A Video Stabilization Algorithm Based on Affine SIFT

Ming Fang, Haoyue Li
School of computer science and technology
Changchun University of Science and Technology
Changchun, China
fangming@cust.edu.cn; lhy102526@163.com

Shuzhe Si
Test and Automation Center
GoerTek Limited by Share Ltd
Weifang, China
reg.si@goertek.com

Abstract—Electronic image stabilization (EIS) is an important video enhancement technology which aims at removing annoying random jitter from videos. At present, the commonly EIS algorithm is to filter the feature trajectory, such as Kalman filter and Median filter. However, the filtered trajectories of these methods are greatly deviated from the original trajectory, and the images often lose large information after being stabilized. This paper uses affine SIFT (ASIFT) feature matching method to get the best estimating the affine matrix, then Gaussian low-pass filtering of the original path can compensate for the motion of the smoothed path, then the jitter frame is stabilized. Compared with Kalman filter, the experiments show that Gaussian filter better retains the camera active motion, stabilized video sequences lost less pixel information.

Keywords—electronic image stabilization; Kalman filter; Gaussian filter; ASIFT

I. INTRODUCTION

Electronic image stabilization (EIS) is a technique to obtain a stable video by determining the inter-frame offset through the inter-frame mapping relationship, and then motion smoothing to remove random jitter in the video sequence. EIS technology mainly includes motion estimation and motion correction. Motion estimation calculates the global motion vector between adjacent frames to find the global motion trajectory. Motion correction is to smooth the estimated motion trajectory, remove the high frequency of random jitter, and obtain stable video sequences by motion compensation [1].

According to different motion estimation models, EIS can be classified into three methods: 2D, 2.5D and 3D. The 3D method needs to use SFM algorithm to reconstruct the scene and restore the camera pose [2]. However, in the cases of feature tracking failure, motion blur, camera scaling and fast moving, the robustness of 3D reconstruction will be reduced. The 2.5D also requires some 3D information [3]. The 2D method can estimate the affine transformation matrix between adjacent frames, and the camera path can be represented by its linear multiplication. In general, the 2D method is faster and more robust because it only estimates a linear transformation model between adjacent frames [4]. In this paper, an affine transformation model with 6 degrees of freedom is used in 2D.

II. GLOBAL MOTION ESTIMATION

A. ASIFT Feature Matching

For the random jitter video captured, the camera's optical axis direction changes resulting in image distortion. Although the SIFT algorithm has complete scale invariance, it does not have complete affine invariance. Therefore, there is a limitation in the image feature extraction which has a great change in shooting angle. ASIFT algorithm can simulate the characteristics of the simulated image by selecting the sampling parameters to simulate the images of different longitude and latitude. Then, the feature matching is performed by combining the features of all the simulated images [6]. In Fig. 1, we compare the SIFT algorithm with the matching results of any two frames of ASIFT algorithm in the jitter video, where SIFT has 214 pairs of matching points and ASIFT has 2930 pairs of matching points.

Fig. 1. Comparison of SIFT and ASIFT results:(a)SIFT matching (b)ASIFT matching

Due to the inaccurate motion estimation of the current EIS algorithm and the large deviation between the compensated motion path and the original path, there is a large area loss of pixels after the image stabilization. To solve this problem, we use affine SIFT (ASIFT) feature matching algorithm to accurately estimate the inter-frame motion vector, and then use Gaussian filter to smooth the motion [5]. The filtered path is consistent with the original path, which preserves the camera's active motion well and reduces the black edge phenomenon caused by the loss of pixels in the image after stabilization.
B. Motion Estimation Model

The geometric transformation between the adjacent two frames \(I(p)\) and \(I'(p')\) can be described by 2D affine transformation matrix, then the relationship between the pixel homogeneous coordinates \(p = (x, y, 1)\) and \(p' = (x', y', 1)^T\) of the matching points in the overlap region of the two image can be written as \(p' = Fp\). If ASIFT is used to match, there are \(n\) pairs of two-dimensional matching points \(src\), \(dst\), then the optimal affine change matrix between the two frames can be obtained by solving the \([A | b]\) when sum is the minimum by (1).

\[
F = \begin{bmatrix} a_1 & a_2 & t_x \\ a_3 & a_4 & t_y \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} A \\ b \end{bmatrix}
\]

\[
[A' | b'] = \arg \min_{(A,b)} \sum_{i=1}^{n} \left[ dst[i] - A \times src[i]^T - b \right]^2
\]

The affine transformation matrix corresponding to the pixel coordinates from \(I_m\) frames to \(I_n\) frames in jitter video can be denoted as \(F_1^m = F_1^{m+1} \ldots F_1^n\), the original motion path is \(C(t) = F_1^m F_1^{m+2} \ldots F_1^n\), \(m < n\).

III. MOTION SMOOTHING AND COMPENSATION

The original path \(C(t)\) is smoothed using Gaussian low pass filter proposed by Matsushita. The smoothed path is \(\tilde{C}(t) = S_i C(t)\), where \(S_i\) is the transformation matrix from the original jitter frame \(I_i\) to the stable frame \(I'_i\), and it can be solved by neighborhood transform matrix. The neighborhood radius is defined as \(k\). The definition of the neighborhood range corresponding to the continuous sequence at the time of \(t\) is shown in Fig. 2, and the dotted line area represents the window size.

As shown in Fig. 2, \(k\) frames are available before and after the time of \(t\). For video sequences of length \(L\), special treatment is required when \(k\) frames are not available before or after the time of \(t\). Define a matrix \(M\) of size \(1 \times (2k+1)\), for different times of \(t\), as shown in TABLE I, \(M\) can be divided into three cases, for (a) when \(t \leq k\), because the time before \(t\) is less than \(k\) frame, the former \(i-1\) column of \(M\) stores the unit matrix \(3 \times 3\) of \(I\); (b) when \(t \geq L-k+1\), because the time after \(t\) is less than \(k\) frame, then after the \(j\) column of the \(M\) matrix stores \(I\).

Define \(M(i, j) = [M(i, (i, j), M(i, (i, j), \ldots, M(i, (i, j))]\), the discrete signal length is \(2k+1\), where \(i, j \in [1, 3]\). Define the Gaussian kernel function:

\[
G_k(n) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(n-(2k+1))^2}{2\sigma^2}}
\]

where \(n = 1, 2, \ldots, 4k+1\), \(\sigma = \sqrt{k}\), then \(S_i\) can be solved by (3):

\[
S_i(i, j) = \sum(M(i, j) \odot G_k(n))/\sum(M(3,3) \odot G_k(n))
\]

Where \(\odot\) is the convolution operation, after \(S_0, \ldots, S_6\) is calculated, the original video frame is transformed to perform the image stabilization compensation. The new camera path smoothing degree is determined by \(k\), the greater the value of \(k\), the better the smoothing effect is. In this paper, we set \(k = 50\), and compare with Kalman filter proposed by Andrey Litvin [7]. Fig. 3 shows the effect of Gaussian filter and Kalman filter which respectively smooth the affine transformation matrix of 6 degrees of freedom.

![Fig. 2. Sequential frames in the window \(w = 2k+1\) at the time of \(t\)](image)

<table>
<thead>
<tr>
<th>(M_{(a)})</th>
<th>(M_{(b)})</th>
<th>(\cdots)</th>
<th>(M_{(c)})</th>
<th>(\cdots)</th>
<th>(M_{(d)})</th>
<th>(\cdots)</th>
<th>(M_{(e)})</th>
</tr>
</thead>
<tbody>
<tr>
<td>(I)</td>
<td>(I)</td>
<td>(F_1^i)</td>
<td>(\cdots)</td>
<td>(I)</td>
<td>(\cdots)</td>
<td>(F_1^{(i+1)+1})</td>
<td>(\cdots)</td>
</tr>
<tr>
<td>(F_1^{(i-1)+1})</td>
<td>(\cdots)</td>
<td>(F_1^{(i+1)+1})</td>
<td>(\cdots)</td>
<td>(I)</td>
<td>(\cdots)</td>
<td>(F_1^{(i+1)+1})</td>
<td>(\cdots)</td>
</tr>
<tr>
<td>(F_1^{(i-1)+1})</td>
<td>(\cdots)</td>
<td>(F_1^{(i+1)+1})</td>
<td>(\cdots)</td>
<td>(I)</td>
<td>(\cdots)</td>
<td>(\cdots)</td>
<td>(I)</td>
</tr>
</tbody>
</table>
Fig. 3. Comparison of Kalman filter and Gaussian filter

(a) 

(b) 

(c) 

Fig. 4. Comparison of video stabilization by Kalman filter and Gaussian filter: (a) Original video frame of 145th, 146th, 226th, 227th. (b) Kalman smoothing video frame of 145th, 146th, 226th, 227th. (c) Gaussian smoothing video frame of 145th, 146th, 226th, 227th
According to Fig.3, the path of Kalman filter proposed by Litvin has a large deviation from the original path. Motion compensation for the filtered traces usually results in the loss of most of the information and a large black edge phenomenon. In this paper, we use Gaussian filter proposed by Matsushita to filter the motion path in window. The filtered path is very close to the original path and there is less loss of pixel information in the stable frame. As can be seen from the x-direction trajectory in Fig. 3, there is a leftward scanning motion on the 300th frame of the jitter video, and the Gaussian filter preserves the active motion better. However, the Kalman filter path and the original path have a large deviation at about 200th frame of the video, and the pixel information of the compensated image is seriously lost.

IV. EXPERIMENTS

In this paper, the experimental environment is a dual core 3.10GHz processor, using third party kit mexopenCV to call the openCV function by MATLAB. Fig. 4 shows the estimated motion vector by ASIFT, respectively used Kalman and Gaussian filter for image stabilization contrast.

In Fig. 4, the second line uses the Litvin’s Kalman filter to stabilize the image. There was a serious loss of information and did not keep the active motion very well. In this paper, we use Matsushita’s Gaussian filter method. Considering that when the window length $W < 2k + 1$, to ensure the convolution length is kept at $2k + 1$, we add unit matrix $I$ to matrix $M$. Stabilization effect as shown in the third line of Fig 4, by observing the changes of the window position, it can be seen that the up and down jitter has been suppressed.

V. CONCLUSION

In our lives, hand-held filming equipment due to walking or hand-shaking led to the video caused the viewer to see dizziness, the experiment will be effective in suppressing jitter, but also to avoid the maximum loss of information within each frame. By using ASIFT matching method, the optimal affine transformation matrix can be estimated and camera path $C(t)$ can be calculated. The $C(t)$ is smoothed by Gaussian, the transformation matrix in the window and the defined Gaussian kernel function are convoluted, so we can calculate the compensation matrix $S_t$, then the image stabilization frame can be obtained by $I'_t = S_t I_t$. We also compared with the Kalman method proposed by Litvin, the experimental results show that the Gaussian filter path is better to retain the active motion of the camera.

REFERENCES