and affect the performance of subsequent processes such as video coding (Liang *et al*., 2004) or video surveillance (Marcenaro *et al*., 2001). Therefore, Digital Image Stabilization (DIS) has been proposed to remove the effect of undesired camera motion on videos.

A DIS system is typically composed of two major units: Global Motion Estimation (GME) and Motion Compensation (MC), as is shown in Fig. 1. The purpose of the first unit is to estimate the Global Motion Vectors (GMVs) (i.e., the inter-frame motions) between every two consecutive frames of the input video and the second unit reposition each frame of the video by an appropriate amount called compensating motion vector (CMV) to product a stabilized video. In fact, the CMVs are calculated according to GMVs.

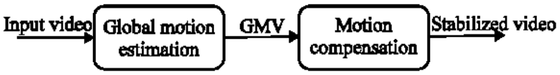

Fig. 1: The typical composition of a DIS system

GME unit usually plays the most important role in a DIS system and essentially determines the performance of the system. Therefore, various DIS algorithms concentrated on GME have been proposed which can be divided into two categories: two-dimensional (2D) algorithm and three-dimensional (3D). 2D algorithms can only estimate translational inter-frame motions which are mostly encountered. Block Matching Algorithm (BMA) (Erturk, 2003) is the most popular 2D algorithm, in which some Local Motion Vectors (LMVs) are first generated through sub-image matching and then GMVs can be obtained by applying a median filter on the LMVs. Project Curve Matching (PCM) algorithm (Bosco *et al*., 2008), Represent Point Matching (RPM) (Hsu *et al*., 2007) and Bit-Plane Matching (BPM) (Ko *et al*., 1999) are other popular 2D algorithms. Unlike 2D algorithms, 3D algorithms can estimate the global inter-frame motion containing both rotation and translation through two

Corresponding Author: Yuefei Zhang, School of Electronic Engineering, University of Electronic Science and Technology of China, No. 2006, Xiyuan Road, West High-Tech Developing Zone, Chengdu, Sichuan, 611731, China
Tel: 86-28-83205886 Fax: 86-28-83205886

major steps: (1) extracting image features from each frame of an input video and (2) solving a specific geometrical model with the features to produce GMVs. The greatest challenge of 3D algorithms is that image features can be robustly extracted and tracked frame by frame. Consequently, the performance of 3D algorithm essentially depends on the characteristics of images features and different features are suitable for different applications. SIFT algorithm has been utilized to detect the image features for image stabilization (Battiato *et al.*, 2007). This method is highly invariant to illumination changes and image transformations. A 3D DIS method adopting edges as image features has been proposed which works well for the videos with robust edge information (Lingqiao *et al.*, 2008). Another DIS method utilizes Good Feature Points (GFPs) to estimate inter-frame motions, which is suitable for the videos with fixed background (Wang *et al.*, 2009).

This study focus on the DIS technique for the In-Car videos which are acquired from the cameras mounted on the vehicles running on freeway. As a matter of fact, few DIS researches are specific to In-Car videos and the methods in current literatures directly apply some existing DIS methods to In-Car videos. Some 2D DIS methods, such as RPM (Hsu *et al.*, 2007) and BPM (Zhang *et al.*, 2010), have been utilized to deal with the effect of undesired camera motions on In-Car videos. These methods can remove the undesired inter-frame translations of In-Car videos. However, the image instabilities appearing in In-Car videos often contain rotations, in which case the performances of 2D methods would be not as well as in the translational only case. A 3D DIS method (Morimoto and Chellappa, 1996), for In-Car videos, has been presented which employs Good Tracking Points (GTP) as image features. A feature tracking scheme is introduced to this method, which keep the image features can be tracked robustly. However, the scenes or backgrounds of In-Car videos always change along with vehicle movement, especially when moving objects (such as moving cars) appear on adjacent lanes. As a result, the image features will be mismatched, which will reduce the performance of this method. In order to robustly remove the effects of undesired camera motions on In-Car videos, we proposed a digital image stabilization method based on lane-line matching. The lane-lines of In-Car videos, in our method, are chosen as the image features to estimate inter-frame motions. This is because that such image features can be extracted from In-Car videos robustly and are hardly affected by environment, especially on freeway.

The algorithms for MC units, in current literatures, can be divided into two kinds: Motion Vector Integrate (MVI) (Erturk, 2001b; Hsu *et al.*, 2007; Yang *et al.*, 2007) and Frame Position Smooth (FPS) (Erturk, 2003; Wang *et al.*, 2009; Yang *et al.*, 2009). The first kind of MC method produces CMVs through smoothing or low-pass filtering GMVs directly and the second one through three steps: (1) accumulating GMVs to generate a Frame Position Signal (FPS), (2) smoothing the FPS to generate a Smoothed Frame Position Signal (SFPS) and (3) subtracting the SFPS from the FPS to generate a series of CMVs. It should be emphasized that our study are mainly concentrated on the GME for In-Car video stabilization, so we directly adopt a popular MC method: Kalman Filter Based (KFB) algorithm (Erturk, 2001a), to produce the Compensating Motion Vectors (CMVs) in the proposed method.

THE POSITIONS OF THE LANE-LINES IN IN-CAR VIDEOS

Here, we analyze the positions of the lane-lines in stable In-Car videos and unstable ones. To do this, the coordinate systems for car camera and image plane are established first of all. Then, the positions of the lane-lines in a stable In-Car video and in an unstable one are provided based on the coordinate systems.

The coordinate systems for car camera and image plane: We first establish the coordinate system CS$_c$: (X$_c$, Y$_c$,Z$_c$) of a camera, as is shown in Fig. 2. The X$_i$-Y$_i$ plane is the image plane of the camera and parallel to the X$_c$-Y$_c$ plane. The Z$_c$-axis pointing forward has the same direction with the camera view and pass through the origin of the image plane. Some parameters about the camera configuration are known beforehand, including the height h of the camera and the focal length f.

A car camera is generally mounted in the front of a vehicle, which is illustrated in Fig. 3. Consider that the

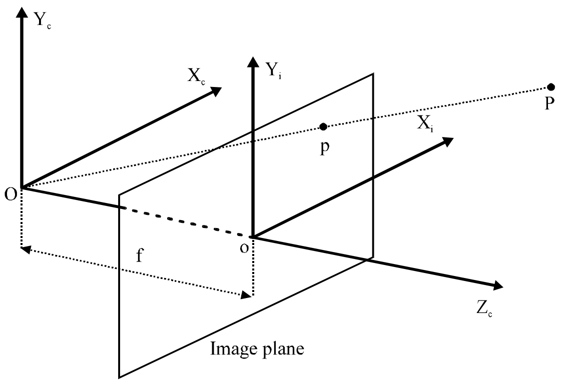

Fig. 2: The coordinate systems for camera and image plane

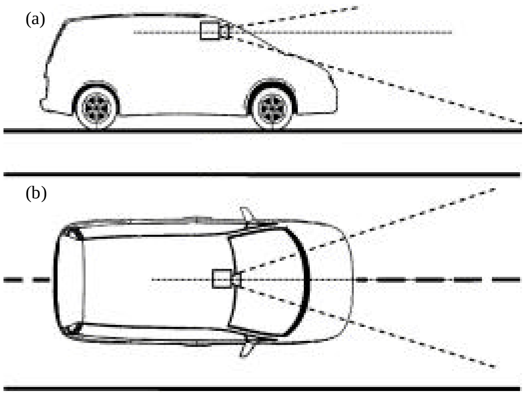

Fig. 3: The setup of car camera

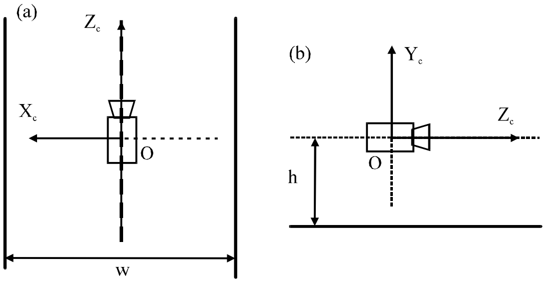

Fig. 4: The relationship between the real world and the coordinate system for car camera

vehicle moves steadily on a freeway. The relationship between the camera coordinate system and the real world is presented in Fig. 4a and b. The Z_c-axis is aligned at the middle of the lane and the Y_c-axis pointing upward is perpendicular to the road. In particular, the height h of the camera is constant in the stable case. Rather than fixed, the camera coordinate system moves along the middle line at the same speed as the vehicle. In this way, the car camera can be simplified by using the coordinate system CS_c together with the image plane X_i-Y_i.

The positions of lane-lines in stable In-Car videos and unstable ones: Video capture indeed is the mapping from the points (x_c,y_c) in CS_c onto an image plane frame by frame, which is depicted by perspective projection (Bas and Crisman, 1997):

$$\begin{cases} x_i = f\frac{x_c}{z_c} \\ y_i = f\frac{y_c}{z_c} \end{cases} \tag{1}$$

where, x_i and y_i are the coordinates of the point (x_c, y_c) in the image plane or frame. Consider the case that a car camera is stable. Let, p_1 and p_2 be two points of a lane-line. Accordingly, the coordinates in CS_c of $p_1$and p_2 at time n are:

$$(\frac{w}{2},-h,z_1) \text{ and } (\frac{w}{2},-h,z_2)$$

respectively. Substituting the coordinates into Eq. 1 can produce the coordinates of $p_1$and p_2 in the image plane at time n (i.e., the coordinates in frame n):

$$p_1:(f\frac{w}{2z_1},-f\frac{h}{z_1}) \text{ and } p_2:(f\frac{w}{2z_2},-f\frac{h}{z_2})$$

Consider that the vehicle as well as the car camera has moved by ΔZ along the Z_c axis, when the capture of the next frame (i.e., frame n+1) start. Accordingly, the coordinates of p_1 and p_2 in the image plane at time n+1 (i.e., the coordinates in frame n+1) become:

$$(f\frac{w}{2(z_1-\Delta Z)},-f\frac{h}{(z_1-\Delta Z)}) \text{ and } (f\frac{w}{2(z_2-\Delta Z)},-f\frac{h}{(z_2-\Delta Z)})$$

respectively. Let, the equation of a line be:

$$\frac{y-y_1}{y_2-y_1}=\frac{x-x_1}{x_2-x_1} \tag{2}$$

where, (x_1, y_1) and (x_2, y_2) are two distinct points of the line. From the coordinates of p_1 and p_2 and Eq. 2, the lane-lines in frames can be determined as follows:

$$\begin{cases} L_1: \dfrac{y+f\frac{h}{z_1}}{-f\frac{h}{z_2}+f\frac{h}{z_1}} = \dfrac{x-f\frac{w}{2z_1}}{f\frac{w}{2z_2}-f\frac{w}{2z_1}} \\ L_2: \dfrac{y+f\frac{h}{z_1-\Delta Z}}{-f\frac{h}{z_2-\Delta Z}+f\frac{h}{z_1-\Delta Z}} = \dfrac{x-f\frac{w}{2(z_1-\Delta Z)}}{f\frac{w}{2(z_2-\Delta Z)}-f\frac{w}{2(z_1-\Delta Z)}} \end{cases} \tag{3}$$

where, L_1 and L_2 are the lane-lines in frame n and frame n+1, respectively. By simplifying Eq. 3, we get:

$$\begin{cases} L_1: y=-\frac{h}{w} \\ L_2: y=-\frac{h}{w} \end{cases} \tag{4}$$

These two equations indicate that the lane-lines of two consecutive frames of a stable In-Car video always appear in a fixed position.

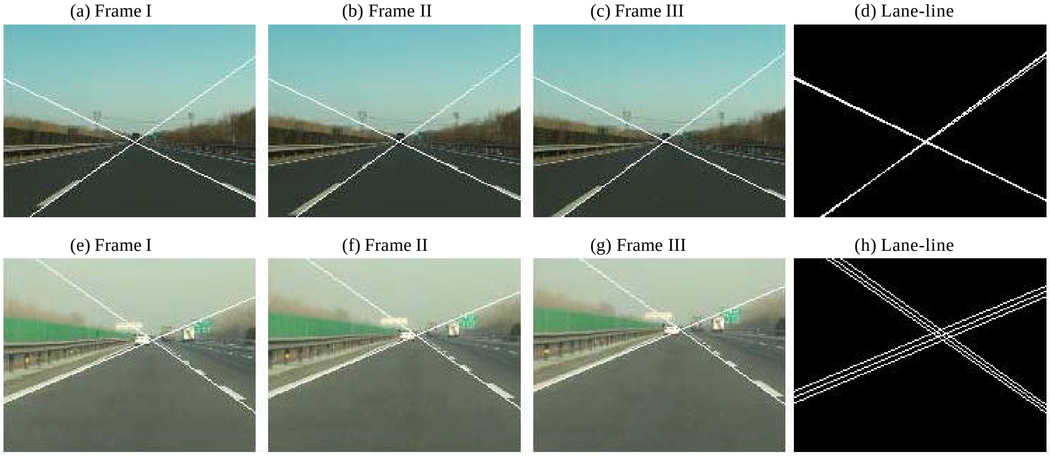

Fig. 5: The lane-line positions of a stable In-Car video and an unstable. (a-c) are three consecutive frames of a stable In-Car video and (d) the lane-lines of such frames are drawn together. (e-g) are three consecutive frames of an unstable In-Car video and the lane-lines of such frames are drawn together in (d). The lane-lines are denoted by the white lines

On the other hand, camera instability will lead to not only the undesired inter-frame motions of In-Car videos but also the undesired motions of lane-lines. Consequently, the lane-lines in unstable frames will deviate from the fixed positions. Figure 5 illustrates the positions of the lane-lines in a stable In-Car video and an unstable one, in which (Fig. 5a-c) are three consecutive frames of a stable In-Car video and (Fig. 5e-g) an unstable. The lane-lines (i.e., the white lines) of the stable video are drawn together in Fig. 5d and the unstable in Fig. 5h. Clearly, the lane-lines of the stable video appear in a fixed position, while the unstable in different positions. Hence, we assume that the global motions of lane-lines reveal the undesired inter-frame motions of In-Car videos. In other words, the GME for In-Car video stabilization can be performed through estimating the global motions between the lane-lines of every two consecutive frames of an In-Car video.

EXTRACTING LANE-LINES FROM IN-CAR VIDEO

The proposed method begins with the lane-line extraction from an In-Car video. Figure 6a illustrates a sample frame of an In-Car video, where the lane-lines look like strips with an orientation of about 45 degree and most of them appear in the lower part of the image. Consequently, the lane-line detection of the proposed method is performed on the lower half of the image and the detection area is divided into two independent Sub-Detection-Areas (SDAs) for two lane-lines, as is shown in Fig. 6a.

Within each SDA, we compute the convolution of the input image I(x, y) and a Sobel operator (Forsyth and Ponce, 2003) S(i, j) with the size of $W_S \times H_S$, as Eq. 5:

$$R(x,y)=\sum_{i=1}^{W_s}\sum_{j=1}^{H_s} I(x+i-c_i, y+j-c_j)S(i,j) \tag{5}$$

where, (c_i, c_j) denotes the center in the coordinate system of S(i,j). In the next step, the result image R is thresholded through setting some pixels of R to 1 and the others to 0, as follows:

$$B(x,y)=\begin{cases}1 & \text{if} \quad R(x,y)>T\\ 0 & \text{otherwise}\end{cases}$$
$$T=\frac{1}{WH}\sum_{x=1}^{W}\sum_{y=1}^{H}R(x,y) \tag{6}$$

where, T is the threshold, B is the binary image resulting from Eq. 6, W and H are the width of R and the height, respectively. Figure 6 b illustrates the result of above process, in which the lane-line edges have been extracted. However, it can be observed that some small objects still exist which do not belong to any lane-lines. Hence, the detection is performed for the sizes of all connected objects of B and then the connected objects which are smaller in the size than 10 pixels are eliminated. The result of this process is illustrated in Fig. 6c, in which each lane-line has two edges: the inner edge and the outer edge. Only the inner-edges, as a matter

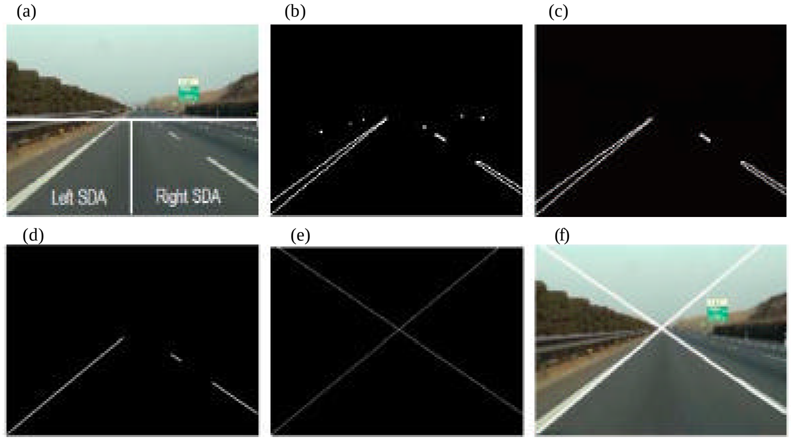

Fig. 6: The procedure of extracting the lane-lines from an In-Car video. (a) Sample frame, (b) binary image, (c) edges of lane-line, (d) inner-edges, (e) lane-lines and (f) lane-lines on road

of fact, are desired in the proposed method, since the inner-edges are sufficient to represent the lane-lines. Consequently, inner-edge detection is made row by row in each SDA of Fig. 6c. The points of 1 close to the right side, for each row of the left SDA, are marked as the inner-edge points of the left lane-line. Meanwhile, the points of 1 close to the left side, for each row of the right SDA, are marked as the inner-edge points of the right lane-line. Figure 6d shows the inner-edges of each lane-line.

Next, we apply Hough Transform (HT) (Forsyth and Ponce, 2003) on each SDA of Fig. 6d to find two straight lines: l_1 and l_2 that best fit the inner-edges. As a result, the parameters: $\{\{a_i, b_i\}, i = 1,2\}$ of l_1 and l_2 can be obtained, in which a_i and b_i are the slope of l_1 and the intercept, respectively. According to Eq. 7, the points (x, y) of l_1 and l_2 are computed which are drawn in Fig. 6e.

$$\begin{cases} l_1 : y = a_1 x + b_1 \\ l_2 : y = a_2 x + b_2 \end{cases} \qquad x \in [1, W] \tag{7}$$

Finally, the intersection $V:(v_x, v_y)$ of l_1 and l_2 is computed as Eq. 8.

$$\begin{cases} v_x = \dfrac{b_1 - b_2}{a_1 - a_2} \\ v_y = \dfrac{a_2 b_1 - a_1 b_2}{a_1 - a_2} \end{cases} \tag{8}$$

In particular, it should be indicated that this intersection is the vanishing point of the lane-lines. Combining l_1, l_2 and V can depict the lane where the vehicle travels, as shown in Fig. 6f.

GLOBAL MOTION ESTIMATION BASED ON LANE-LINES MATCHING

As mentioned earlier, the GME for In-Car videos can be performed through estimating the global motions of the lane-lines extracted from every two consecutive frames. To do this, we construct feature triangle (FT) serving as the simplified form of lane-lines. Figure 7a illustrates the FT, in which one vertex of FT is the vanishing point (i.e., the red point) of lane-lines and the other two (i.e., the yellow points) are determined by Eq. 9:

$$\begin{cases} y_i = \min\{v_y + \dfrac{L|a_i|}{\sqrt{1+a_i^2}}, v_y - \dfrac{L|a_i|}{\sqrt{1+a_i^2}}\} \\ x_i = \dfrac{y_i - b_i}{a_i} \end{cases} \tag{9}$$

where, L is the distance between the vanishing point and the other vertices and is set to one quarter of the image width. Consequently, the purpose of the GME for In-Car videos is converted into finding out the global motion between the FTs of every two consecutive frames.

The global inter-frame motions usually contain rotation and translation, which can be characterized by using rigid transform model. Equation 10 presents such model:

$$\begin{cases} x_n = x_{n-1} \cos\theta_n - y_{n-1} \sin\theta_n + T_{xn} \\ y_n = x_{n-1} \sin\theta_n - y_{n-1} \cos\theta_n + T_{yn} \end{cases} \tag{10}$$

where, θ_n denotes the rotation of frame n with respect to its previous frame and (T_{xn}, T_{yn}) the translation. The (T_n, T_n) denotes the points of frame n. In fact, the

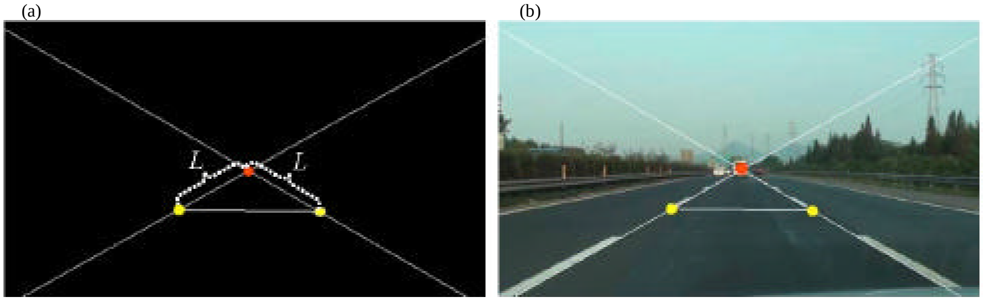

Fig. 7: The feature triangle and the lane-lines. The points are the vertices of the feature triangle, in which the red point is the vanishing point of the lane-lines and the yellow ones are the vertices generated by using Eq. 9. (a) The feature triangle (FT) and (b) the feature triangle on road

objective of GME for DIS is to obtain the motion parameters: {θ_n, T_{xn}, T_{yn}) which can be achieved by solving the transform model in Eq. 10 with some feature points. We select the middle points of the FT edges along with the vertices as the feature points and then substitute these points into Eq. 10 to produce a linear system as Eq. 11:

$$\begin{pmatrix} x_n^1 \\ y_n^1 \\ \vdots \\ x_n^i \\ y_n^i \\ \vdots \\ x_n^6 \\ y_n^6 \end{pmatrix} = \begin{pmatrix} 1 & 0 & -y_{n-1}^1 & x_{n-1}^1 \\ 0 & 1 & x_{n-1}^1 & y_{n-1}^1 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & -y_{n-1}^i & x_{n-1}^i \\ 0 & 1 & x_{n-1}^i & y_{n-1}^i \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & -y_{n-1}^6 & x_{n-1}^6 \\ 0 & 1 & x_{n-1}^6 & y_{n-1}^6 \end{pmatrix} \begin{pmatrix} T_{xn} \\ T_{yn} \\ \sin\theta_n \\ \cos\theta_n \end{pmatrix} \qquad (11)$$

where, (x_ni, y_n^i) is the ith selected feature point from frame n. Solving this system can produce the GMVs: {θ_n, T_{xn}, T_{yn}} between frame n and its previous frame. It should be noted that Eq. 11 is an over-determined system which has no exact solution. Generally, least square method (Howard-Anton, 2005) can be used to yield an approximation solution through solving the normal system defined as Eq. 12:

$$(Tx_n \quad Ty_n \quad \sin\theta_n \quad \cos\theta_n)^T = \left(A^T A\right)^{-1} A^T b \qquad (12)$$

where, A is the coefficient matrix on the right of Eq. 11 and b the constant vector on the left. Finally, the motion parameters: {arccsin θ_n, T_{xn}, T_{yn}} are the global motion vectors of frame n with respect to its previous frame.

MOTION COMPENSATION

Here, Kalman Filter Based (KFB) method (Erturk, 2001a) is utilized to generate the Compensating Motion Vectors (CMVs) in the proposed method. For this purpose, GMVs are accumulated to yield a frame position vs. Frame Number Signal (FPS), as follows:

$$FPS(n) = \sum_{i=1}^{n} GMV(i) \quad n = 1 \cdots N-1 \qquad (13)$$

where, GMV(i) is the global motion vector between frame i and i-1 and N is the frame number of the input video. This signal is then low-pass filtered or smoothed to remove the high-frequency component while retain the low-frequency one which is considered as the stable frame position. The CMV that we need can be generated as follows:

$$CMV(n) = FPS(n) - SFPS(n) \qquad (14)$$

where, SFPS(n) is the stable or smoothed frame position signal (SFPS).

A time-invariant linear system, in KFB method, is established to estimate the SFPS, as is shown in Eq. 15.

$$\begin{cases} X(n) = AX(n-1) + W(n) \\ Z(n) = HX(n) + V(n) \end{cases} \qquad (15)$$

The X(n) in the top equation (i.e., the state equation) denotes the estimated stable position of frame n and the Z(n) in the bottom equation (i.e., the observation equation) the observed position (i.e., the FPS(n)). A and H are state transform matrix and measurement matrix, respectively. W~N(0,Q) and V~N(0,R) are process noise and measurement noise respectively, which are assumed to be the white Gaussian noise that are independent of each other. For a DIS system, the state equation is constructed in the form of a constant velocity model:

$$\begin{pmatrix} EP_{Tx}(n) \\ EP_{Ty}(n) \\ EP_{\theta}(n) \\ dx(n) \\ dy(n) \\ d\theta(n) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} EP_{Tx}(n-1) \\ EP_{Ty}(n-1) \\ EP_{\theta}(n-1) \\ dx(n-1) \\ dy(n-1) \\ d\theta(n-1) \end{pmatrix} + \begin{pmatrix} W_x(n) \\ W_y(n) \\ W_{\theta}(n) \\ W_{dx}(n) \\ W_{dy}(n) \\ W_{d\theta}(n) \end{pmatrix} \quad (16)$$

where, {$EP_{Tx}(n)$, $EP_{Ty}(n)$, $EP_{\theta}(n)$} is the estimated stable frame position of frame n; dx(n), dy(n) anddθ(n) are the velocities of $EP_{Tx}(n)$, $EP_{Ty}(n)$ and $Ep_{\theta}(n)$ at time n, respectively. The observation equation can be constructed as follow:

$$\begin{pmatrix} FPS_{Tx}(n) \\ FPS_{Ty}(n) \\ FPS_{\theta}(n) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} EP_{Tx}(n-1) \\ EP_{Ty}(n-1) \\ EP_{\theta}(n-1) \\ d_{Tx}(n-1) \\ d_{Ty}(n-1) \\ d_{\theta}(n-1) \end{pmatrix} + \begin{pmatrix} V_x(n) \\ V_x(n) \\ V_x(n) \\ V_{dx}(n) \\ V_{dy}(n) \\ V_{d\theta}(n) \end{pmatrix} \quad (17)$$

where,{$FPS_{Tx}(n)$, $FPS_{Ty}(n)$, $FPS_{T\theta}(n)$} is the observed frame position at time n.

Indeed the KFB method works iteratively in two stages: predication stage and correction stage. The first stage falls into two steps: (1) Estimate the priori state X_n^- from time n-1 to n according to Eq. 16 and (2) estimate the priori estimate error covariance P_k^- at time n as follows:

$$P_n^- = AP_{n-1}A^T + Q \quad (18)$$

where, Q is the process noise covariance and P_{n-1} is the posteriori estimate error covariance at time n-1. The correction stage can be divided into three steps: (1) Compute the Kalman gain K_n at time n as follows:

$$K_n = P_n^- H^T (HP_n^- H^T + R)^- \quad (19)$$

where, R is the measurement noise covariance; (2) Compute the posteriori state X_n^+ at time n as follows:

$$X_n^+ = X_n^- + K_n(Z_n - HX_n^-) \quad (20)$$

where, Z_n is the observed state at time n computed from Eq. 17; (3) Compute the posteriori estimate error covariance at time n by Eq. 21:

$$P_n = (I - K_n H)P_k^- \quad (21)$$

where, I is an identity matrix. After each prediction and correction pair, above process is repeated with the previous a posteriori estimate used to predict the new a priori estimate.

As a matter of fact, the posteriori states X_n^+ are the stable or smoothed frame position that we desire. Correcting the frame positions of the input video, according to the CMVs from Eq. 14, can produce the stabilized video.

RESULTS

The performance of the proposed DIS method was evaluated through two schemes. Three simulated unstable In-Car videos of different categories, in the first scheme, were constructed to evaluate the performance of the proposed method. In the second scheme, four real unstable In-Car videos captured at different segments of a freeway were utilized to evaluate the performance of the proposed method.

Simulated case: In order to construct the simulated videos, a stable In-Car video was captured by using the camera mounted on the vehicle which moved steadily on a freeway. Next, adding artificial jitter to each frame of the acquired video can produce a simulated unstable In-Car video. In fact, the Artificial Jitter Vectors (AJVs) are the real Compensating Motion Vectors (CMVs), since repositioning each frame of a simulated video, according to the AJVs, can produce a stable video. Three categories of artificial jitters, in our experiments, were considered: (1) translational motion only; (2) rotational motion only and (3) compounded motion with rotation and translation. Figure 8 shows the sample frames of the simulated In-Car videos of different categories, in which SVR, SVT and

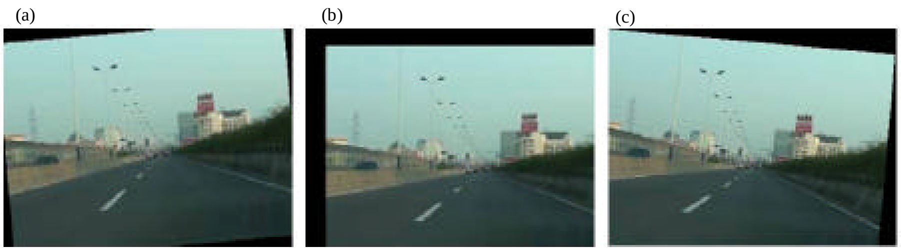

Fig. 8: The sample frames of the simulated In-Car videos of different categories. (a-c) are the frames of the videos of rotational only category, translational only category and compounded category, respectively

Table 1: The MSE in the rotational motion case

DIS	Our method	EMA	PMA
MSE (θ): SVR	0.1009	0.6412	0.9539

Table 2: The MSE in the translational motion case

DIS	Our method	BMA	PCM
MSE (Tx): SVT	1.4870	125.0951	16.3625
MSE (Ty): SVT	1.3674	14.7775	18.1250

Table 3: The MSE in the compounded motion case

DIS	Our method	EMA	PMA
MSE (Tx): SVC	1.5779	15.0773	30.8423
MSE (Ty): SVC	0.2122	6.9384	7.3090
MSE (θ): SVC	0.1794	1.1652	1.3647

SVC are the videos of rotational only category, translational only category and compounded category, respectively. The black parts are the undefined areas resulting from the artificial jitters.

Firstly, the mean square error (MSE), between the real CMVs (i.e., the AJVs) and the CMVs obtained by using a DIS method, was computed to measure the accuracy of the proposed method result. Eq. 22 gives the MSE, in which N is the frame number of the input video.

$$\mathrm{MSE}=\frac{1}{\mathrm{N-1}}\sum_{\mathrm{i=1}}^{\mathrm{N-1}}[\mathrm{AJV(i)-CMV(i)}]^2 \qquad (22)$$

Table 1-3 present the evaluation results for the simulated videos of different categories. In addition, the results of four classical DIS methods (i.e., BMA (Zhang *et al*., 2010), PCM (Bosco *et al*., 2008), Point Matching Algorithm (PMA) (Morimoto and Chellappa, 1996) and Edge Matching Algorithm (EMA) (Lingqiao *et al*., 2008)) are presented together. From Table 1-3, we can observe that the results of the proposed method have smaller MSEs than other DIS methods in either case. This experimental result indicates that the CMVs generated by using the proposed method are more close to the real CMVs than those of other DIS methods. In other words, the proposed method, for In-Car videos, has more accurate results than other DIS methods.

Next, Absolute Difference (AD) curves were computed to evaluate the robustness of the proposed method, as is shown in Eq. 23:

$$\mathrm{AD}=\left|\mathrm{AJV(i)-CMV(i)}\right| \quad \mathrm{i}=1\cdots\mathrm{N}-1 \qquad (23)$$

Figure 9 presents the evaluation results, in which the comparisons are made between the results of the proposed method and other classical DIS methods. Figure 9a gives the results for SVR of our method (i.e., the black line), PMA (i.e., red line) and EMA (i.e., the blue line); Fig. 9b and c give the results for SVT of our method (i.e., the black line), BMA (i.e., red line) and PCM (i.e., the blue line); Fig. 9d-f give the results for SVC of our method (i.e., the black line), PMA (i.e., red line) and EMA (i.e., the blue line). We can observe that the results of the proposed method are always close to zero for each simulated video. In contrast, the AD curves of other DIS methods, for some frames, are significant above zero. This is because that the scene changes of the In-Car videos affect the performances of these methods, which was further discussed in the following. Consequently, we can conclude that the proposed method works robustly for the simulated In-Car videos of different categories.

The frames 28, 33, 38, 43 and 48 of SVT are further provided and the stabilized SVTs generated by using different DIS methods, as is shown in Fig. 10. With the aid of the red lines in Fig. 10a, we can observe that the artificial jitters have resulted in obvious instability of the frames in SVT. The results of the proposed method are illustrated in Fig. 10b. It can be seen that the frames generated by the proposed method, compared with Fig. 10a, have little undesired frame jitter. On the other hand, we can observe that there are still obvious undesired translations in the results of BMA and PCM, as is shown in Fig. 10c and d. This is because that the appearance of the moving vehicle in the adjacent lane leaded to the mismatches of these two methods. However, the lane-lines were hardly affected by the vehicle, which kept the robustness of the proposed method. Moreover, the lane-lines of these frames were extracted and drawn in one image to provide more clear results, as is shown in the last image of each subfigure. We can observe that the lane-lines in the frames of SVT appear in different positions because of undesired frame jitters. Besides, the lane-lines of the frames in Fig. 10b appear in almost the same position while those in Fig. 10c and d in different positions. Consequently, the results of the proposed method according to the analysis earlier, are more close to a stable video than the other two DIS methods. This experiment demonstrates that the proposed method based on lane-line matching is resistant to the moving objects of In-Car videos.

Real case: Four real In-Car videos (RV1-RV4) for evaluation were captured at different freeway segments, as is shown in Fig. 11. Following the evaluation measure adopted by Morimoto and Chellappa (1998) and Shen *et al*. (2009), the performance of the proposed method in the real case was evaluated by using Inter-frame Transformation Fidelity (ITF) with Peak Signal-to-Noise Ratio (PSNR) which is given by:

$$\begin{cases}\mathrm{ITF}=\dfrac{1}{\mathrm{N}-1}\displaystyle\sum_{\mathrm{i=1}}^{\mathrm{N-1}}\mathrm{PSNR(i)}\\[2ex]\mathrm{PSNR(i)}=10\log_{10}\dfrac{255^2}{\mathrm{MSE(i)}}\end{cases} \qquad (24)$$

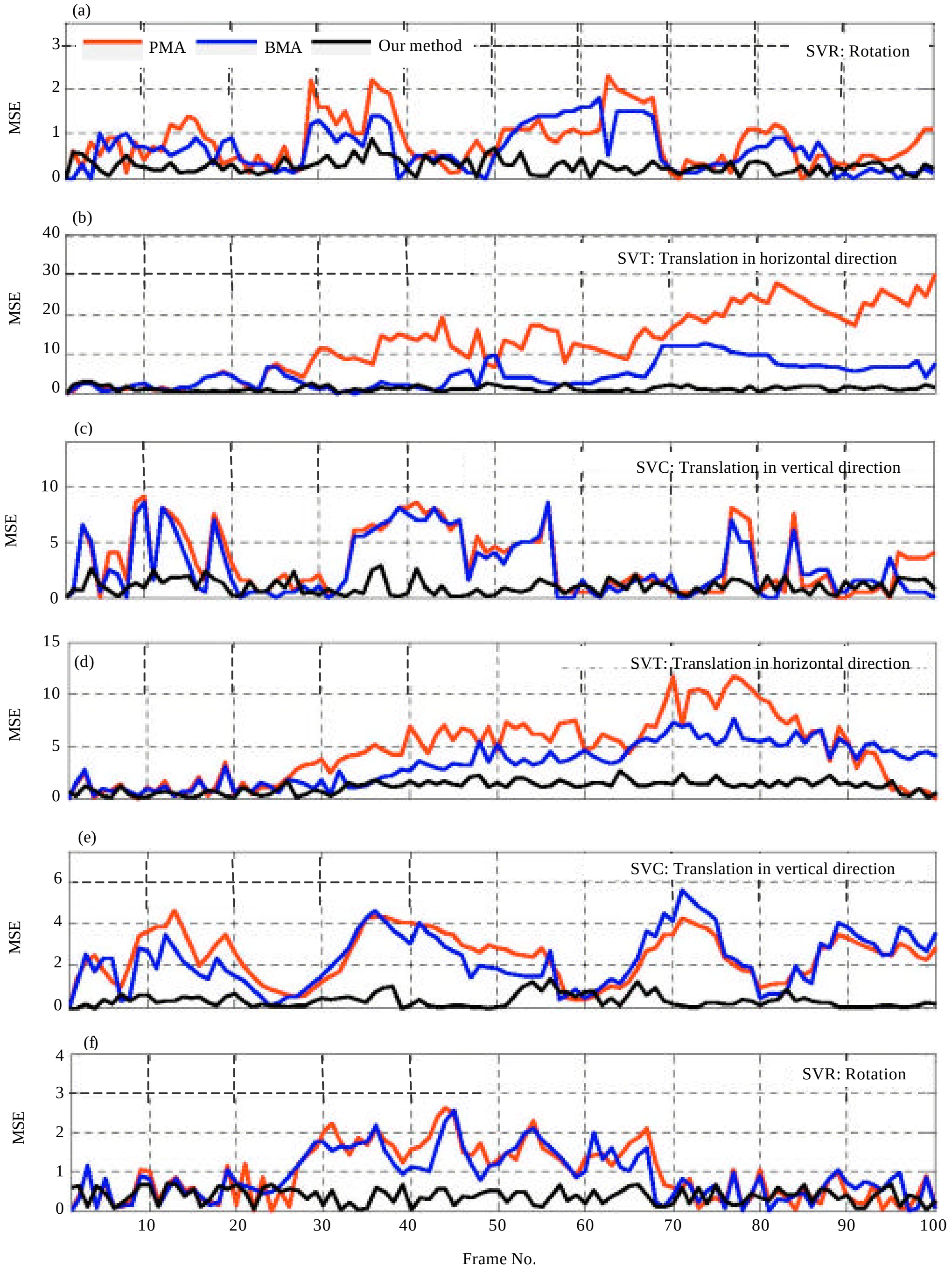

Fig. 9: The absolute differences between the real CMVs and the CMVs obtained by using different DIS methods. The results for SVR are provided in (a), the results for SVT are provided in (b) and (c) the results for SVC are provided in (d), (e) and (f)

swhere, N denotes the frame number of the input video. MSE(i) is the Mean Square Error between frame i with respect to frame i-1. The bigger ITF means the input video more close to a stable video. It should be noted that the frames of the input video should be converted into gray images before computing ITF, since the PSNR in Eq. 24 is

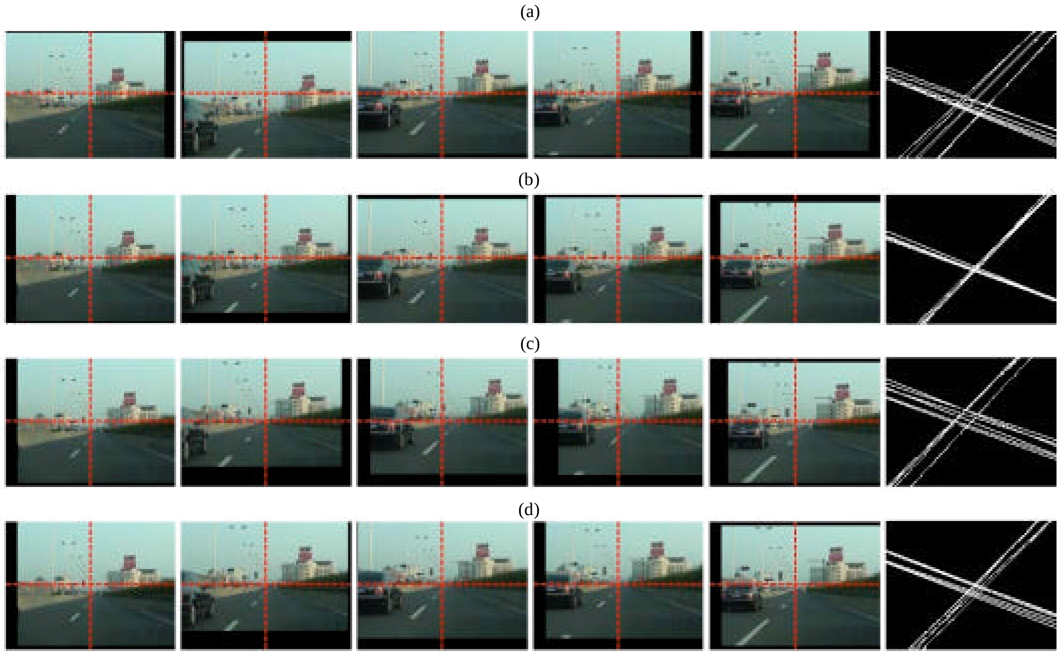

Fig. 10: The frames 28, 33, 38, 43, 48 and the lane-lines of SVT and those of the stabilized videos generated by using different DIS methods. (a) shows the frames and lane-lines of SVT and (b-d) show those of the proposed method, BMA and PCM, respectively

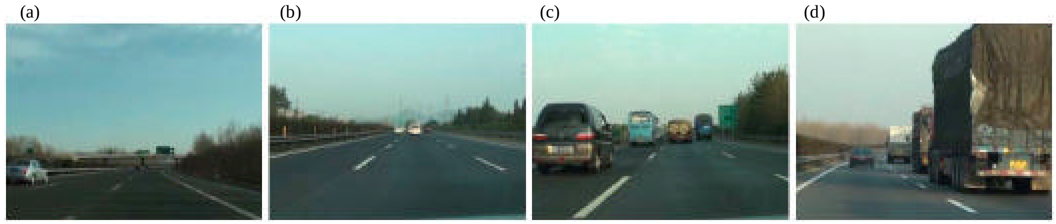

Fig. 11: The sample frames of the real In-Car videos: RV1, RV2, RV3 and RV4

Table 4: The ITFs (dB) of RV1~RV4 and the stabilized videos generated by using different DIS methods

DIS	Unstable	Our method	EMA	PMA	BMA	PCM
ITF: RV1	20.2301	25.7244	24.9832	24.7764	22.1813	22.6423
ITF: RV2	21.2511	27.9016	27.2142	26.4519	22.4728	23.1885
ITF: RV3	17.6724	24.5114	20.0135	19.9339	18.1484	18.5676
ITF: RV4	18.6967	25.4932	22.4653	22.8508	19.3462	20.5989

specific to gray images. In this experiment, the real In-Car videos were stabilized by using the proposed method, BMA, PCM, PMA and EMA, separately. Then, the ITFs of each method result as well as the real videos were computed which are provided in Table 4.

We can observe that the results of each method are bigger than RV1~RV4. In addition, the 3D DIS methods (i.e., the proposed method, PMA and EMA) have bigger ITFs than 2D methods (i.e., the BMA and PCM). This is because that the inter-frame motions of real In-Car videos generally contain not only translation but also rotation. In particular, the results of the proposed method are better than the other two 3D methods, especially for RV3 and RV4 with complex scenes or backgrounds. This is because that RV3 and RV4 were captured at the segments with heavy traffic and the moving vehicles in the adjacent lanes degraded the performances of PMA and EMA. This experimental result indicates that the proposed method can work robustly for the real In-Car videos which were captured in different traffic conditions.

Following the evaluation for the simulated videos, we presented a series of sample frames of RV4 and the stabilized ones generated by using different DIS methods. Figure 12 illustrates the frames 152, 169, 187, 205 and 216 of RV4 and the stabilized videos, in which (a) illustrates the frames of RV4 and (b-f) illustrate the

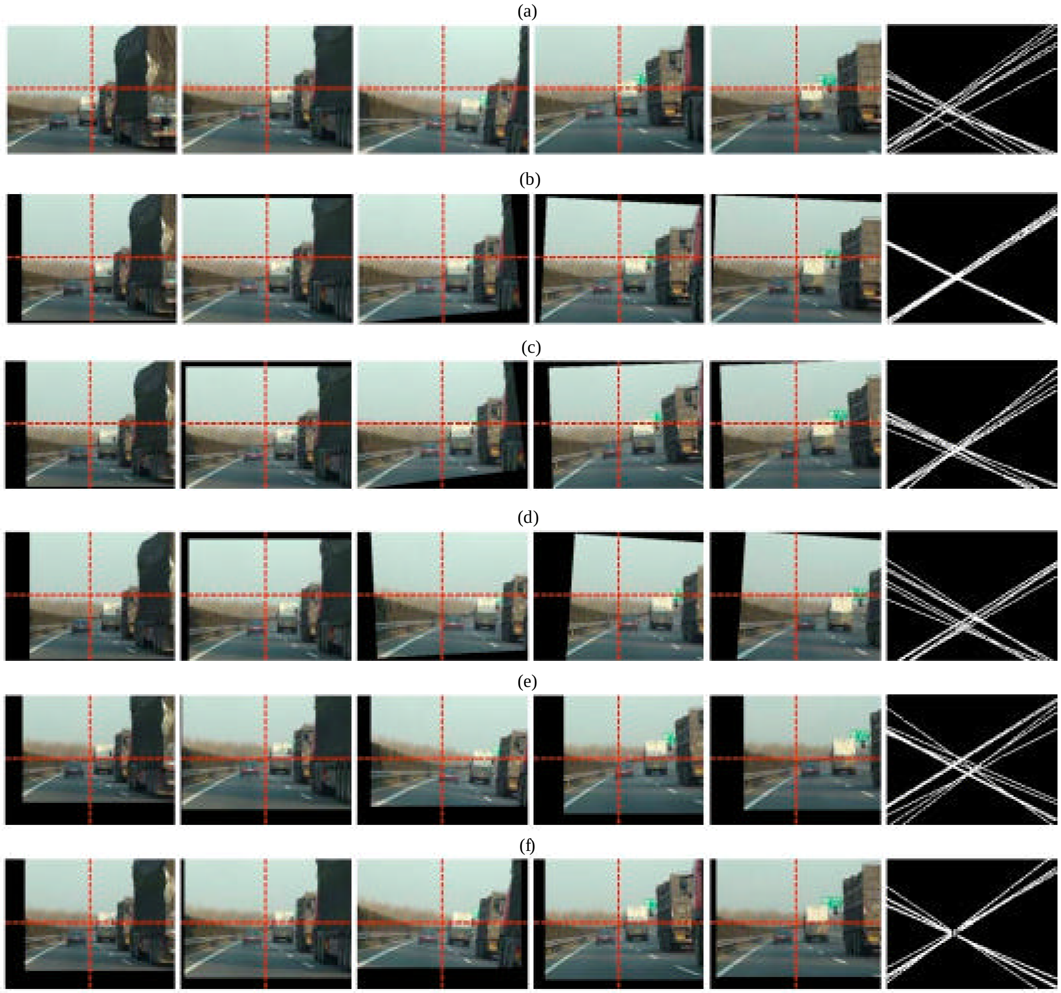

Fig. 12: The sample frames 152, 169, 187, 205, 216 and the lane-lines of RV4 and the stabilized ones. (a) shows the frames of RV4 and (b-f) show the frames of the stabilized videos generated by using the proposed method, EMA, PMA, BMA and PCM, respectively

results of the proposed method, EMA, PMA, BMA and PCM, respectively. Note that there are moving vehicles in the adjacent lanes. Nevertheless, the proposed method still can remove the effect of the undesired camera motion or frame jitter, because the lane-lines were rarely affected by these vehicles. In contrast, there are still undesired inter-frame motions in the results of the other two 3D methods (i.e., the EMA and PMA) because of the moving vehicles in the adjacent lanes. On the other hand, the 2D DIS methods (i.e., the BMA and PCM) lost the ability to remove the effect of undesired camera motion, since the real In-Car videos commonly contain undesired inter-frame rotations. Besides, the lane-lines of these frames were extracted for evaluation, as is shown in the last image of each subfigure. We can observe that the lane-lines of the frames generated by using the proposed method appear in almost the same position while those by using the other DIS methods in different positions. This result means that the results of the proposed method are more close to the real stable In-Car videos than the other DIS methods.

CONCLUSION

A digital image stabilization method for In-Car videos was presented. This method choose the lane-lines of an input In-Car video as the image features for global motion estimation and then corrected the input video to product the stable video. The experiment results showed that the proposed method can efficiently remove the effects of undesired camera motions on In-Car videos. However, we have to point out that the lane-line detection part of the proposed method would not work well for the In-Car videos without clear lane-lines, since our researches were mainly concentrated on the GME for In-Car video stabilization. Consequently, adopting a more robust lane-line detection scheme can improve the performance of the proposed method, which will be studied in our future works. In addition, it should be emphasized that we preliminarily considered the In-Car videos with straight lane-lines which are mostly encountered, especially on freeways. However, the proposed method can also deal with the In-Car video with curved lane-lines, since the area for lane-line detection is constrained in the lower part of each image where curved lane-lines approximate straight lines. Nevertheless, the lane-line model based on curved line is more coincide with a real lane-line, which can improve the accuracy of the motion estimation of the proposed method. So the DIS methods for In-Car videos with curved lane-line will be further studied in our future works.

ACKNOWLEDGMENT

This study was supported by Guangdong Natural Science Fund (GNSF). The Grant No. is: 8152840301000009.

REFERENCES

Bas, E.K. and J.D. Crisman, 1997. Easy to install camera calibration for traffic monitoring. Proceedings of the IEEE Conference on Intelligent Transportation Systems, Nov. 9-12, Boston, MA, USA., pp: 362-366.

Battiato, S., G. Gallo, G. Puglisi and S. Scellato, 2007. SIFT features tracking for video stabilization. Proceedings of the 14th International conference on Image Analysis and Processing, Sept. 10-14, Modena, Italy, pp: 825-830.

Bosco, A., A. Bruna, S. Battiato, G. Bella and G. Puglisi, 2008. Digital video stabilization through curve warping techniques. IEEE Trans. Consumer Electron., 54: 220-224.

Erturk, S., 2001a. Image sequence stabilisation based on Kalman filtering of frame positions. Electron. Lett., 37: 1217-1219.

Erturk, S., 2001b. Image sequence stabilisation: Motion Vector Integration (MVI) versus Frame Position Smoothing (FPS). Proceedings of the 2nd International Symposium on Image and Signal Processing and Analysis, June 19-21, Pula, pp: 266-271.

Erturk, S., 2003. Digital image stabilization with sub-image phase correlation based global motion estimation. IEEE Trans. Consum. Electron., 49: 1320-1325.

Forsyth, D.A. and J. Ponce, 2003. Computer Vision: A Modern Approach. Prentice Hall, New Jersey, USA., ISBN-10: 0130851, pp: 693.

Howard-Anton, C.R, 2005. Elementary Linear Algebra. 9th Edn., John Wiley and Sons, Inc., New York.

Hsu, S.C., S.F. Liang, K.W. Fan and C.T. Lin, 2007. A robust in-car digital image stabilization technique. IEEE Trans. Syst. Man Cybern. Part C Appl. Rev., 37: 234-247.

Ishak, K.A., S.A. Samad and A. Hussain, 2006. A face detection and recognition system for intelligent vehicles. Inform. Technol. J., 5: 507-515.

Ko, S.J., S.H. Lee, S.W. Jeon and E.S. Kang, 1999. Fast digital image stabilizer based on Gray-coded bit-plane matching. IEEE Trans. Consumer Electron., 45: 598-603.

Liang, C.K., Y.C. Peng, H.A. Chang, C.C. Su and H. Chen, 2004. The effect of digital image stabilization on coding performance. Proceedings of the International Symposium on Intelligent Multimedia, Video and Speech Processing, Oct. 20-22, Hong Kong, China, pp: 402-405.

Lingqiao, L., F. Zhizhong, X. Jingjing and Q. Wei, 2008. Edge mapping: A new motion estimation method for video stabilization. Proceedings of the International Symposium on Computer Science and Computational Technology, Dec. 20-22, Shanghai, China, pp: 440-444.

Marcenaro, L., G. Vernazza and C.S. Regazzoni, 2001. Image stabilization algorithms for video-surveillance applications. Proceedings of the IEEE International Conference on Image Processing, Oct. 7-10, Thessaloniki, Greece, pp: 349-352.

Morimoto, C. and R. Chellappa, 1996. Fast electronic digital image stabilization for off-road navigation. Real-time Imaging, 2: 285-296.

Morimoto, C. and R. Chellappa, 1998. Evaluation of image stabilization algorithms. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, May 12-15, Seattler, WA, USA., pp: 2789-2792.

Shen, Y., P. Guturu, T. Damarla, B.P. Buckles and K.R. Namuduri, 2009. Video stabilization using principal component analysis and scale invariant feature transform in particle filter framework. IEEE Trans. Consumer Electron., 55: 1714-1721.

Wang, C., J.H. Kim, K.Y. Byun and S.J. Ko, 2009. Robust digital image stabilization using feature tracking. Proceedings of the IEEE International Conference on Consumer Electronics, Jan. 10-14, Las Vegas, NV, United states, 1-2.

Yang, S.H., F.M. Jheng and Y.C. Cheng, 2007. Two-dimensional adaptive image stabilisation. Electron. Lett., 43: 446-447.

Yang, J., D. Schonfeld and M. Mohamed, 2009. Robust video stabilization based on particle filter tracking of projected camera motion. IEEE Trans. Circ. Syst. Video Technol., 19: 945-954.

Yanqing, W., C. Deyun, S. Chaoxia and W. Peidong, 2010. Vision-based road detection by monte carlo method. Inform. Technol. J., 9: 481-487.

Zhang, Y., M. Xie and D. Tang, 2010. A central sub-image based global motion estimation method for in-car video stabilization. Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining, Jan. 9-10, Phuket, Thailand, pp: 204-207.