SYNCHRONIZATION FOR MULTI-PERSPECTIVE VIDEOS IN THE WILD

Junwei Liang, Poyao Huang, Jia Chen and Alexander Hauptmann

School of Computer Science, Carnegie Mellon University, Pittsburgh, USA

ABSTRACT

In the era of social media, a large number of user-generated videos are uploaded to the Internet every day, capturing events all over the world. Reconstructing the event truth based on information mined from these videos has been an emerging challenging task. Temporal alignment of videos "in the wild" which capture different moments at different positions with different perspectives is the critical step. In this paper, we propose a hierarchical approach to synchronize videos. Our system utilizes clustered audio-signatures to align video pairs. Global alignment for all videos is then achieved via forming alignable video groups with self-paced learning. Experiments on the Boston Marathon dataset show that the proposed method achieves excellent precision and robustness.

Index Terms— Event Reconstruction, Video synchronization, Video Analysis, Audio Signal Processing

1. INTRODUCTION

With the growing world-wide connectivity and popularity of camera-embedded smart devices, events around the world can now be captured and rapidly shared via social media. When an event happens, especially those with a large crowd of people, different videos would record different moments of the same event at different positions from different perspectives. For example, New Year's Eve at NYC, Carnival in Brazil, and Boston Marathon bombing all have hundreds or even thousands of attendees upload videos of the event. The collection of these user-generated recordings not only enable new applications such as free-view video [1] and 3D-reconstruction [2]; but it may also help our society achieve a more unbiased understanding of the event truth [3]. Such information would particularly important for conflict or violence events, in which the truth of what happened is critical to the general public and for law enforcement to take action. Unlike videos captured by fixed, calibrated surveillance cameras, consumer videos are captured "in the wild" (i.e., at varying time, location, perspectives, with different devices such as smart phones, watches or camcorders.) These videos are noisy and sometimes with low quality. In an unexpected violent event, people are often scared and the videos may be too blurry or shaky to see. Useful information about the event may spread across different time segments of different videos. Therefore, To properly process and analyze a video collection, one main problem that must be solved is to synchronize these video and put them into a global timeline.

Conventional approaches to synchronize two videos relies on evaluation of similarity using visual features. For example, [4] generated self-similarity matrix and [5] relied on a trajectory feature set. However, these methods are only applicable when the objects are within line-of-sight to both cameras, which is an infeasible requirement for videos captured at divergent locations and perspectives.

The broadcasting nature of sound wave makes audio features better candidates. First introduced in [6], recent works has shown great promise using audio fingerprinting techniques [7, 8, 9]. These works suggested different landmark feature extraction strategies that encode the audio into a representation for matching. In [8], Schweiger *et al.* utilized RANSAC-like methods to further improve the robustness. However, these audio fingerprinting approaches are more feasible for indoor scenario with only one high signal-to-noise ratio source. Synchronization under the in-the-wild paradigm is particularly challenging since: 1) The cameras may locate far apart in space and be disjoint in time, and hence less likely sharing the same audio environment. 2) Depending on the locations of cameras and sound sources, cameras may observe completely different audio patterns due to physical broadcasting of sound waves (e.g. one camera close to the sound source while the others are not) , which contradicts the assumption in audio fingerprinting approaches. 3) Additionally, noisy environment and differences in hardware shared by users impair the recording quality and lead to inconsistencies for pairwise matching.

For multiple videos matching, most previous methods share a bottom-up framework: First matching single pairs and then hierarchically merging into clusters until a global alignment was reached. For example, in [10], pairwise correlations are fist calculated and then iteratively merged into larger groups via reversed-indexing correlation evaluation. Similarly, [11] extended the work in [7] by applying a clustering techniques to the pairwise match scores and grouping video into coherent scenes. In [12], Kammer *et al.* proposed using a minimum spanning tree to achieve global alignment. However, all these grouping approaches are sensitive to outliers which are common with multiple sources and noisy environment. Also, erroneous grouping decision results propagate through iterative grouping.

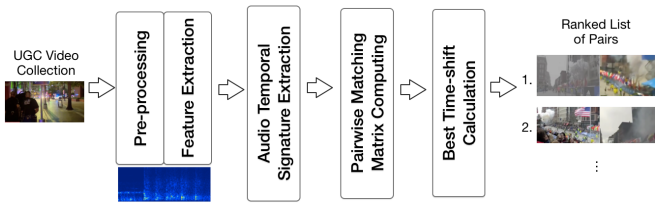

Fig. 1. **Pairwise Video Synchronization**

In contrast to previous works, we consider the video in-the-wild paradigm. We create clustered audio-signatures for pairwise matching which is more robust to noise. To alleviate the affect of outliers, we provide a general formulation treating global synchronization problem as a regularized self-paced learning problem [13]. This approach provides additional robustness to inevitable outliers by dynamically pruning and grouping in each optimization iteration.

Our work has three major contributions. First, we propose a robust pairwise video synchronization method. Second, we formulate and solve a general global alignment problem. Third, we create and provide a Boston marathon bombing dataset to encourage future study for synchronizing video in the wild.

2. VIDEO SYNCHRONIZATION SYSTEM

Our video synchronization system operates in two stages. The first stage is pairwise video synchronization. Our system finds the best synchronization for each video to the others. Then in the second stage, our system find the best global synchronization among all the videos based on the pairwise alignment.

2.1. Pairwise Video Synchronization

Our pairwise synchronization system consists of four components as shown in Figure 1.

Pre-processing and Feature Extraction. Since many user-generated videos are edited before uploading to social media, our system first chunk videos into time-continuous segments based on the shot boundary detection. In an unexpected violent event, people are scared and the video quality may be low and too blurry to see any useful visual evidence. Therefore in this system we focus on the audio modality for synchronization. We extract low-level audio features from the audios, leaving out the videos with no sound.

Audio Temporal Signature Extraction. Most user-generate videos are very noisy. In order to extract useful audio signature at each given time frame, our system first conducts an unsupervised clustering to get an audio signature dictionary, and then assigns each time frame of the video segments to the closest k centers.

Pariwise Matching Matrix Computing. After assigning each time frame, our system computes the matching matrix m_{ij} for each video segment pairs v_i and v_j. Each element of the matching matrix for each pair of video segments is calculated by a function $m_{ij}^{st} = p(v_i^s, v_j^t)$:

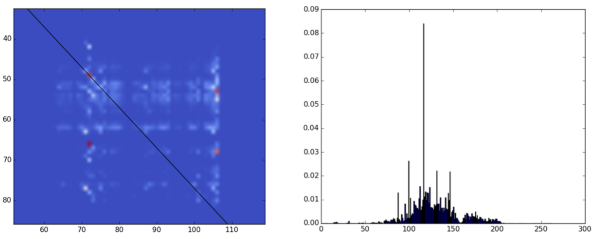

Fig. 2. **Visualization of Pairwise Matching**

$$p(v_i^s, v_j^t) = \sum_{\alpha=1}^{k}\sum_{\beta=1}^{k}\frac{1}{\alpha}\frac{1}{\beta}d(v_i^{s\alpha}, v_j^{t\beta})$$
$$d(v_i^{s\alpha}, v_j^{t\beta}) = \begin{cases} 1 & v_i^{s\alpha} = v_j^{t\beta} \\ 0 & v_i^{s\alpha} \neq v_j^{t\beta} \end{cases}. \quad (1)$$

where $s \in [1, c_i]$ and $t \in [1, c_j]$ give the time frame number of video segment v_i and v_j. v_i^s is a vector of center numbers for the sth frame of v_i. For each time frame in the video segment pair, our system checks through the center number vectors and adds a matching score p to m_{ij}^{st} if the center numbers are the same. A visualization of the matching matrix m_{ij} is shown on the left in Figure 2, where blue color area indicates that no value in both of the center number vectors of the two video segments at a given time frame is matched, while the color range from white to red shows the amount of the matched center numbers.

Best Time-shift Calculation. After we compute the matching matrix for each pair of video segments, the final step is to find the probabilistic scores of how confidence each pair of video segments is indeed aligned and the best time-shift of the alignment between each pair. Each diagonal of the matching matrix corresponds to a time-shift setting of the video segment pair. To find the best time-shift between the video segment pair, we sum up each diagonal's value in the matrix and put them into a matching histogram as shown on the right in Figure 2, where the x axis corresponds to each time-shift and the y axis corresponds to the summed values. Moreover, to encourage continuous signature matching, we add a multiplier for continuous non-zero value sequence when summing up along the diagonal. We get the best time-shift from the diagonal with the highest value as shown in the two graphs in Figure 2, where a black diagonal line that corresponds to the peak in the matching histogram is drawn in the matching matrix. Finally, the probabilistic score $s_o(i, j)$ of local alignment between video segment v_i and v_j is calculated by the average of the maximum summed diagonal value and how dominant the peak is.

2.2. Global Video Synchronization

Let $\mathbf{A} \in \mathbb{R}^{N\times N}$ denotes the global alignment matrix where N is the number of video segments, $A_{ij} \in [0, 1]$ indicates whether video segment i and j are globally aligned. Let $f(i, j)$ and $g(i, j, \mathbf{A})$ denote the utility and penalty function

respectively. The cost function for global video synchronization is:

$$\mathbb{E}(A,\lambda)=\sum_{i=1}^{N}\sum_{j=1}^{N}A_{ij}g(i,j,\mathbf{A})-\lambda\sum_{i=1}^{N}\sum_{j=1}^{N}A_{ij}f(i,j) \quad (2)$$

where λ can be viewed as a parameter controlling the learning rate when minimizing the cost function with the negative l_1-norm regularizer $-\|\mathbf{v}\| = -\sum_{i=1}^{N}\sum_{j=1}^{N}A_{ij}f(i,j)$ with $A_{ij}f(i,j) > 0$. The utility function $f(i,j)$ models the reward when (i,j) are selected to be aligned. For the most possible time-shift $o_{max} = max^{-1}s_o(i,j)$ between i and j, a simplest reward function is the bounded pairwise alignment score: $f(i,j) = min(R_{max}, s_{o_{max}}(i,j))$, where R_{max} is the maximum reward.

In Eq. 2, $g(i,j,\mathbf{A})$ penalizes triplet contradiction in temporal alignment. Conceptually, given the temporal locations of aligned segments i,j and their time shift o_{ij}. Contraction happens if the locations of video $k \in \mathbf{A}$ predicted by o_{ik} and o_{jk} are not the same, which will be penalized by the squared loss function $l(.)$. An example design of $g(i,j,\mathbf{A})$ can be:

$$\sum_{\substack{k\in\mathbf{A}\\ k\neq i\vee j}} l(max_{o_{jk}}^{-1}s_o(j,k)-max_{o_{ik}}^{-1}s_o(i,k)-max_{o_{ij}}^{-1}s_o(i,j)) \quad (3)$$

The complexity to search the solution for Eq. 2 is $O(2^{N^2})$. To solve the optimization problem efficiently, one feasible approach is the alternative search strategy as in [14]. We separate the alignment matrix into an internal latent alignment matrix $\mathbf{A}^{(in)}$ as the parameters for penalty function and another external alignment matrix $\mathbf{A}$ as the optimization target. In each optimization iteration, we fixed one of them and update the other until convergence ($\|\mathbf{A}^{(in)}-\mathbf{A}\|<\epsilon$). The optimization problem become a iteratively solvable self-paced learning problem [15] in the form of:

$$\min_{\mathbf{A},\mathbf{A}^{(in)}}\mathbb{E}(\mathbf{A}^{(in)},\mathbf{A},\lambda) = \sum_{i=1}^{N}\sum_{j=1}^{N}A_{ij}g(i,j,\mathbf{A}^{(in)})-\lambda\sum_{i=1}^{N}\sum_{j=1}^{N}A_{ij}f(i,j) \quad (4)$$

Note that with the fixed $\mathbf{A}^{(in)}$, the optimum $\mathbf{A}^*$ can be can be easily calculated by:

$$A_{ij}^{(t)}=\begin{cases}1 & \lambda f(i,j)-g(i,j,\mathbf{A}^{(in)})>0\\ 0 & \text{otherwise}\end{cases} \quad (5)$$

A similar form holds while updating $\mathbf{A}^{(in)}$ with fixed $\mathbf{A}$. Therefore, with the pairwise (local) alignments $s_o(i,j)$, a feasible global alignment for all videos can be achieved with the following algorithm.

3. EXPERIMENTS

In this section, we show our system's performance in both local (pairwise) and global alignment experiments on the Boston Marathon Dataset.

Algorithm 1 Global Alignment with Self-paced Learning

Input: The pairwise video matching score
Output: The alignment matrix A

```
Initialize A^(in) = I, A = 0
repeat
    Update A = argmin_A E(A, A^(in), λ)
    Update A^(in) = argmin_{A^(in)} E(A, A^(in), λ^(in))
until converge
```

3.1. Dataset Description

We collect a real-world event synchronization dataset "Boston Marathon 2013", in which two consecutive explosions happened on the sidewalk near the finish line of a traditional city marathon in Boston in 2013. This event received widespread international media attention and synchronizing videos of such event is very useful to event reconstruction and analysis. We constructed the queries "Boston marathon 2013 explosion", "Boston marathon 2013 bomb", "Boston marathon 2013 after explosion", "Boston marathon 2013 after bomb" to crawl videos from Youtube and Dailymotion, two of the most popular video sharing websites. We crawled the top 500 search results from each query on Youtube, and all the search results from Dailymotion. We manually refined the relevance of all the crawled search results by removing irrelevant videos, resulting in 347 relevant videos. The relevant video is defined as on-site videos of the "Boston Marathon 2013" event. The dataset will be released to the public.

3.2. Local Alignment

3.2.1. Evaluation Metric

We introduce two granularities, video and frame, in evaluation for the local alignment. The major challenge in evaluation is that it is nearly impossible to manually label all the synchronization ground truth on pairs as the cost is too high.

Video granularity: one pair is considered as correct if the clips have an intersection on the timeline. We manually verify the top predicted pairs rather than label all the ground truth. The evaluation metric is average precision (AP). To be specific, we adapt AP to truncated AP (t-AP) to handle the incomplete ground truth issue. We verify the output from high score to lower score until the precision on the verified ones drops to 10%. We also report precision at top 100 video pairs as an additional metric.

Frame granularity: we calculate the RMSE of the time-shift (tf) only on the verified correct pairs in clip granualarity.

$$RMSE=\Big(\sum_{p\in\text{correct pairs}}(tf-\widehat{tf})^2\Big)^{1/2} \quad (6)$$

, where tf is the ground truth time-shift, and $\widehat{tf}$ is the predicted time-shift.

Our verification interface is shown in Figure 3, where each video pair is presented in parallel to the annotator and

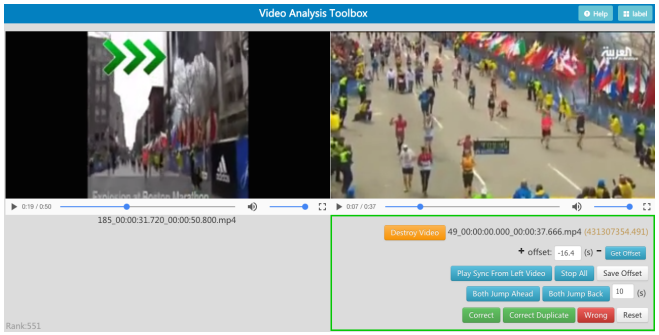

Fig. 3. **Manual Verification Interface**

the audio of the two videos are played separately through the left and right channel. Noted the video pairs with identical sound (e.g. with exact same music) are labeled as correct.

3.2.2. Local Alignment Experiments

In the local alignment experiments, we compare the performance of three common low-level audio features and the traditional audio fingerprinting method [6]. The low-level features includes: 1) MFCC: The MFCC features (13MFCC + 13delta +13ddelta) are computed every 25ms with 10ms shift. 2) Short-Time-Fourier-Transform (STFT): The STFT features are computed every 16ms with 8ms shift. 3) Chroma: 12-dimensional chroma features are calculated every 64ms with 10ms shift. The MFCC and Chroma features are calculated using the openSmile toolkit [16]. We implement the traditional fingerprinting method (Audio Fingerprint) using the dejavu package[1]. Experimental results are shown in Table 1.

Comparing the traditional audio fingerprint method to our system, all three features show significant improvement both on video-level and frame-level evaluations. As discussed in Section 1, the user-generated videos are very noisy. While the audio fingerprint method has been proven useful for music matching, it fails to find mutual acoustic evidence among noisy videos. Comparing the three low-level features, chroma-based system outperforms the others by a significant margin, suggesting that chroma features are better for video synchronization.

Table 1. Results of local video synchronization

Mehods	p@100	t-AP	RMSE
Audio Fingerprint	0.290	0.304	40.375
MFCC-Sync	0.730	0.690	0.575
STFT-Sync	0.740	0.754	4.851
Chroma-Sync	0.890	0.921	0.400

3.3. Global Alignment Experiments

In global alignment experiments, we sub-sampled 72 video segments and manually labeled the global time. 40 segments are alignable while the rest 32 are standalone. An alignment is correct if the time difference to the ground truth is less than 100 ms. We use the best local synchronization result with chroma features. The baseline methods build the global alignment considering top k uni/bi-directional matches and merge them into aligned groups. Video i and j are aligned if one(uni-)/both(bi-) of them are in the top k ranked list of $s_o(i, j)$. The evaluation metrics are precision, recall and F1 score.

Table 2. Results of global video synchronization

	Precision	Recall	F1
uni-match top 5	0.518	0.126	0.203
uni-match top 15	0.481	0.348	0.406
bi-match top 5	0.566	0.176	0.269
bi-match top 15	0.526	0.345	0.417
proposed	**0.763**	**0.374**	**0.502**

Table 2 summarizes the global synchronization result. The proposed methods achieved the best result since it minimizes triplet contradictions within the aligned set by dynamically merging and pruning alignment matrix ($\mathbf{A}$) in each iteration. For baseline methods, bi-directional matching delivers better precision than uni-directional matching. However, from all the approaches, an improvement in recall is still required for synchronizing video in the wild. The error analysis reveals that small groups are formed but the link between groups are lost. Figure 4 shows the interface of our system that visualizes the global alignment results. The complete system will be released for other real world event analysis.

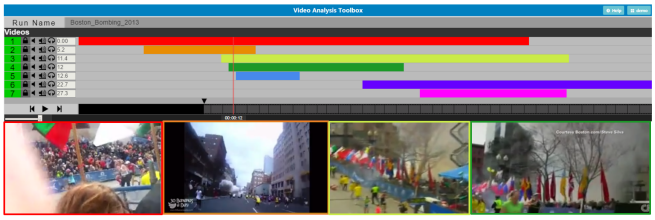

Fig. 4. Interface of our system and a global alignment result for the bombing event in Boston Marathon. Each color bar is associated with the video with the same frame color.

4. CONCLUSION

We presented a video synchronization system which aligns multiple videos in the wild. For pairwise alignment, a robust feature with clustered audio-signatures feasible is proposed for noisy environment with multiple sound sources. Global video synchronization is achieved via self-paced learning which is robust against outliers in pairwise alignment. We also established the Boston Marathon Dataset for studies in event reconstruction and video synchronization.

[1] https://github.com/worldveil/dejavu

5. REFERENCES

[1] Alvaro Collet, Ming Chuang, Pat Sweeney, Don Gillett, Dennis Evseev, David Calabrese, Hugues Hoppe, Adam Kirk, and Steve Sullivan, "High-quality streamable free-viewpoint video," *ACM Trans. Graph.*, vol. 34, no. 4, pp. 69:1–69:13, July 2015.

[2] Yiwei Zhang, Graham M Gibson, Rebecca Hay, Richard W Bowman, Miles J Padgett, and Matthew P Edgar, "A fast 3d reconstruction system with a low-cost camera accessory," *Scientific reports*, vol. 5, 2015.

[3] Jay D Aronson, Shicheng Xu, and Alex Hauptmann, "Video analytics for conflict monitoring and human rights documentation," 2015.

[4] Emilie Dexter, Patrick Pérez, and Ivan Laptev, "Multi-view synchronization of human actions and dynamic scenes.," in *Proceedings of the British Machine Vision Conference*, 2009.

[5] C. Lu and M. Mandal, "An efficient technique for motion-based view-variant video sequences synchronization," in *2011 IEEE International Conference on Multimedia and Expo*, July 2011, pp. 1–6.

[6] Jaap Haitsma and Ton Kalker, "A highly robust audio fingerprinting system with an efficient search strategy," *Journal of New Music Research*, vol. 32, no. 2, pp. 211–221, 2003.

[7] Lyndon Kennedy and Mor Naaman, "Less talk, more rock: automated organization of community-contributed collections of concert videos," in *Proceedings of the 18th international conference on World wide web*. ACM, 2009, pp. 311–320.

[8] Florian Schweiger, Georg Schroth, Michael Eichhorn, Eckehard Steinbach, and Michael Fahrmair, "Consensus-based cross-correlation," in *Proceedings of the 19th ACM International Conference on Multimedia*, New York, NY, USA, 2011, MM '11, pp. 1289–1292, ACM.

[9] Prarthana Shrstha, Mauro Barbieri, and Hans Weda, "Synchronization of multi-camera video recordings based on audio," in *Proceedings of the 15th ACM International Conference on Multimedia*, New York, NY, USA, 2007, MM '07, pp. 545–548, ACM.

[10] N. J. Bryan, P. Smaragdis, and G. J. Mysore, "Clustering and synchronizing multi-camera video via landmark cross-correlation," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2012, pp. 2389–2392.

[11] Kai Su, Mor Naaman, Avadhut Gurjar, Mohsin Patel, and Daniel PW Ellis, "Making a scene: alignment of complete sets of clips based on pairwise audio match," in *Proceedings of the 2nd ACM International Conference on Multimedia Retrieval*. ACM, 2012, p. 26.

[12] Julius Kammerl, Neil Birkbeck, Sasi Inguva, Damien Kelly, Andrew J Crawford, Hugh Denman, Anil Kokaram, and Caroline Pantofaru, "Temporal synchronization of multiple audio signals," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 4603–4607.

[13] M Pawan Kumar, Benjamin Packer, and Daphne Koller, "Self-paced learning for latent variable models," in *NIPS*, 2010.

[14] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston, "Curriculum learning," in *Proceedings of the 26th annual international conference on machine learning*. ACM, 2009, pp. 41–48.

[15] M Pawan Kumar, Benjamin Packer, and Daphne Koller, "Self-paced learning for latent variable models," in *Advances in Neural Information Processing Systems*, 2010, pp. 1189–1197.

[16] Florian Eyben, Martin Wöllmer, and Björn Schuller, "Opensmile: the munich versatile and fast open-source audio feature extractor," in *Proceedings of the 18th ACM international conference on Multimedia*. ACM, 2010, pp. 1459–1462.