Multimodal Alignment of Videos

Research Summary for the Doctoral Symposium

Mario Guggenberger
Institute of Information Technology
Alpen-Adria-Universität Klagenfurt
9020 Klagenfurt am Wörthersee, Austria
mg@itec.aau.at

ABSTRACT

Most multimedia synchronization methods developed in the past are unimodal and consider only the audio data or the video data. Just recently, methods started to emerge that embrace multimodality by utilizing both audio and video processing to improve synchronization results. Although promising, their results are still not sufficient for fully automatic synchronization of recordings from heterogeneous sources. Video processing is also often too expensive to be used on large corpora of recordings, e.g. as they are commonly produced by crowds at social events. In my doctoral thesis, I will try to develop synchronization methods further by (a) examining fundamental problems that are usually ignored by lab-developed methods and therefore compromising real-world applications, (b) creating a publicly available synchronization-method benchmarking dataset, and (c) developing a low-level video feature based synchronization method with a computational complexity not higher than current state of the art audio-based methods.

Categories and Subject Descriptors

I.4 [**Image Processing and Computer Vision**]: Miscellaneous; H.5.5 [**Information Interfaces and Presentation**]: Sound and Music Computing—*Signal analysis, synthesis, and processing*

Keywords

Audio, video, features, time drift, synchronization, multimodality, crowd

1. INTRODUCTION

Even though much effort has been put into the domain of multimedia video processing during the last decades, there are still many unsolved or just partially solved problems left to be explored. One of those problems is the automatic synchronization of continuous single-shot videos. I consider videos that have been recorded at the same place (within a restricted and semantically meaningful area, e.g. the area in front of a stage) and the same time (within a restricted time frame, e.g. during the same concert or theatrical performance). Although these videos depict the same visual 3-dimensional scene mapped to a sequence of 2-dimensional image frames, their content often differs in terms of perspective, movement, recording parameters, frame rate, data format, or resolution. These differences result from the usage of different types of cameras, experience of the operators and intended purposes of recording. Synchronizing all those recordings from an event opens up various interesting use cases like detecting key moments by looking at the frequency of concurrent recordings, temporal stitching of clips to get a complete and continuous coverage of a whole event, creating vivid videos by switching between different perspectives or showing different shots side-by-side, improving presentation quality by picking the best audio and video tracks from concurrent recordings, or reconstructing 3D scenes from recordings of different angles. A popular use case for synchronization are musical or theatrical stage performances where usually many people capture video clips with ubiquitous devices like cameras and smartphones, and newly emerging devices like action cameras and video recording glasses might promote this use case even more. Another important use case are coordinated amateur productions involving multiple consumer recording devices. Although the synchronization of temporally and spatially overlapping videos can be done manually, it is a very tedious and time-consuming task waiting to be solved or at least supported by automatic methods. Currently published methods are still very limited and often not prepared for real-world use.

MM'14, November 3–7, 2014, Orlando, FL, USA.

http://dx.doi.org/10.1145/2647868.2654862 .

2. GOALS

The purpose of my dissertation is to improve the current state of the art in synchronization of continuous single-shot videos towards a precision that mitigates all time-offset distractions and other irregularities that spoil the feeling of synchrony, and to proceed from laboratory concepts towards real-world use. I want to investigate and find methods to achieve precise temporal synchronization of videos without explicitly examining their semantics and without complex high-level processing. The ultimate goal is to reach a level at which the synchronization is so accurate, that a human does not notice any unexpected or negative effects when watching synchronized videos side-by-side and hearing a mix of all the videos' audio signals. While this is not a use case itself, it guarantees a level of synchronization quality that other use cases can be built upon. Reaching this level might be a long

way, but my work will set the first steps towards this direction and towards problems that have not been considered yet. I will specifically focus on the following:

1. **Time drift.** All recording devices have an inherent error of time, which results from inaccuracies and instabilities in hardware crystal oscillators that coordinate the sampling and frame rates. This drift is individual in each device, and a deviation from the nominal sampling rate leads to changes in the recording speed. This makes recordings from different devices incompatible, because their recordings basically exist on different time bases, making parallel playback impossible. The drift is not always noticeable as it depends on different parameters, e.g. the relative drift between devices used in a recording session, and the lengths of the individual recordings. Professional studios and production companies use specialized equipment fed by a common clock signal to avoid the problem, but such devices are usually too expensive for amateurs, and not an option for crowds. This drift needs to be detected and avoided or removed to get a constant synchronization accuracy over time. Just calculating the offset between two videos [6, 9] is insufficient because it aligns them only at a single point and synchronization is lost with increasing distance to that point.
2. **Benchmarking.** Although many synchronization methods exist in the literature, they cannot be objectively compared because their evaluation is usually carried out on custom datasets, which are sometimes very far from a real-world use case. An accurate ground truth for a big dataset would allow a comparison in terms of computational complexity, spacial complexity, synchronization rate, and synchronization accuracy.
3. **Audio/video offsets.** Offsets between the audio and video track of a recording are either introduced by the system, e.g. the encoding pipeline, or by the difference between the speed of sound and light. Recording a scene from up-front and 300 meters distance yields an audio delay of one second between the two recordings at recording time. Audio-based synchronization of the two recordings therefore leads to an offset of one second between the corresponding video tracks, leading to undesired effects when displaying them side by side or switching between them. This means that a good synchronization result requires additional multimodal processing to remove these offsets.
4. **Pure video synchronization.** Purely video-based synchronization methods usually demand complex high-level processing, e.g. feature trajectories [8], or are tied to special use cases, e.g. still camera flashes [9]. I anticipate a video-based algorithm for general use with lower computational complexity compared to currently published methods.

3. RELATED WORK

The related work on this topic is manifold, spanning from the basics of low and high level audio and video processing, fingerprinting, indexing, feature (series) correlation, over to audio, video and multimodal synchronization methods, up to applications like recording quality assessment, automatic (a.k.a. virtual) directing, summary generation, 3D scene reconstruction, and forensics.

Automatic temporal synchronization methods can generally be divided into three types: audio-based [5], video-based [8], and multimodal [6, 9], which is usually a combination of audio- and video-based methods. They are often built upon fingerprinting algorithms and cross-correlation of low-level features or events (e.g. audio frequency or video motion onsets). The prevalent use of audio data as a synchronization reference is a reasonable choice due to the usually omnidirectional pickup patterns of microphones, compared to the strong directionality of video frames. While many methods have their basic building blocks in common, the actual implementations vary greatly, and their evaluations vary even more – they are usually carried out on custom datasets which makes a comparison in terms of runtime, memory consumption, and particularly the synchronization rate and quality, impossible. If datasets are too small, they might (un)intentionally mask problems of complexity. If clips are too short or taken from homogeneous sources, they might mask drift. If the media quality of clips is too high or recorded in lab settings, they might mask low robustness. There are also methods that impose constraints which I want to avoid, because they cannot be used in the uncontrolled domain of user generated recordings. For example, crowd sourced videos cannot be synchronized if there are hard constraints that demand stationary cameras, or manual configuration of cameras. Soft constraints can be fulfilled by post processing, e.g. identical sampling rates can be achieved by resampling. A recent but not exhaustive overview of multi-camera synchronization methods can be found in [6].

4. APPROACH

Since it is not known which features and methods are suitable for the synchronization task, an iterative prototype-driven development approach will be used. Prototypes will be benchmarked by classic measures like precision and recall, ROC curves and F-score.

The first step in this project is to create a precise synchronization ground truth which my developed methods can be evaluated on. As mentioned in Section 3, there is no published dataset available without heavy downsides, and above all, none supplies a clearly defined synchronization ground truth. As a basis, I selected the publicly available Jiku Mobile Video Dataset [7], that contains hundreds of videos recorded by members of the crowd with mobile devices under various conditions at five different musical events, and which therefore perfectly represents a real-world use case. I have additional access to non-public datasets from other research projects. Using a crowd-sourced dataset limits the benchmarking comparability mainly to the crowd use case, but it is an emerging topic and it is also a good check on robustness, as the performance of synchronization methods will tend to perform even better on higher quality recordings like in the case of an amateur production. Creating a synchronization ground truth on this real-world dataset renders also the examination of time drift necessary.

The second step is to extend the audio-based synchronization method from my Master's thesis [1] with short-time video synchronization. This is the first step towards multimodality. At this stage, synchronization points will already be known from the audio synchronization, but they will not be perfectly precise due to some degree of inaccuracy inherent to the audio method, and the video tracks will not

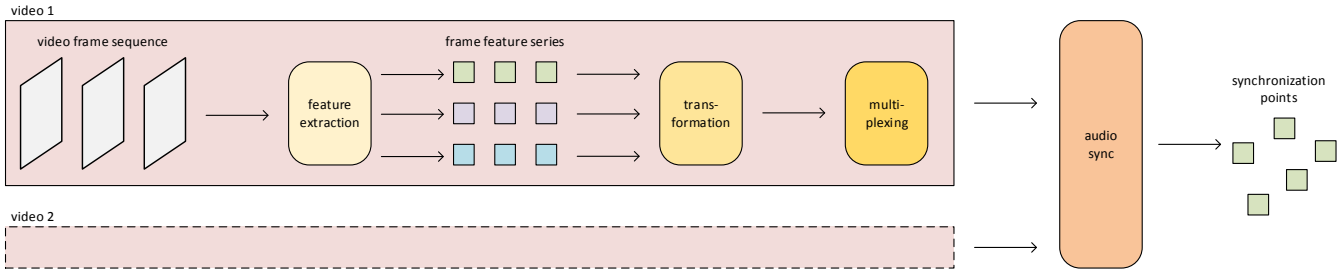

Figure 1: Schematic overview of the anticipated video synchronization system processing two overlapping videos. The final system will be able to process an arbitrary number of videos.

be perfectly in sync due to different speeds of sound and light. Short-time synchronization will try to remove offsets and improve synchronization points, depending on sampling and frame rates, as precise as possible by thoroughly analyzing the multimedia data in a short surrounding interval of a few seconds. The video synchronization extension will be based on local and global low-level features based on color, luminance, or motion, of which possible candidates will be identified and the best fitting ones assessed.

The third step is to develop the video-based synchronization approach into a long-time method, that can process long-running videos in their entirety, with the goal to be usable standalone. Since video frames are usually much less frequent than audio samples, it will not be able to reach an accuracy that is satisfactory enough for related audio tracks, but it will suffice for videos, and it will make the synchronization of videos without audio tracks possible. Section 3 references some already existing video processing synchronization methods, but the novelty of the approach here will be the much higher efficiency through a much lower computational and spacial complexity. I want to introduce a new way of video synchronization by converting the video synchronization problem into an audio synchronization problem through a transformation of multiple time-series of low-level feature values into a single audio-like signal in the audio sample or frequency domain, as sketched in Figure 1. This distantly relates to sonification, where non-audio data gets transformed to audio signals. It will enable the use of the already manifold existing audio synchronization algorithms and audio processing tools and save a lot of time and effort in designing and implementing new video synchronization tools. Focus will be put on the robustness of the method to avoid unnecessary restrictions on applicable use cases. I will conclude with an evaluation of the performance of multimodal synchronization by combining the synchronization results obtained from audio and video.

5. RESULTS

The starting point and foundation of this work is my Master's thesis [1] and the developed synchronization software presented in [2]. In the thesis, I have developed a synchronization method based on audio fingerprinting. Due to its low computational demands, it can be applied to large collections of recordings at once, and I evaluated it on various datasets for real-world applicability. It synchronizes all recordings that belong together (also non-overlapping ones through transitivity), and separates unrelated clusters of overlapping recordings from each other. Besides the automatic synchronization, the software provides a rich graphical interface to the user. It allows precise manual synchronization of recordings and thorough inspection of the automatically calculated synchronization points. It also provides helpful tools for easy manual detection and compensation of drift. Additionally, it includes functions to refine synchronization points through cross-correlation and the detection of non-linear drift through dynamic time warping, which even allows the synchronization of different interpretations (covers) of the same piece of music through non-linear resampling. Summing up, it is a very helpful toolbox for synchronizing and analyzing synchronization results which I have been using for almost all my follow-up research. It is currently limited to audio processing, but designed for extendibility with additional processing methods.

The first contribution of my dissertation research shows that time drift is a major problem that needs to be taken care of. It affects all recordings from non-professional grade devices, particularly mobile devices. The only exception are special cases where the involved devices suffer from comparable amounts of drift, in which case the effect can be ignored. Detailed measurements have shown that the drift can largely differ between different kinds of device makes and models, but its variance is limited inside a production batch of the same make and model. Due to different influences like temperature, age and power supply, the drift is not linear; its main influence is temperature, but the major part of the drift is a constant originating from production. The measured drift of a device at room temperature turned out to be sufficient for removing most of it in post-processing, e.g. by resampling, which turned out to be the most favorable method for drift removal in audio tracks. Drift removal in video tracks remains an open topic and novel methods need to be sought for, since simple methods like resampling or frame dropping/duplicating leads to highly visible quality degradations. This paper is still in submission.

The second contribution is a mobile app that is capable of instant drift measurements of playback and recording devices [3]. A measurement is always conducted between two devices, where one acts as the source of a test tone, and the other as the target that analyzes the tone. By precisely measuring the frequency shift between the two devices, it can quickly calculate the drift and show it to the user on the fly. This tool serves as technical demo to make the community aware of the problem, but also helps to determine devices that go well together for multi-camera recording, or to remove the drift in post-production.

The third contribution is a synchronization ground truth for the Jiku Mobile Video Dataset by using a semi-automatic approach that focuses on the audio tracks. Reviews from two A-conferences attest the dataset great potential, still it has been rejected for being out of scope which makes it also still being in the submission stage. Reviewers acknowledge the need of such a testing ground and benchmarking dataset in the community for video synchronization techniques. The dataset has been cleaned from drift and hundreds of synchronization points that were used to create the ground truth have been adjusted and validated manually. Based on the ground truth, I was able to show that cross-correlation cannot be used to validate or improve the manually defined synchronization points, backing their high accuracy and validity. By evaluating the fingerprinting synchronization method in my Master's thesis, I have shown that this dataset is a precious tool for the assessment of synchronization algorithms, as it allows for very precise measurements of the overall performance and effects of fine-tuning.

An additional contribution worth pointing out is the evaluation of a developed tablet application for the annotation of endoscopic surgery videos [4]. My dissertation research was originally planned to include semantic synchronization of videos recorded at different times or places but containing semantically similar content, e.g. the same surgery carried out on different patients. Since I have shown that non-linear synchronization of different interpretations of pieces of music can work very well, the idea was to synchronize surgery videos to help surgeons find similar sections in videos of similar surgeries. The purpose of the tablet app was to collect metadata that supports the synchronization, but I had to remove this approach from my dissertation research due to time constraints. If time permits, I will still evaluate the possibility of synchronizing such recordings with the developed video synchronization method combined with dynamic time warping to account for the non-linear mapping of the videos.

6. WORK IN PROGRESS

Current work in progress is the preparation for the development of the video-based synchronization method. I am starting with building a framework for the system depicted in Figure 1, selecting feature candidates, building synthetic test sequences, recording basic real-world test videos, and evaluating their possible usefulness through prototyping. Once potential features have been selected, I will build a processing library that transforms video streams into pseudo-audio streams that can be analyzed by audio synchronization methods. The major challenge will probably be the encoding scheme. I expect that the differences between sequential frame features will be more appropriate than the absolute feature values. Scalar feature value series can presumably be directly encoded as time-series signals, or by amplitude (AM) or frequency modulation (FM) on carrier signals. Complex series, e.g from histograms, can probably be encoded by AM or FM on carrier signals of different frequencies, or encoded in the frequency domain and transformed to the time domain by iFFT. I anticipate a scheme to combine multiple features into one output signal by encoding them into different frequency bands. I also anticipate this idea to work since audio synchronization methods heavily rely on energy peaks or sharp local changes in the spectrogram, and changes in the video signal with an appropriate transformation should yield similar patterns. As a simple example, encoding a signal by taking just the luminance of a video as the single feature will yield a signal that can be similarly analyzed like the flash detection in [9], and adding additional features could make the method more robust and versatile for different use cases. If this turns out to be wrong, it should at least be possible to customize the method for different use cases by selecting individual combinations of features based on the visual properties of the videos to be processed.

7. ACKNOWLEDGMENTS

This work was supported by Lakeside Labs GmbH, Klagenfurt, Austria, and funding from the European Regional Development Fund (ERDF) and the Carinthian Economic Promotion Fund (KWF) under grant 20214/22573/33955.

8. REFERENCES

[1] M. Guggenberger. Synchronisation von Multimediadaten auf Basis von Audiospuren. Master's thesis, Alpen-Adria-Universität Klagenfurt, Austria, 2012.

[2] M. Guggenberger, M. Lux, and L. Böszörmenyi. Audioalign - synchronization of A/V-streams based on audio data. In *Multimedia (ISM), 2012 IEEE International Symposium on*, pages 382–383, 2012.

[3] M. Guggenberger, M. Lux, and L. Böszörmenyi. Clockdrift: A mobile application for measuring drift in multimedia devices. In *Proceedings of the 22st ACM International Conference on Multimedia*, MM '14, New York, NY, USA, 2014. ACM. to appear.

[4] M. Guggenberger, M. Riegler, M. Lux, and P. Halvorsen. Event understanding in endoscopic surgery videos. In *Proceedings of the ACM Multimedia 2014 Workshop on Human Centered Event Understanding*, New York, NY, USA, 2014. ACM. to appear.

[5] L. Kennedy and M. 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*, WWW '09, pages 311–320, New York, NY, USA, 2009. ACM.

[6] A. Llagostera Casanovas and A. Cavallaro. Audio-visual events for multi-camera synchronization. *Multimedia Tools and Applications*, pages 1–24, 2014.

[7] M. Saini, S. P. Venkatagiri, W. T. Ooi, and M. C. Chan. The jiku mobile video dataset. In *Proceedings of the 4th ACM Multimedia Systems Conference*, MMSys '13, pages 108–113, New York, NY, USA, 2013. ACM.

[8] S. Shankar, J. Lasenby, and A. Kokaram. Warping trajectories for video synchronization. In *Proceedings of the 4th ACM/IEEE International Workshop on Analysis and Retrieval of Tracked Events and Motion in Imagery Stream*, ARTEMIS '13, pages 41–48, New York, NY, USA, 2013. ACM.

[9] P. Shrestha, M. Barbieri, H. Weda, and D. Sekulovski. Synchronization of multiple camera videos using audio-visual features. *Multimedia, IEEE Transactions on*, 12(1):79–92, 2010.