TEMPORAL SYNCHRONIZATION FROM CAMERA MOTION

Lisa Spencer and Mubarak Shah

School of Computer Science
University of Central Florida

ABSTRACT

This paper presents a method to recover the temporal synchronization between a pair of video sequences using the frame-to-frame motion of the sequences instead of pixel-based comparisons between the two sequences.

A previous method uses the similarity between corresponding frame-to-frame homographies. It works when the transformations are extremely accurate, but is not robust in the presence of noise. To overcome these problems, we have created and tested four new measures. All of the new measures perform well with either precise or noisy data for both real and synthetic sequences.

1. INTRODUCTION

There are an increasing number of applications in computer vision that use inputs from multiple video sequences. These applications require temporal synchronization of the video inputs. Since this synchronization is not always provided, it is often necessary to recover the temporal synchronization from the video sequences themselves.

Traditional methods assume that the cameras are viewing the same scene, and use pixel-based comparisons between sequences to determine time alignment. These methods cannot be used in situations where the camera fields of view do not overlap, when the cameras are different sensor types, or when the camera fields of view are vastly different, because features visible in one camera may not be present another.

In this paper, temporal alignment is recovered using the frame-to-frame motion of each of the input video sequences. Pixel-based methods may be used to find this motion, since successive frames of video from a *single* camera will have considerable overlap and similar camera parameters. Instead of comparing pixels between videos, the algorithm compares transformations. This makes it possible to align two videos that are viewing completely different scenes, as long as the cameras have the same motion.

Previous work is reported in Section 2. Section 3 defines the temporal synchronization problem. The mathematical models used to describe the frame-to-frame transformations are reviewed in Section 4. Five temporal synchronization evaluation measures are detailed in Section 5. Section 6 contains the experimental results for real and synthetic sequences, and conclusions are given in Section 7.

2. RELATED WORK

Caspi and Irani [3] proposed a method for using the camera motion to align two non-overlapping sequences taken from cameras with an unknown, but fixed relationship. In this process, the subtle differences between the frame-to-frame motions in the two sequences are used to find the inter-sequence homography. The same frame-to-frame motions are also used to find the temporal alignment by finding the time offset that results in the highest similarity between the two sets of transformations.

3. TEMPORAL SYNCHRONIZATION

Temporal synchronization is achieved by determining which frame of the second sequence corresponds to a given frame of the first sequence. Assuming that both cameras are capturing frames at the same constant rate, this correspondence can be described by a constant offset, Δt, which is the integer number of frames that separates the two sequences. If the two sequences are initiated by manually pressing the "record" button at the same time, this Δt will be a small number, less than a half second. Since 15 frames takes 2/3 of a second with NTSC (30 Hz.) and 3/5 of a second with PAL (25 Hz.), the range of [-15, 15] should be a sufficient interval to search for Δt.

In order to use the frame-to-frame motion of the sequences to recover the temporal alignment, the two cameras must be rigidly joined together and there must be non-uniform motion of the two cameras. If the two cameras are fixed, there will not be any global motion to detect. If the two cameras are on a platform with constant motion, like a motorized turntable, the frame-to-frame motion will be uniform for every frame, and the temporal synchronization cannot be automatically recovered using only the global motion.

More formally, given two unsynchronized sequences, S and S', taken by two moving cameras with an unknown but fixed relationship, we wish to find Δt such that frame i in sequence S corresponds to frame $i+\Delta t$ in sequence S'. The offset, Δt, is assumed to be an integer, and can be either positive or negative. For this discussion, we will assume that the cameras are using the same frame rate (not NTSC with PAL). If the rates are different, the method can still be used as long as appropriate temporal scaling is applied.

4. FRAME-TO-FRAME TRANSFORMATIONS

When the translation of the camera position is negligible compared to the distance to the scene or when the scene is planar, the global motion can be described by a homography (2D). When there is significant translation of the camera position between the two images and the scene is not planar, a fundamental matrix must be used to describe the global motion (3D).

The measures presented here for evaluating the temporal alignment can be used for both these cases. The only restriction on the inter-sequence transformation is that the relationship between the cameras is rigid.

4.1. Frame-to-frame homography

A homography models the relationship between the location of a feature at (x, y) in one frame and the location (x', y') of the that same feature in the next frame with nine parameters, as shown in Equation 1.

$$x' = \frac{h_{11}x + h_{12}y + h_{13}}{h_{31}x + h_{32}y + h_{33}}, y' = \frac{h_{21}x + h_{22}y + h_{23}}{h_{31}x + h_{32}y + h_{33}} \quad (1)$$

The homography is an exact description of the transformation for a projective camera model when the scene is planar or the camera is stationary (no translation, only rotation). However, due to the computational complexity of solving a nonlinear of set of equations like Equation 1, a simpler linear approximation is often used.

A simplified way to represent the frame-to-frame motion for a stationary camera or planar scene is the affine motion model, shown in Equation 2. This model is linear, and uses six parameters.

$$x' = a_1 x + a_2 y + b_1, y' = a_3 x + a_4 y + b_2 \quad (2)$$

The coefficients of the affine transformation can be used in simple expressions to describe the frame-to-frame motion of a pixel [2]. The horizontal translation is given by b_1, and the vertical translation is given by b_2. The divergence (isotropic expansion) is $a_1 + a_4$, the curl (roll, or rotation about the viewing direction) is given by $a_3 - a_2$, and the deformation (squashing or stretching) is given by $a_1 - a_4$.

There are well-documented methods for determining the motion parameters, including a hierarchical model [1] and refinements for excluding local motion [7].

4.2. Frame-to-frame fundamental matrix

When there is significant camera motion and the scene is not planar, the relationship between successive frames can be described by a fundamental matrix. The fundamental matrix, F, relating two images is defined as the 3×3 matrix such that $p'^T F p = 0$, where p is the point (x, y) in ome image and p' is the point (x', y') in the other image. Since this equation can clearly be multiplied by a positive or negative scalar, F can only be determined up to a scale factor.

If the camera matrices P and P' are known, the fundamental matrix can be computed by

$$F = [e']_\times P' P^+ \quad (3)$$

where e' is the epipole defined by $e' = P'C$, C is the center of the first camera defined by $PC = 0$, $[v]_\times$ is the anti-symmetric matrix such that multiplying this matrix by any vector will produce the same result as the cross product of vector v and the vector, and P^+ is the pseudo-inverse of P defined by $PP^+ = I$.

If the fundamental matrix F is known, a pair of cameras $P = [I|0]$ and $P' = [M|m]$ can be found for which F is the fundamental matrix [6]. We can use this M for recovering the temporal synchronization for the 3D case (when the camera is moving) in the same way that the homography is used in the 2D case (when the camera is not moving).

5. TEMPORAL SYNCHRONIZATION EVALUATION MEASURES

The temporal synchronization process consists of evaluating some error measure for each value of Δt in a given range of positive and negative integers. The value of Δt with the smallest error corresponds to the best alignment of the two sequences. This is shown in Equation 4.

$$\Delta t = \arg \min_{\Delta t \in [-r, +r]} error(T_i, T'_{i+\Delta t})_{i=1+r}^{n-r} \quad (4)$$

where r defines the range of integer frame offsets to search, and the error measure compares tranformation T_i from sequence S with transformation $T'_{i+\Delta t}$ from sequence S' for a range of values of i. If n is the number of frames in each sequence, the range of i is $1 + r$ to $n - r$ instead of 1 to n to keep the number of frames evaulated for each Δt constant, as well as to ensure that $i + \Delta t$ is between 1 and n.

Several different error measures to use for $error(X, Y)$ are described in the next sections, followed by experimental results showing the effectiveness of each of the measures.

5.1. Similarity

The measure used by Caspi and Irani [3] to compare transforms from the two sequences is *similarity*. If T_i in sequence S occurred at the same time as $T'_{i+\Delta t}$ in sequence S', they are related as shown in Equation 5.

$$T'_{i+\Delta t} = HT_iH^{-1} \tag{5}$$

In Equation 5, H is the 3×3 transformation matrix that maps pixels from an image in sequence S to pixels in the corresponding image in S'. Therefore, T_i and $T'_{i+\Delta t}$ are similar matrices, so the vector formed by the eigenvalues of T_i should be parallel to the vector formed by the eigenvalues of $T'_{i+\Delta t}$, according to linear algebra properties of similar matrices [8]. The degree of similarity can be quantized by measuring the cosine of the angle between the two vectors of eigenvalues, using a dot product, as shown in Equation 6.

$$sim(A,B) = \frac{eig(A)^T eig(B)}{|eig(A)||eig(B)|} \tag{6}$$

In Equation 6, $|V|$ represents the magnitude of vector V and $eig(M)$ represents the vector formed by the eigenvalues of the matrix M. This will give a value of 1.0 for matrices that are perfectly similar, and decrease to 0.0 as the similarity degrades.

The similarity error measure consists of the sum of one minus the similarity of each pair of transforms for a given value of Δt, as shown in Equation 7. Good values for Δt are expected to produce similarities close to one, resulting in small values of $error_{SIM}$.

$$error_{SIM}(T_i, T'_{i+\Delta t}) = \sum_{i=1+r}^{n-r} (1 - sim(T_i, T'_{i+\Delta t})) \tag{7}$$

This measure was originally intended for use with a 2D sequence, where the camera translation was negligible. It is still valid when the cameras are allowed to translate, because the sequence-to-sequence homography still maps planar points and maintains the relationship in Equation 5.

In practice, the similarity calculations resulted in a number very close to one (like 0.99998), even when the wrong Δt was used. To understand why, the three eigenvalues of the 3×3 affine transform coefficient matrix were computed analytically, and found to be 1, $\frac{1}{2}(a_1 + a_4 + \sqrt{a_1^2 + 4a_2a_3 - 2a_1a_4 + a_4^2})$ and $\frac{1}{2}(a_1 + a_4 - \sqrt{a_1^2 + 4a_2a_3 - 2a_1a_4 + a_4^2})$. There is no dependency on b_1 or b_2, the global pixel translation. Several other measures were created to take advantage of global pixel translations and other easily observable global motion in order to find a method more tolerant to inaccuracies.

The two common quantities derived from the global pixel translation (b_1 and b_2 in the affine model) are the magnitude and direction of the translation. These are used for the first two new measures.

5.2. Translation magnitude

In most cases, a larger motion in one sequence should correspond to a larger motion in the other sequence, since the cameras are rigidly connected. The *translation magnitude* measure is based on this idea. It calculates the correlation coefficient of the translation magnitude in each pair of corresponding frames for a given Δt.

The correlation is ± 1 when two random variables are linearly dependent. Since a larger translation in one sequence corresponds to a larger translation in the other sequence, we are only concerned with positive correlation. The best temporal alignment should be at the value of Δt where the correlation is closest to +1. We used one minus the correlation as the error measure, so the best alignment occurs when the error is the minimum. Equation 8 defines the error from Equation 4 for the translation magnitude measure.

$$error_{tm}(T_i, T'_{i+\Delta t}) = 1 - corr[tm(T_i), tm(T'_{i+\Delta t})] \tag{8}$$

where $tm(T_i)$ is

$$tm_{affine} = \sqrt{b_1^2 + b_2^2} \tag{9}$$

$$tm_{homography} = \sqrt{\frac{h_{13}^2 + h_{23}^2}{h_{33}^2}} \tag{10}$$

for the affine model and the full 3x3 homography respectively.

5.3. Translation direction

The next measure is based on the idea that when one camera moves in some direction, the other should move in a related direction. The relation depends on the relative orientation of the two cameras.

The *translation direction* measure calculates the correlation of the translation direction in one sequence with the translation direction in the other. One minus the absolute value of the correlation is used as the error measure. The absolute value allows for negative correlation, such as when the cameras are pointing in opposite directions. The error for the translation direction measure is given in Equation 11.

$$error_{td}(T_i, T'_{i+\Delta t}) = 1 - |corr[dir(T_i), dir(T'_{i+\Delta t})]| \tag{11}$$

where $dir(T_i)$ is defined as:

$$dir_{affine} = \tan^{-1} \frac{b_2}{b_1} \tag{12}$$

$$dir_{homography} = \tan^{-1} \frac{h_{23}/h_{33}}{h_{13}/h_{33}} \tag{13}$$

for the affine model and full 3x3 homography respectively.

Before calculating the correlation, the angles from the second sequence were adjusted so that they were within $180°$ of the angles from the first sequence by adding or subtracting $360°$. Without this adjustment, $+179°$ and $-179°$ would appear to be $358°$ apart, instead of $2°$ apart.

The translation direction is undefined when the translation magnitude is zero ($b_1, b_2 = 0$), and unstable when the translation magnitude is very small. To prevent these spurious angles from degrading the result, a weighted correlation is used. The angles are weighted by the square of the product of the translation *magnitudes* for the corresponding frames.

5.4. Roll motion

After creating measures based on the global translation, the other expressions easily obtained from the affine model (listed in Section 4.1) were considered. The divergence is not a good candidate, since the zoom during the sequence is fixed, and the camera does not change its distance from the scene for the 2D sequences. The deformation is also not expected to change during the sequence. This leaves curl, defined as rotation about the optical axis, and also called roll motion.

When the two cameras are pointed in roughly the same direction, a roll motion in one sequence should correspond with a roll motion in the other sequence. The *roll* measure calculates the correlation between the roll motion (curl) of corresponding frames of the two sequences. One minus the absolute value of the correlation is used as the error measure. The absolute value allows for negative correlation, such as when the cameras are pointing in opposite directions. Equation 14 defines the error for the roll measure.

$$error_{roll}(T_i, T'_{i+\Delta t}) = 1 - |corr[roll(T_i), roll(T'_{i+\Delta t})]| \tag{14}$$

where $roll(T_i)$ is defined as:

$$roll_{affine} = a_2 - a_3 \tag{15}$$

$$roll_{homography} = h_{12} - h_{21} \tag{16}$$

for the affine model and the full 3x3 homography respectively.

5.5. Combined measure

Measures based on translation are expected to perform better in sequences exhibiting larger and more diverse translation. Measures based on roll should work better when there is significant rotation. The correlation not only indicates the best Δt for a given measure, but also quantifies the quality of the match. We can incorporate both the translation magnitude and roll motion measures in a combined measure that will work for either type of motion, using the correlations to compute a weighted average.

6. EXPERIMENTAL RESULTS

The five measures described in the previous section were used to evaluate the temporal alignment of both synthetic and real sequences, for both 2D and 3D cases. The synthetic data was used in both its original form (ground truth) and with noise added. The sequences and graphs of the results are available on the web page: http://www.cs.ucf.edu/∼vision/projects/time_shift/time_shift.htm.

6.1. Synthetic data

Synthetic data was generated in a virtual environment. The actual content of the images was not used to calculate the time shift; only the positions and orientations of the cameras were used. Two sets of sequences were generated; one in which the camera translated and rotated (3D), and the other in which the camera only rotated (2D). Each set contained four sequences of 60 frames. Within a set, each sequence used the same reference motion, but with a different angle offset from the reference viewpoint.

6.2. Perfect synthetic data results

The frame-to-frame homographies for each sequence of the 2D set (no camera translation) and fundamental matrices for each sequence of the 3D set (translating camera) were calculated from the ground truth motion. The five methods described in the previous section were used to evaluate the best time shift value for each of the six possible combinations of the four sequences for both the 2D and 3D data sets. All five error measures correctly showed a minimum error at $\Delta t = 0$ in all cases.

6.3. Noisy synthetic data

Noise was introduced to the synthetic data by using the ground truth to find a set of correspondences in each pair of successive frames in each sequence. These corresponding points were rounded, then used to generate frame-to-frame homographies or affine transformations for the 2D sequences using the direct linear transformation (DLT) algorithm [6], or frame-to-frame fundamental matrices for the 3D sequences using the 8-point algorithm [5]. This simulates uniform noise equivalent to a matching algorithm that always finds the best pixel, but doesn't refine the match to full precision subpixel accuracy.

A sample of the results for noisy 2D and 3D using pairs of the synthetic sequences is shown in Figure 1. The similarity measure is incorrect for all six 2D homography cases, four of six 2D affine cases, and four of six 3D cases. The four new measures correctly show the minimum at $\Delta t = 0$ in all cases.

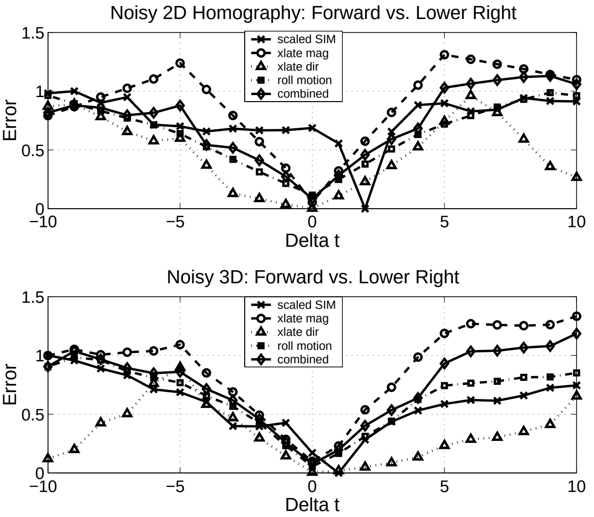

Fig. 1. Noisy synthetic data results. The correct alignment is $\Delta t = 0$.

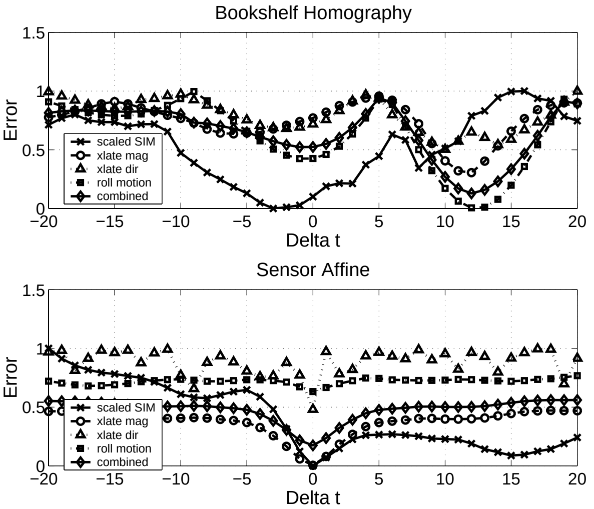

Fig. 2. Results from the "Bookshelf" and "Sensor" sequences. The correct alignments are $\Delta t = 12$ and 0.

Since the similarity measure performed much worse for both the noisy homography and the noisy affine data than for the perfect homography, it would appear that the accuracy of the transformations has a greater impact on the similarity measure than the choice of model.

6.4. Real data

Several pairs of real video sequences were analyzed using all five error measures. The results are summarized in Table 1. The "HAF" column indicates whether the frame-to-frame tranformation was modeled with a homography (H), affine (A), or fundamental matrix (F). The numbers indicate the value for Δt.

Seq	HAF	Sim	TMag	TDir	Roll	Comb
Bookshelf	H	-3	12	10	12	12
	A	12	12	15	12	12
Sensor	H	-4	0	0	0	0
	A	0	0	0	0	0
Football	H	0	0	0	5	0
	A	-20	0	0	4	0
Haifa	H	-3	0	0	0	0
	A	0	0	0	0	0
Seq2	H	+20	0	0	0	0
	A	2	0	0	0	0
Corner	F	1	1	1	1	1

Table 1. Results from real sequences for the five methods

In the "Bookshelf" sequence, two cameras with approximately the same field of view were mounted on a tripod, with one vertical and the other sideways. The sequence consists primarily of roll motion, with very little translation (pan or tilt). The correct offset is $\Delta t = 12$. As expected, the roll measure performed better than the translation measures, and the error was lower as well. The combination method chose the correct frame offset. A plot of the error for this sequence using the homography is shown in Figure 2.

In the "Sensor" sequence of Caspi and Irani [4], one sequence used a regular video camera and the other used an infrared camera. There is minimal camera movement at the beginning of the sequence, and the motion present is primarily horizontal, with minimal roll. All of the new measures produced the correct time offset, and as expected, the roll motion measure had larger errors than those based on translation. The similarity measure produced the wrong result when using homographies. The results of applying the four measures using the affine model are shown in Figure 2.

Caspi and Irani used very accurate homographies to get the correct answer with the similarity measure. Our method was able to get the correct time offset with less accurate homographies as well as the simplified affine model.

The "Football" sequence of Caspi and Irani [4] consists of two cameras with approximately the same field of view that are aligned side-by-side, with minimal overlap. The sequence is dominated by views of buildings. Even though the roll measure produced an incorrect value for the time shift, the combination method still achieved the correct result.

In the "Haifa" sequence of Caspi and Irani [4], one camera is zoomed into a small portion of the field of view of the other camera, through a fence. Our new measures correctly found the time shift even with imperfect homographies and using the affine model.

"Seq2" is also from Caspi and Irani. Two cameras with different fields of view are pointed at a building. The camera has significant pan and tilt as well as roll motion. All of the new methods found the corrent time shift using homographies that were not accurate enough for the similarity measure, as well as with the affine motion model.

The final real sequence was the "Corner" sequence. The cameras in this sequence had significant motion, requiring the use of the fundamental matrix for the frame-to-frame transformations. One camera was digital, while the other was analog. They were pointing in roughly the same direction, with different zoom factors. All the measures found the correct time offset in this 3D sequence.

6.5. Summary

Table 2 shows the overall results for the four different temporal shift measures. For the synthetic data, the results are shown for all six possible 2D and six possible 3D combinations. The noisy data additionally includes the six 2D affine cases. "Correct" indicates that the algorithm uniquely identified the correct time shift.

Measure	Perfect Syn H and F	Noisy Syn H, A and F	Real H, A and F
Similarity	All 12 correct	14 of 18 wrong	6 of 11 wrong
Translation Magnitude	All 12 correct	All 18 correct	All 11 correct
Translation Direction	All 12 correct	All 18 correct	2 of 11 wrong
Roll Motion	All 12 correct	All 18 correct	2 of 11 wrong
Combination	All 12 correct	All 18 correct	All 11 correct

Table 2. Performance Summary for the Five Methods

7. CONCLUSION

Based on the summary data, the combination measure appear to be a good metric for recovering the temporal alignment for real or noisy data for both 2D (minimal camera translation) and 3D (significant camera translation). The measures based on translation have difficulty when there is little translation, and the measures based on roll don't work well when there is minimal roll motion. Instead of choosing roll or translation, the combination measure can operate on sequences both types of sequences, and still determine the correct time alignment. The similarity measure does not seem to be able to distinguish between noise and misalignment.

The new measures gave much better results than the previously proposed similarity measure in the presence of noise. We have shown that the similarity measure cannot handle inaccuracies in the frame-to-frame transformations. This level of accuracy is not a reasonable assumption, especially in the presence of local motion. In addition, if the application does not require perfect transformations, it should not be necessary to spend extra time refining the spatial alignment just to determine the temporal alignment.

One possible improvement on this process is to use the temporal alignment obtained this way to find the spatial alignment between the two cameras. The spatial alignment can then be used to determine more accurately the relationship between the frame-to-frame transformations, which in turn can be used to refine the temporal alignment.

Applications that use inputs from multiple video sources require temporal synchronization. We have demonstrated new methods that can be used when external synchronization is not available. These methods work by extracting information from the frame-to-frame transformation matrices and using correlation to find the best temporal offset. The methods were demonstrated on synthetic and real 2D and 3D sequences, with both perfect and noisy frame-to-frame transformations. The measure that uses a combination of translation and roll rotation was correct in all of the experiments.

8. REFERENCES

[1] J. Bergen, P. Anandan, K. Hanna and R. Hingorani, "Hierarchical Model-Based Motion Estimation," *European Conference on Computer Vision*, pp. 237-252, 1992.

[2] M. Black and Y. Yacoob, "Tracking and Recognizing Rigid and Non-Rigid Facial Motions using Local Parametric Models of Image Motion," *Proceedings of the Fifth International Conference on Computer Vision*, pp. 374-381, 1995.

[3] Y. Caspi and M. Irani, "Alignment of non-overlapping sequences," *Proceedings of the Eighth International Conference On Computer Vision*, pp. 76-83, 2001.

[4] Y. Caspi and M. Irani, "Alignment of Non-Overlapping Sequences," http://www.wisdom.weizmann.ac.il/NonOverlappingSeqs

[5] R. Hartley, "In Defence of the 8-point Algorithm," *Proceedings of the Fifth International Conference on Computer Vision*, pp. 1064-1070, 1995

[6] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2000.

[7] M. Irani, B. Rousso, S. Peleg, "Computing Occluding and Transparent Motions," *International Journal of Computer Vision*, 12(1):5-16, February 1994.

[8] C. Pearson (ed.), *Handbook of Applied Mathematics - Second Edition*, Van Nostrand Reinhold Company, 1983.