Video Synchronization via Feature Correspondence

André L. G. Ruas
Universidade Federal de Minas Gerais
Department of Computer Science
Av. Antonio Carlos , 6627 - Belo Horizonte, Brazil
algruas@gmail.com

Abstract

in order to correctly reconstruct a dynamic scene via stereopsis we are required to have images obtained at the same time instant, if we are to apply stereopsis to a pair of video sequences we have to guarantee that the video sequences are synchronized.

Obtaining the synchronization for video sequences can be complex and potentially costly task. This paper treats the problem of estimating the temporal relation between two unsynchronized video sequences obtained from the same scene. It describes a simple algorithm for estimating the temporal offset of a pair of video sequences. The algorithm is based on pairwise frame correspondence using a similarity measure for estimating the temporal similarity between two given frames. The matching is performed by searching for every frame in one of the sequences the frame within a temporal search region in the other sequence that maximizes the similarity. We describe a method by which the frame search region for the corresponding frame can dynamically, estimated, prove that under certain statistical assumptions the corresponding frame is expected to lie inside the search region and empirically demonstrate the effectiveness of our method based on experiments performed with ground truth data.

1. Introduction

In multiple view scene reconstruction, the images obtained from the scene are required to be acquired at the same time instant in order to effectively recover the 3D location of a projection point via triangulation. When reconstructing a dynamic scene from video sequences we are required to know the temporal alignment between the sequences. Although some times this temporal relation can be given to us by hardware, or obtained manually, there are cases, such as prerecorded videos, in which hardware synchronization is not available, and manual synchronization is too time consuming. In such cases other software synchronization methods can provide significant savings in time and cost. The temporal relation between the sequences is estimated in this paper by using a sequence to sequence correspondence strategy, and by fitting the correspondences to a line. It searches for a given frame at one of the sequences (the reference sequence) the corresponding frame in the target sequence as the frame within a search region in the other sequences that maximizes a temporal frame similarity measure, the similarity between to frames is measured as inverse to the "noise" estimated in an affine stereo self calibration. Having obtained the correspondence the temporal alignment of the sequences is then matched to a linear model, We present a method by which the search region for the corresponding frame is dynamically estimated based on the absolute errors of the past correspondences and show that it tends to encompass the correct corresponding frame under reasonable assumptions.

The next section provides a quick revision of existing methods for estimating the synchronization and some other methods related to this work, Section 3 states assumptions and provides a formal definition of the problem being treated, Section 4 describes how the synchronization can be achieved and closes by defining an algorithm for estimating the synchronization, Section 5 describes implementation and test platform as well as show results to some of the tests, Section 6 presents the author's conclusions about the method and comments on possible improvements.

2. Related Work

Tomasi and Kanade[11] noted that, under the assumption of an affine projection, a $2V \times N$ matrix of N point projection across V views (Measurement matrix) is given by a product of a $2V \times 3$ projection matrix and a $3 \times N$ matrix containing the world points. and by consequence the rank of a perfectly matched measurement matrix is bounded above by 3.

Wolf and Zomet[13] noted that the rank 3 constraint

for the measurement matrix is only valid for synchronized scenes and proposed an correspondence free algorithm for synchronizing multiple sequences of the same scene by minimizing the rank of a tracking state stack matrix. Tresadern and Reid[12] based on the work of Wolf and Zomet, developed a method for temporal alignment of image sequences of the same scene which uses as a similarity measure the the smallest singular value of a measurement matrix composed by interest point matching and correspondence, and deals with outliers in the sequence matching by using the RANSAC[4] algorithm. The problem of temporal alignment was also treated by Caspi et al[10] who developed an algorithm which synchronizes based on the correspondence between object trajectories obtained in both sequences and Cardeal[3] in his thesis improved on the method proposed by Caspi by extending it to N different views for any arbitrary $N > 2$, this work was inspired on the work of Reid and improves on that work in the sense that the search region for the matching frames is dynamically estimated by means of absolute error in the frame correspondence and by doing so decreases the ambiguity generated by periodic movements in the scene, and produces a great increase in speed.

As in the method proposed by Reid and Tresadern, our method has to deal with the problem of interest point detection and matching, they resolved point correspondence issues by using a set of markers to define the interest points, in this work we automatically detect and match points using only image information by using a feature detector (SURF). The problem of interest point detection is traced at Moravec(1981)[9] which detected a corner as a point in the image with low self similarity, his work was followed by Harris(1988)[6] who developed Harris corner detector probably the most widely used corner detector in history, the main problem with these detectors is the influence that image scale has on the result. Lindeberg[7] introduced the concept of automatic scale selection, and Lowe[8] used difference of gaussian to perform scale space analysis and create a method for point detection and correspondence named SIFT. Bay[1, 2] focusing on, efficiency, approximated the difference of gaussian filter with box filters to create a interest point detector called SURF which is robust and extremely efficient when compared to other methods.

3. Problem Formulation

Assuming we have 2 sequences to be aligned, let the sequences be defined by $\mathbf{S}_1 = \{f_1^1, f_2^1, \dots f_n^1\}$ and $\mathbf{S}_2 = \{f_1^2, f_2^2, \dots, f_m^2\}$ where f_j^i correspond to the j-th frame of the i-th sequence, let $\tau(f_i^j) = \tau_j(i)$ be time of capture the frame f_i^j, the problem of temporal alignment can be formally defined as the problem of finding an relation $r(i,j) = \{\tau_1(i) = \tau_2(j)\}$.

Now assuming that both sequences have constant framerates defined by FR_1 and FR_2 we have:

$$\tau_j(i) = \tau_j(0) + FR_j \cdot i$$

$$\tau_1(i) = \tau_2(j) \Rightarrow$$
$$\tau_1(0) + FR_1 \dot{i} = \tau_2(0) + FR_2 \cdot j \Rightarrow$$
$$i = \frac{\tau_2(0) - \tau_1(0)}{FR_1} + \frac{FR_2}{FR_1} \cdot j = \alpha + \beta j$$

And the problem of temporal sequence alignment is simplified to finding a line $i = \alpha + \beta j$.

4. Methodology

4.1. Synchronization Rating.

If the depth variation of the observed scene is relatively small we can approximate that the projection (u^c, v^c) in the c-th view of every point (X, Y, Z) in the scene by an affine projection given by:

$$\begin{bmatrix} u^c \\ v^c \end{bmatrix} = \mathbf{P}^c \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

Now if we have n different points on the scene we can define a measurement matrix $\mathbf{W}$ as:

$$\mathbf{W} = \begin{bmatrix} u_1^1 & u_2^1 & \dots & u_n^1 \\ v_1^1 & v_2^1 & \dots & v_n^1 \\ u_1^2 & u_2^2 & \dots & u_n^2 \\ v_1^2 & v_2^2 & \dots & v_n^2 \end{bmatrix} = \begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_2 \end{bmatrix} \begin{bmatrix} \mathbf{X}_1 & \mathbf{X}_2 & \dots & \mathbf{X}_n \end{bmatrix}$$

where every $[u_i^c v_i^c]^T$ correspond to the projection of the i-th scene point observed by the c-th view.

This matrix for perfectly matched points projected at the same time instant has rank bounded above by 3 as was noted by Tomasi and Kanade, not only that if $[u_i^1 v_i^1]$ and $[u_i^2 v_i^2]$ are projections of the same object point at a different time instant and the object point is at different positions in the corresponding instants than the rank of $\mathbf{W}$ the rank bound does not apply as noticed by Wolf and Zomet.

If we define a sequence of M frames to be a window frame, and one of the two sequences S_1 and S_2 as the reference sequence S_{ref} and the other as the target sequence S_{tg} we can define a matrix $W(F, f)$ for each frame $F \in S_{ref}$ $f \in S_{tg}$ as:

$$W(F,f) = \begin{pmatrix} u_{ref,1}^F & \dots & u_{ref,n}^F & u_{ref,1}^{F+1} & \dots & u_{ref,n}^{F+M-1} \\ v_{ref,1}^F & \dots & v_{ref,n}^F & v_{ref,1}^{F+1} & \dots & v_{ref,n}^{F+M-1} \\ u_{tg,1}^f & \dots & u_{tg,n}^f & u_{tg,1}^{f+1} & \dots & u_{tg,n}^{f+M-1} \\ v_{tg,1}^f & \dots & v_{tg,n}^f & v_{tg,1}^{f+1} & \dots & v_{tg,n}^{f+M-1} \end{pmatrix}$$

where $[u^j_{c,i} v^j_{c,i}]^T$ corresponds to the projection of the i-th feature in the j-th frame of the sequence c.

If we have perfectly identified and matched N features points across the video frames we can expect that the rank of $W(F, f)$ to be 3 only if frames F and f were captured at the same time instant, in practice however noises generated by the camera, pixelization and incorrect matching almost always will cause $\mathbf{W}(F, f)$ to have full rank.

By decomposing the matrix $\mathbf{W}(F, f)$ in singular values we can expect the smallest singular value of $W(F, f)$ to be determined only by the system noise if F and f were captured at the same instant, and by noise plus some amount otherwise, so we can search for every frame $F \in S_{ref}$ the frame $f \in S_{tg}$ that minimizes the smallest singular value of $W(F, f)$ as the corresponding frame and fit the obtained pair (F, f) to a line.

Figure 1. Synchronized frames.

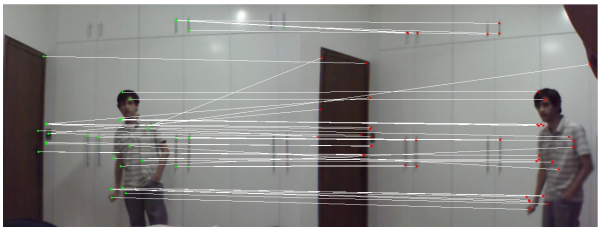

Figure 2. Unsynchronized frames.

The $W(F, f)$ when applied to the perfectly synchronized frames in Figure.1 have a smallest singular value of 23.34 (in pixels), when applied to the unsynchronized frames in Figure.2 have a smallest singular value of 206.49

4.2. Search region for synchronized frame

One issue with this method is that if we have ambiguity in the scene (like periodic movement)[12, 3] the outliers generated will reduce the quality of the result, and the other is that searching every frame in the target sequence can consume a large amount of time.

If the space being examined is large the probability of generating false matchings increases and the outliers generated in the correspondence can compromise the quality of the result. The solution to the outlier problem proposed in Reid's work is to use an method robust to outliers like RANSAC[4], but this would also have to deal with the problem of the space search. This work deals with both problems by dynamically estimating a search region for the corresponding frame that is centered at the expected location for the corresponding frame and decreases in size proportional to the estimated absolute error of the collected in the last k frames.

The estimation method works as follows:

given an initial estimation for the temporal correspondence ($j = \alpha + \beta \cdot i$) and an initial search range SR_0.

1. for every frame $F_i \in S_{ref}$, search frames between $f_{\alpha+\beta i-SR_i}$ and $f_{\alpha+\beta i+SR_i}$ for matching frame f_j using the singular value metric.
2. using least squares method update estimation of $j = \alpha + \beta \cdot i$.
3. obtain absolute error AE for the last k points.
4. update search range $SR_i \leftarrow \gamma \cdot SR_{i-1} + \delta \cdot AE + \epsilon$

k, γ, δ and ϵ are constants defined empirically where $0 \leq \gamma \leq 1$ and δ is proportional to $\frac{1}{k}$. Intuitively what the search range update does is smoothly increase the search range as the absolute error in the last points become greater and smoothly decrease when the absolute error becomes smaller. ϵ ensures that the search range doesn't drop to zero. The search range variation is:

$$SR_i - SR_{i-1} = \gamma \cdot SR_{i-1} + \delta \cdot AE + \epsilon - SR_{i-1} =$$

$$\delta \cdot AE + \epsilon - (1 - \gamma) SR_{i-1}$$

So the search range increases whenever $\delta \cdot AE + \epsilon > (1 - \gamma) SR_{i-1} \Leftarrow AE > \frac{(1-\gamma)SR_{i-1}-\epsilon}{\delta}$ and decreases otherwise, and the search range never becomes smaller then $\frac{\epsilon}{1-\gamma}$.

Also if the corresponding frame is not within the search region, if we assume that the matching will return a random frame in the search range we can expect the average absolute error for the frame i frame to half the search range $AE = \frac{SR_{i-1}}{2}$ if we are considering the search range for the last k frames then the expected absolute error will be on average $\sum_{l=i-k}^{i-1} SR_l$ it can be proven that under these assumptions, given good choices for k , γ, δ and ϵ if we are looking for the matching frame in the wrong region then the search range grows exponentially. The proof for the case where $k = 1$ follows:

$$SR_i = \gamma \cdot SR_{i-1} + \delta \cdot AE + \epsilon \, and \, AE = \frac{SR_{i-1}}{2} \Rightarrow$$

$$SR_i = (\gamma + \frac{\delta}{2}) \cdot SR_{i-1} + \epsilon$$

So if we choose γ and δ such that $(\gamma + \frac{\delta}{2}) > 1$ for $k > 1$ we can say that the search range grows exponentially while the corresponding frame lies outside the search region.

Now if we assume that a good matcher will give us the corresponding frame with an absolute error that is in average a constant C the search region tends to:

$$SR_i = \frac{\delta \cdot C + \epsilon}{1 - \gamma}$$

So under these assumptions our search range will decrease if the corresponding frame lies within the search region and will exponentially increase if the corresponding frame is not in the search region we are looking for. So the corresponding frame can be expected to be included in the search region for the next frames.

Note that the static assumptions made here are not valid for every scene, if the scene presents periodic movement the search region can converge to a local minimum if the initial search region is poorly estimated. However if the scene presents little ambiguity then the assumption is a reasonable one, one possible way to deal with local minimums due to periodic movement is to randomize the matching order of the reference sequence frames, as the random order of the matching decreases the chance that the search region will include a local minimum.

4.3. Point detection and correspondence.

In order to construct matrix W we have to detect and match interest points in the frames being examined.

For the point detection and matching we chose to work with Bay's detector SURF based of it's efficiency and the robustness of it's descriptors.

the SURF detector return's a set of image points and descriptors for each point, the point descriptors are based on haar like features and are returned as a set of values that can either have 64(regular SURF) or 128(extended SURF) different values. In both cases the matching is performed by comparing the descriptors. To obtain the best correspondence possible we would have to examine all the matching possibilities, but to do so would be unreasonable given the size of the matching possibilities space. Matching heuristics like Best-Bin-First[8] and KD-trees due to Friedman[5] have been show to provide good results with SURF descriptors, for the present work we simply search for each point in the reference frame the point whose descriptor have the closest euclidian distance in the target frame with acceptable results for the frame correspondence.

Algorithm1 provides the pseudo-code for the method designed in this section.

A preliminary estimative for the temporal alignment can be obtained by matching some random frames and generating a line using an robust estimator, or it can be assumed that the search region initially contains all target sequence frames.

Algorithm 1 Synchronization Algorithm

```
generate a primary estimative for the line i = α + βj
initialize searchRadius with initial value for search Radius for corresponding frame.
for each frame F_j ∈ S_ref do
    expected ← α + βj
    smallestSingularValue ← ∞.
    for i ← expected − searchRadius to expected + searchRadius do
        detect and match interest points in frame F_j ∈ S_ref and f_i ∈ S_tg.
        generate matrix W(F_j, f_i) based on the matches.
        decompose W using SVD.
        singularValue ← W'ssmallestSingularValue.
        if singularValue < smallestSingularValue then
            smallestSingularValue ← singularValue
            match ← i
        end if
    end for
    update line i = α + βj by linear regression.
    estimate searchRadius for the next frame.
end for
```

5. Implementation and Results

Our algorithm was implemented using the language C using openCV library our experiments were conducted using an AMD Atlhon x2 5200+ with 2 Gigabytes of RAM, running under windows operating system. To obtain ground through data we used a pair of Neptune webcam's with focal length of 4.3 mm and maximum resolution of 2 mega pixels, the videos obtained videos were algorithmically synchronized via computer clock to generate ground through data. And manually unsynchronized for the testing of out method. In all experiments the video frames had a resolution of 640×480 pixels and SURF was applied using a threshold for the hessian of 250 and extended descriptors, compare window size M of $W(F, f)$ was fixed at 3 frames.

Figure.3 shows the result of the frame correspondence while searching all frames in the target sequence for the

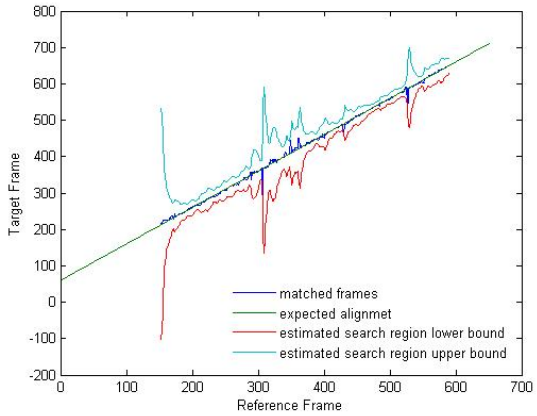

Figure 3. temporal alignment estimation 1.

corresponding frame in total 450 reference frames were matched against 800 target frames, the sequences had the same frame-rate and an initial offset of 60 frames the estimated alignment was $j = 0.996 \cdot i + 61.56$ witch leads to an error of $-0.004i + 1.56$ frames, a reasonable estimation and the running time was $127s$ for the point extractions and $783s$ for the correspondence search, in average 200 points from each frame were detected in each frame and $W(F, f)$ took in consideration the 50 best matching points found. The search range estimated by our method was also plotted in the graph note that the correct frame always lies within the search region.

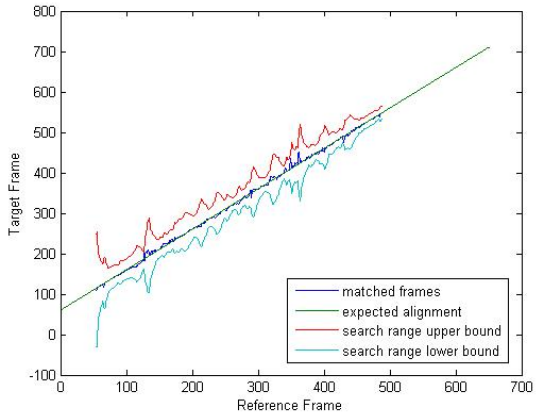

Figure 4. temporal alignment estimation 1.

Figure.4 shows the result of the matching for the same data set that generated Figure.3 by searching frames inside the estimated search region, the estimated alignment in this case was $j = 1.001 \cdot i + 60.2$ witch give us an error of only $0.001i + 0.2$ frames. Also the running time for the correspondence search was $71s$, the parameters used by the estimator were $\gamma = 0.8$, $\delta = 0.6$, $\epsilon = 2$ and $k = 3$

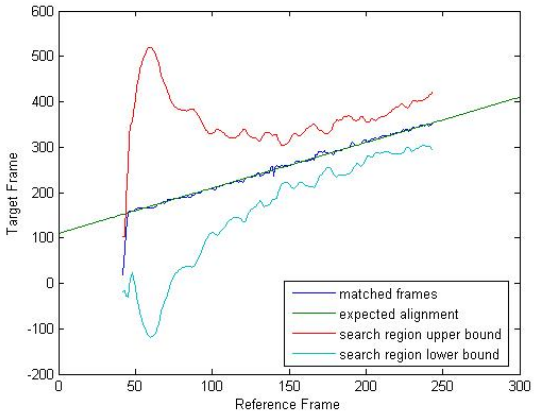

Figure 5. temporal alignment estimation 1.

Figure.5 shows the result of the algorithm applied to an incorrect initial search region. The search region quickly grows in size and then converges back to a small value. the sequences had an offset of 110 an the initial search region estimated an maximum offset of 60 frames. the inclusion of outliers in regression leads an estimation of the alignment as $j = 1.08i + 98.3$ witch give an error of $0.08i - 11.7$ frames by the time the algorithm reaches the end of the sequences.

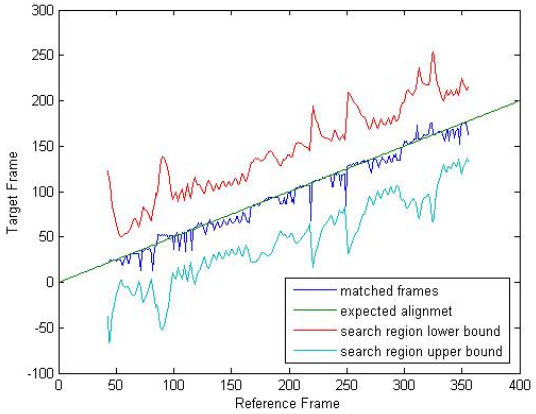

Figure 6. temporal alignment estimation 1.

Figure.6 shows the result of the application of the algorithm to a pair of sequences with differing frame-rates. The frame-rate and 0 initial offset. the frame rate of the target sequence is 2 times the frame rate of the reference sequence.

the estimated offset was $j = 0.49 \cdot -2.3$ and gives an error of $-0.01i - 2.3$ the loss of precision is due to the acceleration of the frame-rate of the target sequence decreases the precision of the method twofold the time for the matching in this data set was of $43s$.

6. Conclusions

The estimator for the synchronization presented in this paper, under the assumption that scene has low depth variation, have been demonstrated to have a good accuracy for determining the temporal alignment, our method for dynamic estimation of the search range for the correspondence, is shown to correctly encompass the correct matching frame for the great majority of the frame search while keeping the search range small, this approach resolves many of the ambiguity issues related with periodic movement described by Reid and Cardeal [12, 3] an at the same time greatly improving execution time. A possible downside of the method is that if we have poor initial estimation for the initial search region, and when dealing with scenes with a high amount of periodic movements the method can get stuck at a local minimum, but if the movement is random it can be expected that the absolute error will become higher until the search region encompasses the correct target frame.

We also improve on the work of Reid in the sense that our space feature detection is automatic and correspondence is based on image data alone. However the assumption that the scene can be approximated with an affine projection is many of times impractical, as future work we aim at developing a method that applies to more general cases.

References

[1] Herbert Bay, Beat Fasel, and Luc Van Gool. Interactive museum guide: Fast and robust recognition of museum objects. In *Int. Workshop on Mobile Vision*, 2006.

[2] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *Ninth European Conference on Computer Vision*, 2006.

[3] Flávio Luis Cardeal. *Spatio-Temporal Alignment of Video Sequences Captured from Multiple Viewpoints*. PhD thesis, Universidade Federal de Minas Gerais, 2005.

[4] Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, pages 381 – 395, 1981.

[5] J. H. Friedman, J. L. Bentley, and R. A. Finkel. An algorithm for finding best matches in logarithmic expected time. *ACM Transactions on Mathematical Software*, 3:209–226, 1977.

[6] Chris Harris and Mike Stephens. A combined corner and edge detector. In *Fourth Alvey VisionConference*, pages 147–151, Manchester, 1988.

[7] Tony Lindeberg. Scale-space theory: A basic tool for analysing structures at different scales. *Journal of Applied Statistics*, pages 224–270, 1994.

[8] David G. Lowe. Distinctive image features from scale-invariant key points. *International Journal of Computer Vision*, 2:91–110, 2004.

[9] Hans Moravec. Rover visual obstacle avoidance. In *International Joint Conference on Artificial Intelligence*, pages 785–790, Vancouver, 1981.

[10] Denis Simakov, Yaron Caspi, and Michal Irani. Feature-based sequence-to-sequence matching. In *Workshop on Vision and Modelling of Dynamic Scenes*, Copenhagen, May 2002.

[11] Carlo Tomasi and Takeo Kanade. Shape and motion from image streams under orthography:a factorization approach. *International Journal of Computer Vision*, 9(2):137–154, 1992.

[12] Phil Tresadern and Ian Reid. Synchronizing image sequences of non-rigid objects. In *British Machine Vision Conference*, Norwich, 2003.

[13] Lior Wolf and Assaf Zomet. Correspondence-free synchronization and reconstruction in a non-rigid scene. In *Workshop on Vision and Modelling of Dynamic Scenes*, Copenhagen, May 2002.