Video Synchronization Based on Displacements of Center of Motion

Raviteja Vemulapalli
Department of Electrical Engineering
Indian Institute of Technology, Madras, India
ravitejav@ieee.org

Luis Salgado
Grupo de Tratamiento de Imágenes
Universidad Politécnica de Madrid, Spain
L.Salgado@gti.ssr.upm.es

***Abstract*—Video synchronization is one of the first steps in most of the multi-camera systems. In this paper we introduce a novel, computationally simple and reliable approach for video synchronization that does not require any pre-computed camera geometries or tracking certain features. We define a feature called Center of Motion (COM) and obtain its trajectories along the temporal axis for both videos. Then the temporal shift is estimated by correlating the trajectories. We follow a subsequence based approach to make our algorithm reliable and assign a confidence measure to the estimated temporal shift. Whenever our algorithm fails to estimate the temporal shift reliably it indicates its inability rather than giving a wrong output. Experimental results demonstrate promising capability of our approach.**

I. Introduction

A multi-camera system that captures the same scene from different viewpoints is commonly used for various applications such as surveillance, multiple view synthesis for object recognition, dynamic scene reconstruction, etc. For such systems it is always important to represent all the videos on a single time scale. If all the cameras are connected to the same external clock signal then they will be synchronized automatically. But when cameras do not have a common clock signal they have to be synchronized using the visual information present in the videos. And for videos that have been already captured synchronizing based on visual information is the only option.

Video synchronization algorithms can be divided into two categories: intensity based and feature based. Feature based methods use tracking and matching of specific features like points on moving objects, object trajectories, etc. whereas intensity based methods[1] use all the pixels and avoid feature tracking and matching. In [2] Caspi and Irani proposed a method based on matching the trajectories of objects. But this method uses geometric correspondences for synchronization and hence a sufficient number of correspondences across images are necessary, which may not be possible always. In [3] Wedge *et al.* also used trajectories of moving objects. In [4] Yan and Pollefeys proposed a method based on correlating distribution of space-time interest points [5] in different videos. In [6] Ushizaki *et al.* proposed a method that uses co-occurrence of appearance changes of objects in motion that are observed from different viewpoints. In this method the spatial integral over the image plane of temporal derivatives of brightness is used as a temporal feature. But space-time interest points and appearance changes of objects in motion are highly view dependent features and hence the methods proposed in [4, 6] have high chances to fail when the two camera views are significantly different. In [7] Whitehead *et al.* discussed different theoretically possible variants of the video synchronization problem and proposed a method that uses tracking of moving objects and pre-computed camera geometries.

In this paper we present a feature based algorithm which does not require any feature tracking or pre-computed camera geometries. We define COM for every frame and obtain its displacements along the videos. Then we correlate the displacement sequences to find the temporal shift. We also follow a subsequence based approach to make our algorithm reliable. The rest of the paper is organized as follows. In section II we discuss our new temporal feature, displacement of COM. We present our method in section III and our experimental results in section IV. We conclude the paper and discuss possible future work in section V.

II. New Feature: Displacement of COM

Center of motion (COM) is a virtual center (like center of mass) of all dynamic pixels (pixels belonging to moving objects). It is given by

$$X_{com} = (\sum x_i)/N \qquad Y_{com} = (\sum y_i)/N$$

where x_i and y_i denote the x(row) and y(column) coordinates of a dynamic pixel and N is the number of such pixels.

A. Finding COM and its Displacement Sequences

To identify the dynamic pixels in a frame we use background subtraction. Various background subtraction techniques from simple to complex ones like frame differencing, adaptive Gaussian mixture model [8], wallflower [9], Bayesian background modeling [10], nonparametric model [11] have been proposed in the past. In this paper we use the simplest technique, frame differencing, in which the previous frame is considered as background model for the current frame, to keep the approach computationally simple. Though this technique results in poor detection of dynamic pixels, it gives a good estimate of COM. Fig. 2(a) shows

978-1-4244-4859-3/09/$25.00 ©2009

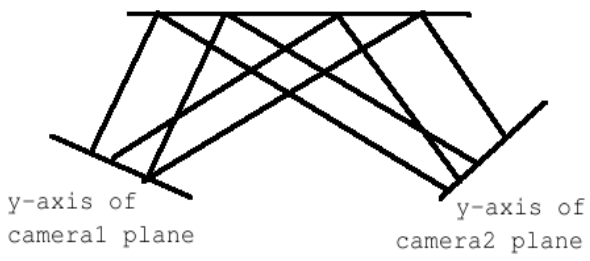

Fig. 1. Problem associated with y-components

Fig. 2. (a) Dynamic pixels identified using frame differencing; (b) COM calculated from the obtained dynamic pixels

the dynamic pixels identified using frame differencing. We can see that this technique suffers from foreground aperture and walking person problems [9]. But it still gives a good estimate of COM (Fig. 2(b)). But if waving tress, illumination changes, etc. are present then more complicated background subtraction algorithms must be used to get a good estimate of the position of COM.

Once the locations of COM in all frames of a video are known, we obtain the sequences of x-displacements and y-displacements of COM for that video. Let $(X_1, Y_1), (X_2, Y_2), (X_3, Y_3), ..., (X_N, Y_N)$ denote the coordinates of COM in different frames of a video. Then the corresponding sequences of x and y displacements are $[(X_2 - X_1), (X_3 - X_2), ..., (X_N - X_{N-1})]$ and $[(Y_2 - Y_1), (Y_3 - Y_2), ..., (Y_N - Y_{N-1})]$.

B. Advantage of Displacement of COM

The approaches followed in [2], [3] and [7] require tracking of moving objects whereas COM does not require any tracking. The temporal features like distribution of space-time interest points [4] and co-occurrence of appearance changes of objects [6] are highly view dependent and hence the corresponding approaches have high chances to fail when the views of the cameras are significantly different. Displacement of COM is also a view dependent feature and hence may look differently from different viewpoints. But displacements observed from different viewpoints can be considered as projections of displacements of a real COM onto different camera planes and hence there is an intrinsic relation between them. So displacement of COM is a much better feature compared to the earlier used ones.

C. X-Component of Displacement of COM

Displacement of COM can have both x and y components. After observing closely we realized that only x-component should be used for correlation. Imagine a person moving towards camera1 and away from camera2. Consider y-components of displacements of COM observed in both cameras. When he is near camera1 the y-displacement in camera1 is much large compared to y-displacement in camera2 and vice versa. So y-displacements in both the cameras are not proportional to each other. Fig. 1 shows this effect roughly. This effect on x-displacements is less and can be neglected when cameras are considerably far from the scene. So we consider only x-component for correlation.

III. Subsequence Based Approach

Our approach is applicable only to videos satisfying the following constraints:

1) The scene should have moving objects such that the COM observed in every video has acyclic x-motion. In most of the cases existence of moving objects would be sufficient to satisfy this constraint.

2) Significant part of the scene with moving objects should be covered by both the cameras simultaneously at same frame rate and cameras should not be too close to the scene.

We first calculate COM for every frame in the videos and find its displacements along both the videos as described in section II. Then by correlating the sequences of x-displacements we estimate the temporal shift. Here we follow a subsequence based approach to make our algorithm more reliable.

Given two videos or image sequences seq1 and seq2, we form two new sequences

Im1 = seq1
Im2 = seq2(d+1 : L-d)

where d is the maximum possible shift and L is the length of the sequences. Now we consider subsequences of Im2. Since there is no specific value for the length(l) of subsequences which is sure to work, we start with subsequences of certain length α. Note that subsequences are chosen such that adjacent elements of a subsequence are also adjacent elements in Im2. Each subsequence is considered as a valid subsequence only if it satisfies the following constraints:

1) The first frame of the subsequence should have moving objects in it.

2) Certain minimum percentage (β) of the frames in the subsequence should have moving COM.

These two constraints ensure that only those parts of the sequence that have significant motion of COM are being used. If the number of valid subsequences is greater than certain number m, then all the valid subsequences are correlated with Im1. Each subsequence indicates a value for the temporal shift which corresponds to the correlation peak. Among all indicated shift values the one that is indicated by maximum number of subsequences is considered as the true shift if more than certain percentage (γ) of all valid subsequences indicate this particular shift value. In all other cases we increase the length l of subsequences by Δl and continue with the same procedure until either the temporal shift is found or the length l reaches a limit t. Reaching the limit t indicates the inability of the approach to find the correct shift. Our algorithm can be summarized as follows:

****

$l = \alpha$

While $(l < t)$

Consider all possible subsequences of Im2 with length l and check for their validity.

If (number of valid subsequences $> m$)

then correlate each subsequence with Im1 to get corresponding temporal shift value.

Consider the shift value s that is indicated by maximum number of subsequences.

If (% of subsequences indicating $s > \gamma$)

then s is the output.

EXIT

end

end

$l = l + \Delta l$

end

Indication of inability of the algorithm.

****

By following a subsequence based approach only when more than certain percentage (γ) of all valid subsequences give the same shift value, we consider it as the true shift. This is equivalent to associating a confidence measure to the estimated shift. Gamma is a measure of the confidence and we can say that the value of temporal shift is say n with a confidence measure of γ. If the algorithm fails to find the shift with a confidence measure of γ, the value of γ may be decreased to get a shift with lower confidence measure. The value of Δl affects the computation time. Ideally we should use $\Delta l = 1$ so that we check for every value of l but this increases the computation time. By choosing a higher value for Δl we can reduce the computation time. But if the value is too high we may completely skip the range of values of l for which the temporal shift can be estimated with high confidence measure.

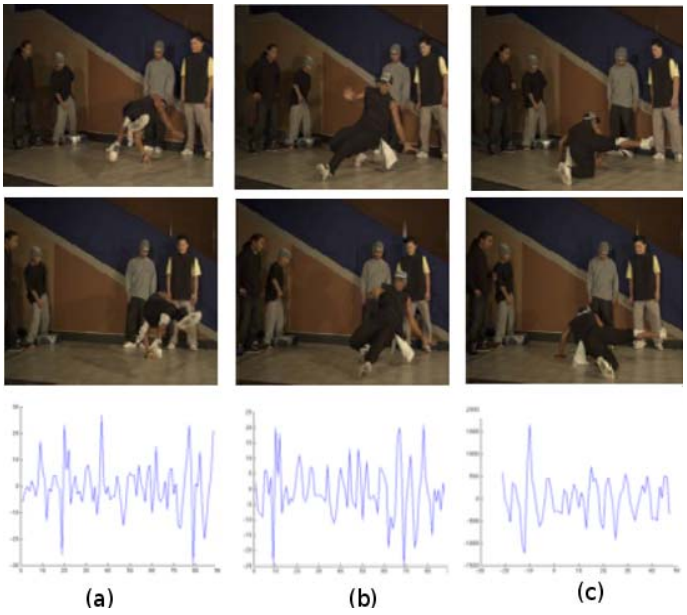

Fig. 3. The first and second rows show some frames of Bdancer1 and Bdancer2 videos taken at same instances of time; (a) x-displacements along the Bdancer1 video; (b) x-displacements along the Bdancer2 video; (c) correlation for a subsequence of Bdancer2 with Bdancer1.

IV. Experimental Results

We tested our approach of different sets of videos and were able to estimate the temporal shift exactly.

Values of the parameters used:
$\alpha = 20,\ \beta = 75,\ \Delta l = 10,\ m = 10,\ \gamma = 75,\ t = \text{length}(Im1) - m.$

A. Breakdancer (Bdancer)

Fig. 3 shows some frames of the Bdancer1 and Bdancer2 videos of Breakdancer set. There is a temporal shift of 10 frames in between them with camera1 starting first. The two camera views are significantly different in this case. Observe the right most person. He can be seen fully in the first video, whereas he is covered by the dancer to some extent in the second video. Figures 3(a) and 3(b) show the x-displacement sequences of both the videos and Fig. 3(c) shows the correlation between first sequence and a subsequence ($l = 20$) of second sequence with peak at -10. Our subsequence based approach estimated the temporal shift as 10 with a confidence measure of 85% using subsequences of length 20.

B. Ballroom (Broom)

Fig. 4 shows some frames of the Broom1 and Broom2 videos of Ballroom set. There is a temporal shift of 30 frames in between them with camera1 starting first. The two camera views are significantly different in this case also. People at the left and right ends can be seen only in one of the cameras. We can see three dancing couples in the first frame of top

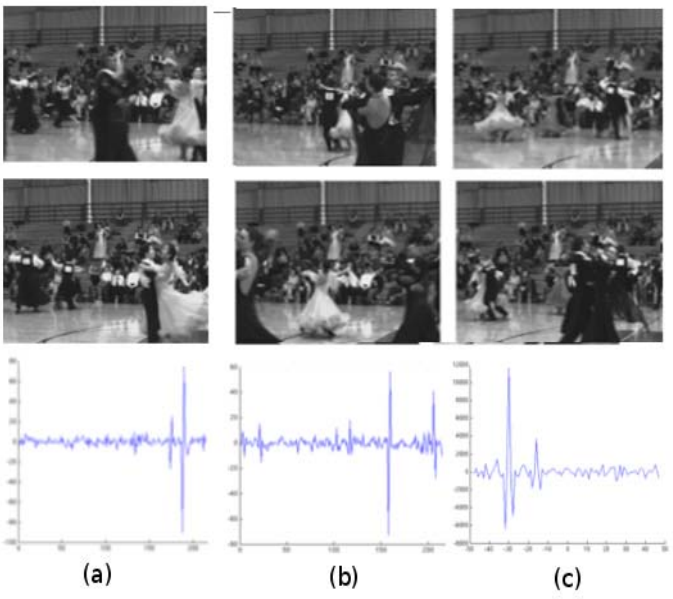

Fig. 4. The first and second rows show some frames of Broom1 and Broom2 videos taken at same instances of time; (a) x-displacements along the Broom1 video; (b) x-displacements along the Broom2 video; (c) correlation for a subsequence of Broom2 with Broom1

row of Fig. 4 and only two in the first frame of second row with only one couple being common, though both frames are captured at the same instant of time. Figures 4(a) and 4(b) show the x-displacement sequences of both the videos and Fig. 4(c) shows the correlation between first sequence and a subsequence ($l = 120$) of second sequence with peak at -30. Our subsequence based approach estimated the temporal shift as 30 with a confidence measure of 75% using subsequences of length 120.

V. Conclusion and Future Work

We have presented a novel, computationally simple and reliable method for video synchronization based on correlation of x-displacements of COM observed in the videos. We followed a subsequence based approach to increase the reliability of our method. We also associated the estimated shift with a confidence measure. We tested our algorithm on different sets of videos and obtained promising results. Advantages of our algorithm compared to earlier proposed ones are reliability, usage of a much better temporal feature and simplicity in terms of computation.

In this paper we have used the simplest background modeling technique (frame differencing) but using better techniques like mixture of Gaussians or nonparametric model may result in much better estimation of position and displacements of COM and hence the temporal shift. We also assumed that both the cameras have same and fixed frame rate. In case of different or varying frame rates interpolation techniques should be used. One can also work towards making this offline algorithm online.

Acknowledgment

We would like to thank the Ministerio de Ciencia e Innovación of the Spanish Government for supporting this work under project TEC2007-67764 (SmartVision), and the Comunidad de Madrid for supporting us under project S-0505/TIC-0223 (Pro-Multidis).

References

[1] Y. Caspi and M. Irani, "Spatio-Temporal Alignment of Sequences", in *IEEE Transactions on PAMI, 24(11):14091424, November 2002*.

[2] D. S. Y. Caspi and M. Irani, "Feature-based sequence-to sequence matching" in *Workshop on Vision and Modeling of Dynamic Scenes*, 2002.

[3] Daniel Wedge, Peter Kovesi and Du Huynh, "Trajectory Based Video Sequence Synchronization", in *Proceedings of the Digital Imaging Computing: Techniques and Applications (DICTA 2005)*.

[4] J. Yan and M. Pollefeys, "Video synchronization via space-time interest point distribution" in *Advanced Concepts for Intelligent Vision Systems*, 2004.

[5] Laptex and T. Lindeberg, "Space-time interest points" in *Proceedings of IEEE International Conference on Computer Vision*, pages 432-439, 2003.

[6] Manabu Ushizaki, Takayuki Okatani and Kochiro Deguchi, "Video Synchronization Based on Co-occurrence of Appearance Changes in Video Sequence" in *18th International Conference on Pattern Recognition (ICPR'06)*.

[7] Anthony Whitehead, Robert Laganiere and Prosenjit Bose, "Temporal Synchronization of Video Sequences in Theory and in Practice", in *Proceedings of the IEEE Workshop on Motion and Video Computing (WACV/MOTION05)*.

[8] C. Stauffer and E. Grimson, "Adaptive background mixture models for real-time tracking" in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Fort Collins, CO, volume II, 1999, pp. 246-252.

[9] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers, "Wallflower: Principles and practice of background maintenance" in *Proceedings of 7th International Conference on Computer Vision*, Kerkyra, Greece, 1999, pp. 255-261.

[10] F. Porikli and O. Tuzel, "Bayesian background modeling for foreground detection" in *Proceedings of the 3rd ACM International Workshop on Video Surveillance & Sensor Networks (VSSN '05)*, pp. 55-58, Singapore, November 2005.

[11] A. Elgammal, D. Harwood and L. Davis, "Nonparametric model for background subtraction", in *European conference on Computer Vision*, pages 751-767, Dublin, Ireland, May 2000.