A VOTING SCHEME FOR ESTIMATING THE SYNCHRONY OF MOVING-CAMERA VIDEOS

D.W. Pooley, M.J. Brooks, A.J. van den Hengel, W. Chojnacki

School of Computer Science, University of Adelaide, SA 5005, Australia
CRC for Sensor Signal & Information Processing, Mawson Lakes, SA 5095, Australia
{dwpooley,mjb,anton,wojtek}@cs.adelaide.edu.au

ABSTRACT

Recovery of dynamic scene properties from multiple videos usually requires the manipulation of *synchronous* (simultaneously captured) frames. This paper is concerned with the automated determination of this synchrony when the temporal alignment of sequences is unknown. A cost function characterising departure from synchrony is first evolved for the case in which two videos are generated by cameras that may be moving. A novel voting method is then presented for minimising the cost function in the case where the ratio of the cameras' frame rates is unknown. Experimental results indicate this relatively general approach holds promise.

1. INTRODUCTION

An important problem in computer vision is the recovery of scene properties from multiple videos. For a dynamic scene, this generally requires the analysis of frames captured simultaneously within different videos. (For example, the 3D-analysis of the 1966 Soccer World Cup video films in [1] involved manual estimation of such synchrony.) This paper is concerned with the automated determination of synchronisation information.

A pair of videos can be considered synchronised if their frames have been timestamped by a common clock. Once such timestamps are known, synchronous frames can be estimated via frame interpolation. We assume that each video sequence has a constant frame rate. Two overlapping videos can then be related by the simple linear constraint

$$f_2 = a + bf_1, \qquad (1)$$

where f_1 and f_2 are frame indices from videos 1 and 2, respectively; b is the ratio of the frame rates of the two cameras; and a is the frame offset between the two videos, specifying which frame the second camera captured as the first camera started recording. Observe that a may be positive, negative or zero, and that b must be positive.

The synchronisation problem has previously been examined under particular camera or scene constraints. It is assumed in [2] that a scene is viewed by stationary cameras, and a coarse-to-fine method is presented for estimation of synchronisation parameters and a homography that best relate the two sequences. Under the same constraints, the centroids of tracked moving objects are found in [3, 4]. A repeated random sampling scheme similar to the least median of squares method is used to estimate a frame offset and a homography that are most consistent with a possible set of matches. A similar approach is used more recently in [5] to estimate either a homography or a fundamental matrix. Complexity is reduced by considering trajectories as a whole rather than matching individual points, and sub-frame synchronisation is achieved by interpolating image points. For stationary orthographic cameras, videos are synchronised in [6] using a measure of the error in a special rank constraint. In [7] moving cameras are rigidly held together. An offset is sought such that synchronous frame pairs have transformations to the successive frames in their sequences that best reflect the rigidity of the camera pair.

In this paper, we consider a relatively general form of the problem. In particular, we examine how two videos may be synchronised that arise from moving cameras viewing arbitrarily complex dynamic objects travelling across a stable background. The frame rate ratio and the offset are assumed unknown.

2. COST FUNCTION FOR MOVING CAMERAS

Suppose that two freely moving cameras view a scene comprising static background and an arbitrarily complex dynamic foreground. Assume the resulting sequences have significant overlap, possibly different frame rates, and that scene points associated with the dynamic parts of the scene are tracked through a substantial portion of the videos.

In [3–5], moving scene points alone are used to recover both synchronisation parameters and a homography or fundamental matrix. For moving cameras, such points do not constrain the spatial transformations relating frames from

the same video. To recover a complete spatial and temporal description of the cameras, static scene points must also be used. It is therefore assumed here that the static background can be used to compute any fundamental matrix $\mathbf{F}_{i,j}$ associated with frames i and j of videos 1 and 2, respectively.

The projections of a moving scene point in a pair of synchronous frames will be related by the epipolar geometry encoded in the relevant fundamental matrix. This provides a means of measuring the "departure from synchronisation".

Suppose that videos 1 and 2 have n_1 and n_2 frames, respectively. Let the hth moving scene point project to locations $\mathbf{p}_{h,0}, \mathbf{p}_{h,1}, \ldots, \mathbf{p}_{h,n_1-1}$ in the frames of video 1 and $\mathbf{p}'_{h,0}, \mathbf{p}'_{h,1}, \ldots, \mathbf{p}'_{h,n_2-1}$ in the frames of video 2. The epipolar lines associated with points $\mathbf{p}_{h,i}$ and $\mathbf{p}'_{h,j}$ are then given by

$$\mathbf{l}'_{h,i,j} = \mathbf{F}_{i,j}\mathbf{p}_{h,i} \quad \text{and} \quad \mathbf{l}_{h,j,i} = \mathbf{F}_{i,j}^\top \mathbf{p}'_{h,j}. \qquad (2)$$

For given values of offset a and frame-rate ratio b, an estimate of the departure from synchronisation can then be expressed as

$$E_1(\mathbf{p}, \mathbf{p}', \mathbf{l}, \mathbf{l}') = \frac{1}{w} \sum_{h=1}^{m} \sum_{i=0}^{n_1-1} d(\mathbf{p}_{h,i}, \mathbf{l}_{h,a+bi,i})^2 + \\ \frac{1}{w} \sum_{h=1}^{m} \sum_{j=0}^{n_2-1} d(\mathbf{p}'_{h,j}, \mathbf{l}'_{h,c+dj,j})^2, \qquad (3)$$

where $(c = -a/b, d = 1/b)$, $d(\mathbf{p}, \mathbf{l})$ is the shortest distance between image point $\mathbf{p}$ and image line $\mathbf{l}$, and w is the total number of defined summands. Summands may be undefined due to unavailable image point information or frame indices that are out of bounds. The normalising factor w is used to avoid favouring synchronisation estimates that involve few summands due to small overlap in time.

In order to compute E_1, an estimate of $\mathbf{l}_{h,k,i}$ is needed for non-integer values of k. If the cameras remain stationary, such non-integer values can be handled by interpolating image points as done in [5]. This is insufficient for moving cameras, so epipolar line interpolation is used as a simple alternative. We compute a weighted interpolation of the two lines appearing in frame i with the nearest integer indices, $\mathbf{l}_{h,[k],i}$ and $\mathbf{l}_{h,[k]+1,i}$, with $[k]$ signifying the integer part of k. Each line is first normalised so that the sum of the squares of its first 2 elements equals 1. One of the pair of lines is then negated if this decreases their angular difference. The lines so normalised are denoted $\bar{\mathbf{l}}_{h,[k],i}$ and $\bar{\mathbf{l}}_{h,[k]+1,i}$. The interpolated line is then given by

$$\hat{\mathbf{l}}_{h,k,i} = ([k] + 1 - k)\bar{\mathbf{l}}_{h,[k],i} + (k - [k])\bar{\mathbf{l}}_{h,[k]+1,i}. \qquad (4)$$

An interpolated line $\hat{\mathbf{l}}'_{h,k,j}$ in the other video may be computed similarly. With these interpolated lines, an improved cost function E_2 can be defined by setting

$$E_2 \equiv E_1(\mathbf{p}, \mathbf{p}', \hat{\mathbf{l}}, \hat{\mathbf{l}}'). \qquad (5)$$

3. MINIMISING THE COST FUNCTION

If the ratio of the frame rates of the videos is known, then only a need be estimated. The usual approach is to perform a uniform, discrete sampling of a over a constrained range. For each sample value, a cost function (E_2 would be one possibility) is evaluated. The best value of a is then kept as an initial estimate.

In the case of unknown frame rates, it is prohibitively expensive to carry out a uniform discrete search over both a and b. The novel approach taken here is instead to search for real-valued frame pairs (f_1, f_2) for which there is precisely zero epipolar error in relation to a given, moving scene point. A large set of such pairs is then used to estimate the synchronisation parameters via a voting scheme.

The essential element of this strategy is as follows. Consider a frame i in video 1, and a consecutive pair of frames j and $j+1$ in video 2. Locate the point in each of these three frames that corresponds to a particular moving scene point. For each of the points in frames j and $j+1$, generate the associated epipolar line in frame i. If a simple interpolation of the two normalised lines can be made to pass precisely through the point in frame i, then compute the real-valued k that is *hypothesised* to be exactly synchronous with i. Term the (i, k) thus found a 'synchrony pair'.

The above procedure may be followed for all moving scene points, all frames in video 1, and all consecutive pairs in video 2. In this way a large set of synchrony pairs may be generated. Furthermore, a reciprocal procedure may be followed that instead uses video 2 as the base. Needless to say, this method of searching for synchrony pairs could be very expensive given lengthy videos and many image points. However, this search may be reduced via a random sampling of the input points and frames so as to compile a smaller set of these pairs.

If we now plot our computed synchrony pairs in $f_1 f_2$-space, then, in favourable circumstances, we may fit a line to the points, and set a to be the line's intercept with the f_2-axis, and b to be the gradient of the line (as inspection of equation (1) will confirm). This, however, is far from straightforward if the points are haphazardly spread, or there are secondary lines. For these reasons, we employ an alternative approach.

A single point in $f_1 f_2$-space specifies a line in ab-space. If ab-space is partitioned into a discrete grid over a suitable range of values, the (hypothesised) synchrony pair (f_1, f_2) can be associated with a line of these grid boxes in ab-space, all of which receive an additional 'vote' as a consequence. (This is a form of Hough Transform for lines, a good explanation of which may be found in [8].) This can then be performed for all of the remaining synchrony pairs. With the voting complete, the element with the most support within the surrounding 3×3 window of elements then
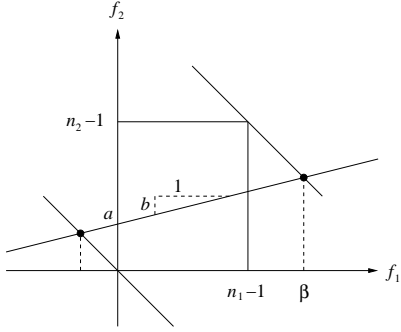
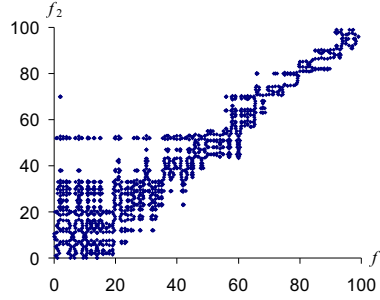**Fig. 1**. Line parameterisation (α, β).
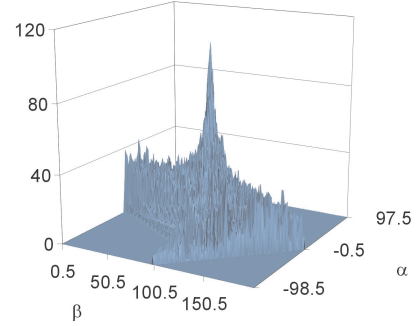

Fig. 2. Synchrony pairs.

Fig. 3. Resulting Hough Transform.

provides an initial approximation to the synchronisation parameters. These parameters can then be refined by applying a gradient-descent procedure (e.g. Levenberg-Marquardt) to the cost function, E_2.

The algorithm used to compute both synchronisation parameters in the moving camera case has therefore been outlined, with the exception of one factor. It emerges that (a, b) is not a convenient parameter space for the Hough transform, and that alternatives can be adopted that have much better properties. Our particular choice has simple bounds on both parameters, whereas b may theoretically increase without limit. Specifically, we apply the parameterisation

$$\alpha = \frac{-a}{b+1}, \quad \beta = \frac{n_1 + n_2 - 2 - a}{b+1}. \quad (6)$$

Figure 1 shows a line in $f_1 f_2$-space and the way that it is described by (α, β). Note that lines of gradient -1 pass through the origin and the point $(n_1 - 1, n_2 - 1)$. Figure 2 shows a typical example of synchrony points appearing in $f_1 f_2$-space for a single moving scene point. The corresponding histogram in $\alpha\beta$-space is given in Figure 3. A prominent peak is evident, with which is associated a good initial estimate of the synchonisation parameters. It should be noted that, under favourable conditions, this method for finding an initial estimate of (a, b) without random sampling is equal in complexity to a 1D search finding just a in the case where b is known. This voting algorithm is also applicable in the case of stationary cameras and image point interpolation.

4. RESULTS

Synthetic testing was undertaken. Note that this affords a systematic analysis of the characteristics of the methods, under a range of conditions. To this end, 50 stationary scene points were randomly chosen within a unit sphere, and were projected onto two cameras orbiting the sphere at different elevations. Random trajectories within the sphere provided

the means of generating the dynamic scene points. Gaussian noise with a standard deviation of 1 pixel was added to all image points, for an image size of 500×500. Projection matrices were precomputed for all frames in the two videos.

Given the true line parameters (a, b), the quality of the estimated parameters $(\hat{a}, \hat{b})$ was assessed by measuring errors in synchronised frame indices. Given frame i in the first video, $(a + bi)$ and $(\hat{a} + \hat{b}i)$ are the true and estimated corresponding frame indices from the second video. A measure of how well frame i has been synchronised is therefore $|(\hat{a} + \hat{b}i) - (a + bi)|$, and is referred to as the *frame synchronisation error*. Analogously, frame j in the second video has associated error $|(c + dj) - (\hat{c} + \hat{d}j)|$. Each video is assigned a *video synchronisation error* equal to the maximum of its frame synchronisation errors, for frames captured during the period when both cameras were recording.

Two main configurations were considered in the synthetic experiments, with true values of $a = 10.63$ and $b = 1.2$ in the first case, and $a = 40.6$ and $b = 1.1$ in the second case. In each case, the videos had lengths $n_1 = 80$ and $n_2 = 100$. Tests were carried out whereby both a and b were estimated for each of the cases $m = 1, 2, 5, 10$, corresponding to various numbers of tracked scene points.

Each case was tested 500 times, with new random scene points for each test. The medians of video synchronisation errors are shown in Table 1, for both the initial estimates found using the Hough transform, and the final estimates after minimisation of E_2 by Levenberg-Marquardt. Note that the median errors for the initial estimates remained unchanged as m was increased. The percentages of the tests for which the final estimates gave video synchronisation errors of less than 0.5 frames, described as 'successes', are shown in Table 2.

The results showed that, for both scenarios, the initial estimates of (a, b) typically produced low video synchronisation errors. The subsequent minimisation usually resulted in very small (sub-frame) errors, even from the use of just a single (moving) tracked scene point. These errors were

Initial Estimate (Hough Transform)				
m	$a = 10.63, b = 1.2$		$a = 40.6, b = 1.1$	
	Video 1	Video 2	Video 1	Video 2
1,2,5,10	1.207	0.998	0.235	0.213
Minimised Estimate (Levenberg-Marquardt)				
m	$a = 10.63, b = 1.2$		$a = 40.6, b = 1.1$	
	Video 1	Video 2	Video 1	Video 2
1	0.219	0.184	0.300	0.273
2	0.127	0.106	0.187	0.169
5	0.091	0.076	0.099	0.091
10	0.067	0.056	0.075	0.067

Table 1. Medians of video synchronisation errors

Percentage of Successes				
m	$a = 10.63, b = 1.2$		$a = 40.6, b = 1.1$	
	Video 1	Video 2	Video 1	Video 2
1	72.2%	75.6%	67.4%	70.2%
2	88.6%	89.4%	84.4%	86.2%
5	99.4%	99.8%	99%	99.4%
10	100%	100%	100%	100%

Table 2. Successful synchronisations

reduced further when more points were used, though with diminishing returns. As the number of points increased, the percentage of tests achieving synchronisation to less than half a frame also increased, reaching 100%. The median errors also showed that, for just 5 scene points, synchronisation to better than 0.1 frames can be expected for both scenarios.

Two real videos were also recorded, and synchronised with this new method. The initial and minimised estimates of (a, b) were $(-8.816, 0.603)$ and $(-9.309, 0.609)$ respectively. The camera specifications gave 0.6 as the true value of b. Well separated frames from video 2 are shown in figure 4, each with the pair of nearest-to-synchronous frames from video 1, aligned underneath to show relative position in time.

5. CONCLUSION

A cost function was presented based upon a measure of departure from synchronisation of videos generated by moving cameras. This involved an epipolar-line interpolating procedure enabling sub-frame estimation. An efficient and novel voting method was then used to find an initial minimiser of this function, with less complexity than a prohibitively full 2D search. Synthetic tests indicate that the method generally synchronises a pair of videos to acceptable sub-frame accuracy.
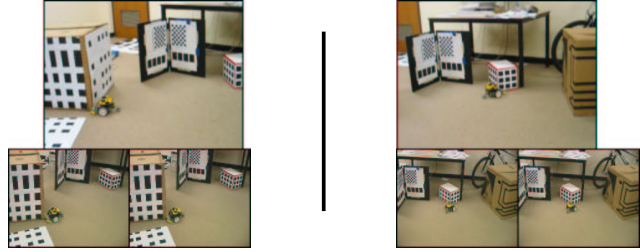

Fig. 4. Diagrammatic examples of synchrony.

6. REFERENCES

[1] I. Reid and A. Zisserman, "Goal-directed video metrology," in *European Conference on Computer Vision, Cambridge, UK, April 14-18, 1996*. vol. 2, pp. 647–658, Springer-Verlag.

[2] Y. Caspi and M. Irani, "A step towards sequence-to-sequence alignment," in *Conference on Computer Vision and Pattern Recognition, Hilton Head Island, South Carolina, June 13-15, 2000*, Los Alamitos, CA, vol. 2, pp. 682–689, IEEE Computer Society Press.

[3] G. P. Stein, "Tracking from multiple view points: Self-calibration of space and time," in *DARPA Image Understanding Workshop, Monterey, CA, November, 1998*. pp. 1037–1042, Morgan Kauffman.

[4] L. Lee, R. Romano, and G. Stein, "Monitoring activities from multiple video streams: Establishing a common coordinate frame," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 8, pp. 758–767, August 2000.

[5] Y. Caspi, D. Simakov, and M. Irani, "Feature-based sequence-to-sequence matching," in *ECCV Workshop on Vision and Modelling of Dynamic Scenes (VAMODS), 2002*.

[6] L. Wolf and A. Zomet, "Correspondence-free synchronization and reconstruction in a non-rigid scene," in *ECCV Workshop on Vision and Modelling of Dynamic Scenes (VAMODS), 2002*.

[7] Y. Caspi and M. Irani, "Alignment of non-overlapping sequences," in *International Conference of Computer Vision, Vancouver, July 7-14, 2001*, Los Alamitos, CA, vol. 2, pp. 76–83, IEEE Computer Society Press.

[8] J. Illingworth and J. Kittler, "A survey of the Hough transform," *Computer Vision, Graphics, and Image Processing*, vol. 44, pp. 87–116, 1988.