Camera Motion Quantification and Alignment

Xiaochun Cao
University of Central Florida
Computational Imaging Lab
xccao@cs.ucf.edu

Jiangjian Xiao
Sarnoff Corporation
Princeton, NJ 08540, USA
jxiao@sarnoff.com

Hassan Foroosh
University of Central Florida
Computational Imaging Lab
foroosh@cs.ucf.edu

Abstract

*We propose a method to synchronize video sequences of distinct scenes captured by cameras undergoing similar motions. For the general camera motion and 3D scene, the camera ego-motions are featured by **fundamental ratios** obtained from the fundamental matrices. In the case of pure translation, translational magnitude features is used. These extracted features are invariant to the camera internal parameters, and therefore can be computed without recovering camera trajectories along the image sequences. Consequently, the alignment problem reduces to matching sets of feature vectors, obtained without any knowledge of other sequences. Experimental results demonstrate the accuracy and applications of the proposed method.*

1 Introduction

The problem of synchronizing video sequences has become an active area since Stein's first method [11]. Recent methods [2, 7, 13, 14, 15] tackle the problem of automatic video synchronization for independently moving cameras and overlapping dynamic scenes. The alignment of non-overlapping sequences was first addressed in [3] based on the assumption that the two sequences are captured by a stereo rig. Wolf and Zomet [16] proposed a method to compute the fundamental matrix for synchronizing the sequences. Rao et. al. [10] extended [4] to synchronize non-overlapping but same events, such as human activities.

The proposed approach herein tackles a more general problem, i.e. the synchronization of video sequences of distinct general scenes captured by cameras undergoing similar motions. Two camera motions are similar when the camera locations and poses in all corresponding time slots are related by a common 3D similarity transformation. In the case of general camera motion, similar camera motions result in the same essential matrices, between synchronized frame pairs. Our method is based on the key observation that the equality of the essential matrices is reflected in the uncalibrated fundamental matrices' upper-left 2×2 elements, which are used as features for synchronization.

For certain special motions, the number of degrees of freedom of the fundamental matrix is less than the seven of general motion, or the epipolar geometry is undefined. Consequently, the fundamental ratios would fail to provide camera ego-motion information. In order to overcome the degeneracies, we analyze the special motions and provide dedicated features to align them. For cameras undergoing pure rotation or zooming, we proposed solutions are solved in [17] based on using rotation angles or zooming factors. In this work, we consider the special case of pure translation and use the relative translational magnitude by slicing the 3D data volume along the epipolar lines.

The main advantage of the new method is that it can handle video sequences of general distinct scenes captured by arbitrarily moving cameras. Different from the previous efforts on temporal alignment of non-overlapping sequences which typically utilize the inter-sequence relationship and hence inherently involve two sequences, the proposed approach is a 1-sequence process and computes the camera ego-motion for each sequence separately. Besides its efficiency allowing a combination-free implementation, our algorithm, as a 2D solution, does not involve scene reconstruction or 3D recovery of the camera trajectories.

2 General Camera Motion

A video sequence $\mathcal{V}$ is an ordered set of images $\{\mathcal{I}_j\}_1^N$ captured by a camera. Assuming a pin-hole camera model and constant camera internal parameters, the camera projection matrix $\mathbf{P}_j$ of the frame $\mathcal{I}_j$ can be factorized as $\mathbf{P}_j = \mathbf{K}[\mathbf{R}_j|\mathbf{t}_j]$, where $\mathbf{K}$ is the camera calibration matrix including the intrinsic parameters, and $\mathbf{R}_j$ and $\mathbf{t}_j$ are camera rotation and translation, respectively. If we have another sequence $\mathcal{V}'$, i.e. $\{\mathcal{I}'_{j'}\}_1^{N'}$, of another distinct scene captured by a camera undergoing similar motion as that of $\mathcal{V}$, the camera projection matrix $\mathbf{P}'_{j'}$ of frame j' in $\mathcal{V}'$ can be factorized as $\mathbf{P}'_{j'} = \mathbf{K}'[\mathbf{R}_j \mid \mathbf{t}_j]\mathbf{H}_s$, where $j' = c(j)$ is the frame index in $\mathcal{I}'$ corresponding to frame j in sequence

The 18th International Conference on Pattern Recognition (ICPR'06)
0-7695-2521-0/06 $20.00 © 2006 IEEE

$\mathcal{I}$, and the 4×4 similarity transformation matrix $\mathbf{H}_s$ relates the locations and poses of the two synchronized cameras. The correspondence relationship, $c(\cdot)$, can be dynamic [10] or modeled by a 1D affine model [2, 3, 14].

In the noise-free case, it is simple to verify that the relative translation $\mathbf{t}_{i,j}$ (orientation $\mathbf{R}_{i,j}$) between frames i and j in sequence $\mathcal{V}$ are equal up to an unknown scale to the relative translation (but the same orientation) between frames $c(i)$ and $c(j)$ in sequence $\mathcal{V}'$. Therefore, we have

$$\mathbf{E}_{i,j} \sim [\mathbf{t}_{i,j}]_\times \mathbf{R}_{i,j} \sim \mathbf{E}'_{c(i),c(j)}, \tag{1}$$

where $\sim$ indicates equality up to multiplication by a non-zero scale factor, and $[\cdot]_\times$ is the notation for the skew symmetric matrix characterizing the cross product [6]. As a result, when the camera calibration matrices $\mathbf{K}$ and $\mathbf{K}'$ of the sequences $\mathcal{V}$ and $\mathcal{V}'$ are identical, the corresponding uncalibrated fundamental matrices $\mathbf{F}_{i,j}$ and $\mathbf{F}'_{c(i),c(j)}$ should be equal, which can be used for video synchronization. However, we are interested in a more general case, where $\mathbf{K}$ and $\mathbf{K}'$ are constant but different between the sequences.

In the case of a simplified camera model, i.e. unit aspect ratio and zero skew, it is easy to verify that the upper left 2×2 sub-matrix of $\mathbf{F}$ has the form:

$$\mathbf{F}^{2\times 2} \sim \begin{bmatrix} \epsilon_{1st}\mathbf{t}^s_{i,j}\mathbf{r}^t_1 & \epsilon_{1st}\mathbf{t}^s_{i,j}\mathbf{r}^t_2 \\ \epsilon_{2st}\mathbf{t}^s_{i,j}\mathbf{r}^t_1 & \epsilon_{2st}\mathbf{t}^s_{i,j}\mathbf{r}^t_2 \end{bmatrix}, \tag{2}$$

where ϵ_{rst} for $r, s, t = 1, \ldots, 3$ is the permutation tensor ([5], p. 172), $\mathbf{r}_i$ are columns of the rotation matrix $\mathbf{R}_{i,j}$. The interesting observation of $\mathbf{F}^{2\times 2}$ is that the ratios among its elements F_{ij} are invariant to the camera internal parameters and reflect only the camera ego-motion. In this paper, we call these ratios the *fundamental ratios*. Therefore, we are able to extract an independent four-dimensional feature vector, $\mathbf{v}_g$, for general camera ego-motion as,

$$\mathbf{v}_g = \text{sign}(F_{11})[F_{11}, F_{12}, F_{21}, F_{22}]/\|\mathbf{F}^{2\times 2}\|_F, \tag{3}$$

where $\|\cdot\|_F$ is the Frobenius norm.

It is unlikely for the 4D vector $\mathbf{v}_g$ to uniquely characterize the relative camera ego-motion, which has five degrees of freedom: both the rotation $\mathbf{R}_{i,j}$ and translation $\mathbf{t}_{i,j}$ have three degrees of freedom, but there is an overall scale ambiguity. In addition, there are four possible setups of relative camera position and orientation for the same essential matrix as shown in ([6], p. 258). However, similar camera ego-motions would result in the same $\mathbf{v}_g$, which can be used to synchronize video sequences as shown in Section 2.1.

2.1 Implementation Details

The initial frame-to-frame correspondences are accomplished using the SIFT feature proposed by Lowe [9]. The fundamental matrix for general camera motion between the two views can be computed using the MAPSAC [12]. The epipole of the pure translational shot is computed using the eigenvalue decomposition of the matrix stacked by all lines connecting the corresponding points.

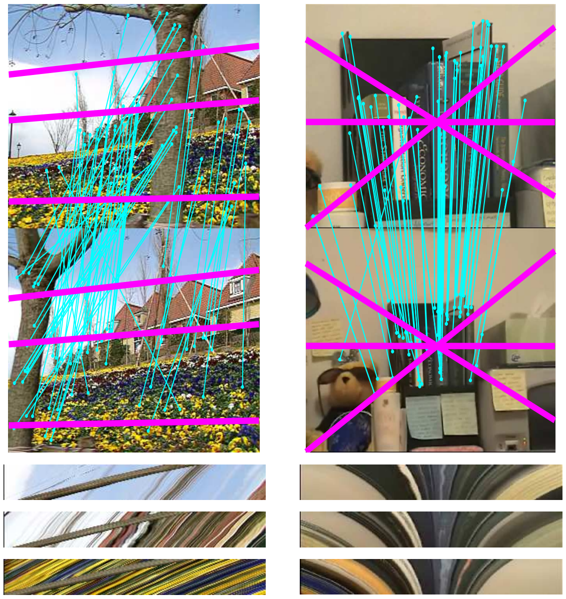
The above scheme assumes that there are nontrivial overlapping areas between the two frames of one video sequence. However, in the case of dynamic timeline model, which can be solved using dynamic programming [1], two frames i and l might be far away and hence have no correspondences for the computation of the fundamental matrices. In this case, the viewing graph theory [8] can be used for computing the fundamental matrices between the frames i and l, when there are two frames j and k which have overlapping areas with both i and l, i.e., the fundamental matrices inside two tri-views (i, j, k) and (j, k, l) are available.

In addition, the camera ego-motion, i.e. the relative translation and rotation, between the two views should be reasonably big in order to overcome noises. Finally, in order to increase the robustness of the proposed method, we use the coarse-to-fine framework since the synchronization in the coarser levels captures global features, and therefore an error in computation of frame correspondences will not be propagated to the rest of the warping path.

3 Pure Translation Camera Motion

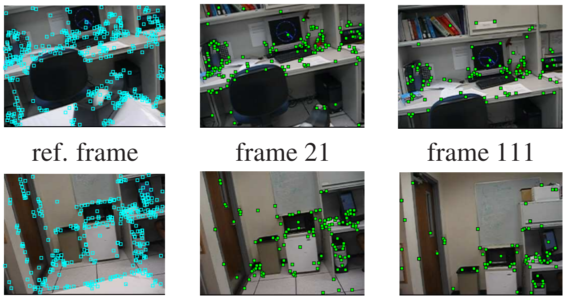
For pure translational camera motion, the fundemental matrix $\mathbf{F}$ between any two frames is invariant and degenerates to $[\mathbf{e}]_\times$, where $\mathbf{e}$ is the common epipole for all frames. It is evident that $\mathbf{F}$ has only two degrees of freedom. Therefore, it is impossible to use the fundamental ratios $\mathbf{v}_g$ in Eq. (3) to align video sequences in this case. In order to quantify the camera ego-motion in the pure translational cases, we slice the three dimensional data volume along the epipolar lines as shown in Fig. 1 (a). Notice that the y-axis is the temporal direction. We observe that the trajectories of the same 3D points are represented by a two dimensional curve in the 2D slice images, and the first order derivative of the trajectory characterizes the relative translational speed. We define the relative translational magnitude, i.e. the relative distance between the first frame and the current frame along the x-axis of the slice shown in Fig. 1 (bottom), as the pure translational camera motion feature, $\mathbf{v}_t$.

One interesting observation is that, since the effects of forward/outward translation and zooming are similar, the pure translational feature $\mathbf{v}_t$ can also be used to synchronize zooming sequences although it is not really a camera motion. For example, the two frames from a zoom out sequences, shown in Fig. 1 (b), can be treated as an outward tracking along a direction parallel to the principal axis. However, the translational moving speed (the zooming factor) is not constant, and therefore the trajectories of the same 3D points are not as straight as those in Fig. 1 (a).



(a) A translational shot (b) A zooming shot

Figure 1. The corresponding points (computed by [9]) are connected by cyan lines, while the epipolar lines in magenta. The bottom slices are cut along the epipolar lines.



ref. frame frame 21 frame 111

ref. frame frame 1 frame 70

Figure 2. (Top) Sample frames from the 1^{st} video with extracted features superimposed, and the reference images (left). (Bottom) Identical elements for the 2^{nd} video.

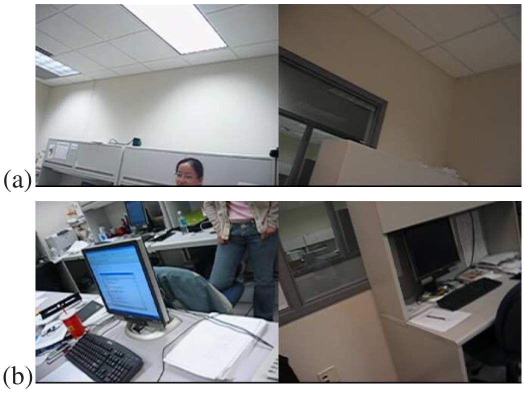


Figure 3. Two appended frames after synchronization.

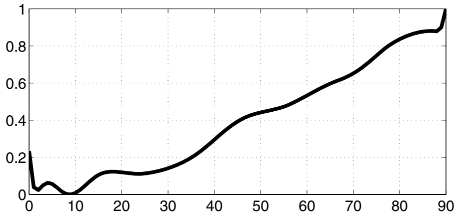


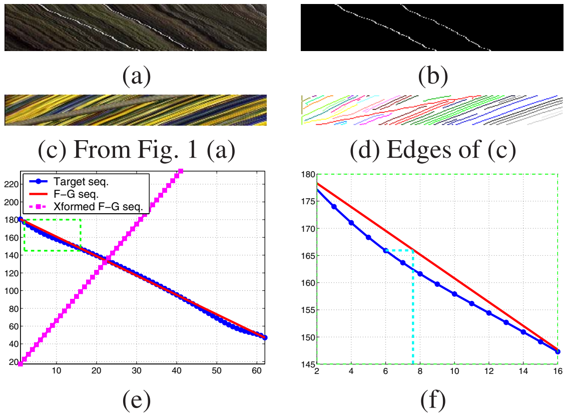
Figure 4. Average distance (rescaled between $[0, 1]$) of the camera general ego-motion features computed for different time delay.

4 Experimental Results

As the first example, we took two sequences shown in Fig. 2. The trajectories of the two cameras were controlled by a CRS Plus robot to make sure they go through the similar motion. For this pair of sequences, we have ground truth information of the temporal dilation ($\alpha = 2.0$) by setting the ratios of the speeds of the robot arm motions as $2 : 1$ for the two sequences. In this example, we manually match one pair of frames between two videos as shown in Fig. 2 left. The extracted SIFT feature points of the starting reference frame are superimposed on the reference images as cyan square markers, and the matched correspondences are shown in blue square markers in each frame. The estimated affine timeline model is $c(j) = 2.0014 * j + 18.1617$, whose temporal dilation parameter, $\alpha = 2.0014$, accurately matches the ground truth information.

In the second example, we take advantage of the available knowledge of $\alpha = 2.0$. The input sequences are non-overlapping and have different lighting conditions as shown in Fig. 3. The two sequences are challenging in that some frames are very textureless, and that there are nontrivial moving chairs and talking person in the sequence in Fig. 3. Fig. 4 shows the average distance between the camera ego-motion features $\mathbf{v}_g$ as a function of the time delay. The graph goes to zero at 9, i.e. the time delay between the two video sequences. Corresponding frames based on this time delay and known time dilation 2 are shown in Fig. 3.

The 18th International Conference on Pattern Recognition (ICPR'06)
0-7695-2521-0/06 $20.00 © 2006 IEEE



(a)

(b)

(c) From Fig. 1 (a)

(d) Edges of (c)

(e)

(f)

Figure 5. (a) The slice cut from the 3D volume along the cyan epipolar line in Fig. 6. (b) The two extracted curves of the foreground street lamps as marked by red arrows in Fig. 6. (e) The computed translational speed of the two sequences. (f) A close view of (e).

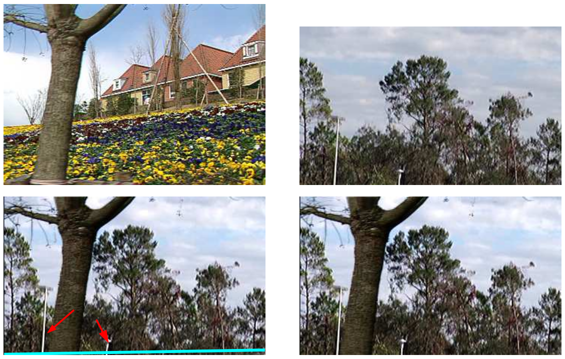


Figure 6. The top are two frames from the source and target sequence, respectively. The bottom are two consecutive target frames superimposed with the foreground tree layers extracted from the corresponding source frames after synchronization.

To demonstrate our algorithm in the pure translational motion. We use the standard flower garden (F-G) sequence and the sequence (called the target sequence hereafter) from [18] as shown in Fig. 6. Due to non-constant moving speed of the camera and the shaking, the slice cut from the 3D volume of the target sequence is not a straight line anymore. To synchronize the two sequences, we used the left street lamp (the left curve in Fig. 5 (b)) and the tree branch (the curve with smallest slope in Fig. 5 (d)) to compute the relative translational magnitudes, which are illustrated in Fig. 5 (e). After inverse and scale the flower-garden curve, we have the red and blue curves. Evidently, we cannot have a linear correspondence relationship between the sequences. For example, in the close view in Fig. 5 (f), the frame 6 of the target sequence should match frame 8 in the flower-garden sequence. After applying the dynamic programming, we have temporal alignment between the two sequences. In order to verify our results, we composited the layers of the foreground tree branch computed using the method proposed in [18] into the target background. Some of the frames are demonstrated in Fig. 6.

References

[1] A. Bryson. *Dynamic Optimization*. Addison Wesley, 1999.

[2] R. Carceroni, F. Padua, G. Santos, and K. Kutulakos. Linear sequence-to-sequence alignment. In *Proc. IEEE CVPR*, pages 746–753, 2004.

[3] Y. Caspi and M. Irani. Alignment of non-overlapping sequences. In *Proc. IEEE ICCV*, pages 76–83, 2001.

[4] Y. Caspi, D. Simakov, and M. Irani. Feature-based sequence-to-sequence matching. In *Proc. VAMODS workshop with ECCV*, 2002.

[5] H. Goldstein. *Classical Mechanics*. Addison-Wesley, Reading, MA, 2nd edition, 1980.

[6] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004.

[7] I. Laptev, S. Belongie, P. Perez, and J. Wills. Periodic motion detection and segmentation via approximate sequence alignment. In *Proc. IEEE ICCV*, pages 816–823, 2005.

[8] N. Levi and M. Werman. The viewing graph. In *Proc. IEEE CVPR*, pages 518–524, 2003.

[9] D. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, 2004.

[10] C. Rao, A. Gritai, M. Shah, and T. Syeda-Mahmood. View-invariant alignment and matching of video sequences. In *Proc. IEEE ICCV*, pages 939–945, 2003.

[11] G. Stein. Tracking from multiple view points: Self-calibration of space and time. In *DARPA IU Workshop*, pages 521–527, 1998.

[12] P. H. S. Torr. Bayesian model estimation and selection for epipolar geometry and generic manifold fitting. *Int. J. Comput. Vision*, 50(1):35–61, 2002.

[13] P. Tresadern and I. Reid. Synchronizing image sequences of non-rigid objects. In *Proc. BMVC*, pages 629–638, 2003.

[14] T. Tuytelaars and L. V. Gool. Synchronizing video sequences. In *Proc. IEEE CVPR*, pages 762–768, 2004.

[15] A. Whitehead, R. Laganiere, and P. Bose. Temporal synchronization of video sequences in theory and in practice. In *Proc. IEEE WACV/MOTION*, pages 132–137, 2005.

[16] L. Wolf and A. Zomet. Sequence-to-sequence self calibration. In *Proc. ECCV*, pages 370–382, 2002.

[17] J. Xiao, X. Cao, and H. Foroosh. A new framework for video cut and paste. In *Proc. MMM*, 2006.

[18] J. Xiao and M.Shah. Accurate motion layer segmentation and matting. In *Proc. IEEE CVPR*, pages 698–703, 2005.