MoVieUp: Automatic Mobile Video Mashup

Yue Wu*, Tao Mei†, *Senior Member, IEEE*, Ying-Qing Xu‡, *Senior Member, IEEE*, Nenghai Yu*, *Member, IEEE*, and Shipeng Li†, *Fellow, IEEE*

***Abstract*—With the proliferation of mobile devices, people are taking videos of the same events anytime and anywhere. Even though these crowdsourced videos are uploaded to the cloud and shared, the viewing experience is very limited due to monotonous viewing, visual redundancy, and bad audio-video quality. In this paper, we present a fully automatic mobile video mashup system that works in the cloud to combine recordings captured by multiple devices from different view angles and at different time slots into a single yet enriched and professional looking video-audio stream. We summarize a set of computational filming principles for multi-camera settings from a formal focus study. Based on these principles, given a set of recordings of the same event, our system is able to synchronize these recordings with audio fingerprints, assess audio and video quality, detect video cut points, and generate video and audio mashups. The audio mashup is the maximization of audio quality under the *less switching principle*, while the video mashup is formalized as maximizing video quality and content diversity, constrained by the summarized filming principles. Our system is different from any existing work in this field in three ways: 1) our system is fully automatic, 2) the system incorporates a set of computational domain-specific filming principles summarized from a formal focus study, and 3) in addition to video, we also consider audio mashup which is a key factor of user experience yet often overlooked in existing research. Evaluations show that our system achieves performance results that are superior to state-of-the-art video mashup techniques, thus providing a better user experience.**

***Index Terms*—Mobile video, video-audio mashup, filming principle, cloud media computing**

I. Introduction

We are now witnessing a rapid proliferation and renovation of cloud computing techniques. In this cloud computing world, people are taking and sharing more and more videos with few to no restrictions on mobility and high-speed connection to the cloud [1]. Among them, there exist *multi-camera recordings* which are captured simultaneously at the same event and partially overlap in time [2]. The viewing experience of such *multi-camera recordings* is very limited. First, it is time consuming to watch all of them one-by-one to get an overview of the event. Viewers will lose interest due to the monotonous view of a mobile video. Second, the contents of videos captured by different people can be similar, making them bad choices either for collecting or sharing. Third, the quality of these videos is not guaranteed as they are often captured under poor conditions by amateurs using handheld devices. To deal with these challenges, mobile video mashup, which synchronizes and combines multi-camera recordings into a single yet enriched and professional looking video-audio stream, has become an emerging research topic. Fig. 1 shows an illustration of a typical video mashup, where the mashup is only based on video signal.

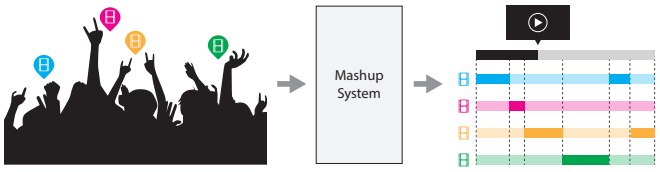


Fig. 1. An illustration of a video mashup. Given a set of recordings of the same event, our system is able to generate a mashup video based on a set of summarized filming principles. Selected parts of the source video recordings are shown in deep colors. The automatically generated mashup video provides a richer and much more professional view of the event than any single recording, thus significantly improving user experience.

Numerous challenges have created barriers to producing a successful mobile video mashup. The first challenge comes from the quality of the input videos. Though imaging techniques have taken an enormous leap forward, video quality is affected significantly by shakiness, blurriness and many other factors occurring during the capturing. Besides, audio quality is limited by both surroundings and the microphone itself. To the best of our knowledge, there has been limited research on assessing non-intrusive audio quality. Another challenge is that we do not know how human editors will create the video mashup. Different editors may have different editing styles, making it hard to learn a general model from existing videos. Even though we know the process of human editing, we still need to transfer it to computable formulations. Such a transfer is the foundation to overcoming the two basic challenges in video mashup: when is the appropriate time to switch to another video/audio source, and how does the mashup system select the best video/audio source among all the candidates.

Some works addressing these challenges have been published previously. These approaches can be summarized into three categories: rule-based [3], optimization-based [2], and learning-based [4], [5]. Rule-based methods imitate the process of human editors. However, the requirements in mobile video mashup are more like user preferences rather than strict conditions. Shrestha *et al.* propose conducting video mashup by optimization, in doing so only visual quality and diversity are actually optimized in their system [2]. Additionally, the optimization is only a local approximation through greedy search. *Jiku Director* is proposed to learn a transition matrix

Y. Wu and N. Yu are with the Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei 230027, China (e-mail: wye@mail.ustc.edu.cn; ynh@ustc.edu.cn).

T. Mei and S. Li are with Microsoft Research, Beijing 100080, China (e-mail: tmei@microsoft.com; spli@microsoft.com

Y.-Q. Xu is with the Department of Information Art & Design, Tsinghua University, Beijing 100084, China (e-mail: yqxu@mail.tsinghua.edu.cn)

(shooting angle, distance to the target, and shot length) to select video shots [4], [5]. The learned editing rule is not video content-based. We argue that automatic video editing is an organic combination of the film grammar and the computable elements, while none of these methods solve the mashup problem in such a comprehensive way. Besides, they pay little attention to audio [2], [4], [5], which plays an important role in both visual and aural experience. For video mashup, audio can be utilized to determine the switching frequency of shots (video cut point detection). The reason to do audio mashup is three-fold: 1) The final output is comprised of both visual and aural signals. Thus the audio source with the best quality should be selected as the final output. 2) Almost none of the audio sources record the whole event in real cases. 3) The Quality of a video is good does not mean that the correspondent audio is also good. Hence we cannot use the corresponding audio stream when a video is selected.

In this paper, we propose a new mobile video mashup system—MoVieUp which integrates film grammar into an optimization problem with rule-based constraints. We summarize a set of computational filming principles from a focus study. Based on these principles, we design a system that generates a mashup of both audio and video. Specifically, source videos are synchronized with audio fingerprints. For audio mashup, we select the best quality parts of different audio sources and stitch them together to generate a full aural recording of the whole event under the *less switching principle*. For video mashup, we first detect cut points by measuring tempo suitability and semantic suitability on audio. Given these cut points, we formalize video mashup as the optimization of visual quality and diversity under the constraints of camera motion consistency. A post-processing step is performed for semantic completeness of videos. The final output is the combination of the video and audio mashup.

Our main contributions in this work are as follows:

1) We propose a fully automatic system for mobile video mashup, which is significantly different from previous works requiring kinds of manual intervention.
2) The system incorporates a set of computational domain-specific filming principles summarized from a formal focus study. These computational principles provide the guidance for switching cameras and selecting the best cameras in the multi-camera setting.
3) In addition to video, we also consider audio mashup which is a key factor of user experience yet often overlooked in existing research. This has significantly improved the overall watching experience of mobile video.

The rest of the paper is organized as follows. We discuss some related works in Section II. Section III presents the focus study. Section IV describes our mobile video mashup system. We conduct some experiments to evaluate the performance of the proposed system in Section V , followed by conclusions and discussions in Section VI.

II. Related Work

Mobile video mashup has emerged as a popular research topic in recent years. Beside researching video mashup itself, there are also connections with video editing processes.

TABLE I
Comparison of video mashup systems

	MoVieUp (this paper)	VD [2][1]	Jiku [5]
diversity	Yes	Yes	Yes
shakiness	Yes	Yes	Yes
tilt	Yes	No	Yes
occlusion	Yes	No	Yes
audio mashup	Yes	No	No
cut point	Audio+Video	Manual	Learning[2]

[1] VD is short for Virtual Director [2].
[2] Transition matrix for cut points learnt by Jiku Director is the same to all videos, thus not content-based.

A. Video mashup

There have been a few works on video mashup in recent years. Shrestha *et al.* propose an automatic mashup generation system from multiple-concert recordings [2]. They formulate mashup generation as an optimization of many factors like video quality, diversity, and cut point suitability. However, the authors do not present a practical method for measuring cut point suitability, making the system not fully automatic. Some video quality factors which are common in mobile videos, like tilt and occlusion, are not considered either. Saini *et al.* propose the *Jiku Director* to do online mobile video mashups [4], [5]. They learn a Hidden Markov Model (HMM) for shot selection and shot length determination. The problem with this approach is that there are many kinds of editing styles, both linear and non-linear. Shot selection and shot length should be content based, influenced by motion intensity, audio tempo, and many other factors [6]. Besides, the system is not fully automatic in that the accuracy of the automatic classification of camera locations is limited. Manual intervention is preferred, especially in the learning phase. Neither of the above methods consider audio mashup, nor do they pay enough attention to the principles of video editing. Arve *et. al* have recently proposed an automatic editing system for footage from multiple social cameras [7]. However, the system is highly dependent on 3D reconstruction of scenes, which often fails with mobile videos as we considered. There are a few other systems called mashup [8], [9]. However, they are dealing with selection of video clips from different movies, which is quite different from the multi-camera settings for mobile video mashup.

In Table I, we provide a comparison of our proposed system with the two existing mashup systems from the perspective of both audio and video mashup.

B. Video editing

Mobile video mashup is also related to video editing, including video summarization, camera selection, and home/music video editing.

Video summarization. Video summarization shares a goal with video mashup in that both of them aim to maximize information content. Sundaram *et al.* propose a utility framework

for automatic skim generation from computable shots [10]. They apply visual film syntax to the arrangement of shots (selection, scale, duration, order, etc.) [11], which is referential in our mobile video mashup. Detection of computable shots is often based on a mimic model of human memory [12]. Similar models can be used to diversify mashup video.

Camera selection. Camera selection has been widely explored in some specific scenarios like lectures and meetings. Cameras are often selected by recognizing speakers or detecting faces [13], [14]. Tracking and audio based localization are used in related systems [15], [16]. All the above mentioned methods can be classified as speaker-based. However, for video-audio mashup, we consider more general cases like a concert where speakers are not the only focus. A noisy environment and bad visual quality make it hard to do audio localization or face detection.

Home/music video editing. There have been many works that focus on home/music video editing. Hua *et al.* present AVE—an automated home video editing system, to extract a set of highlight shots from a set of home videos [17]. They propose two sets of rules to ensure representativeness of the original video, as well as the coordination between video and audio. A similar approach is extended to automatic music video generation, in which temporal structures of videos and music are analyzed for matching [6]. However, these systems cannot be applied to *multi-camera recordings* since mobile video mashup requires synchronization and video diversity.

III. Focus Study

Mobile video mashup is highly related to the film editing conducted by human beings. In film editing, editors use the film grammar to grasp the attention and emotions of an audience with fragments of recorded time, arranged shadows and sounds to convey a story. *"Metaphorically, the "grammar" of the film refers to theories that describe visual forms and sound combinations and their functions as they appear and are heard in a significant relationship during the projection of a film. Thus, film grammar includes the elements of motion, sound, pictures, color, film punctuation, editing, and montage." [18].* Basic elements of the film grammar include shot, movement, and distances (full, medium, and close-up) [11]. They use various shot lengths, shooting angles, and arrangement of distances to express different meanings of the shots. For example, local object motion can be expressed with medium shot, while close-up is often used for static shots or shots with moderate motion.

Mobile video mashup mimics the process of film editing to convey the event originally expressed by the *multi-camera recordings*. Cut points in mobile video mashup correspond to the boundaries of different shots. Selection of video shots corresponds to the selection of camera positions. To create rich and professional looking videos, it is required to know the grammar of the film language and how human editors apply the grammar to video editing.

A major barrier to apply the film language to mobile video mashup is that the film language is not strict rules. Though there have been many works talking about the film grammar and some of the observations in this section are not new, it is still essential to summarize existing filming principles and explore new guidelines that are both specifically related to mobile video mashup and feasible to be interpreted into computational rules, which lays the foundation for the proposed system. We survey previous user study [2] and the literature on video editing [11], [19]. Further, we conduct a formal focus study centered on the two basic problems of video mashup:

- **Switching of shots**: when should the mashup switch to another video source?
- **Selection of cameras/recorders**: which video/audio source should be chose next?

A. Participants and procedure

We invited a professor of Arts & Design and a graduate student in cinematography, to attend the focus study. The professor has been working on video-related research for more than 20 years. The graduate student has much experience in video editing, particularly in collaboration with TV stations.

The focus study is loosely structured and conducted in a discussion format. We first show the typical capturing scenarios (like concerts and competitions) and major drawbacks (lighting, shakiness, occlusion, etc.) of mobile videos. Then we try to explain what is mobile video mashup and ask the two of them some questions about it. The questions are structured as follows:

- *Video mashup discussion*: This discussion is to investigate video switching frequency and video shot selection. We ask questions such as: Are there any requirements for the duration of a video shot? What factors will affect the duration and how do they affect it? Can we draw a certain connection between these factors and the duration? Are there any requirements for the switching? What will affect video quality? How does one avoid the monotony problem of mobile videos? How does one switch between video shots smoothly? Do you have any suggestions on how to improve the viewing experience?
- *Audio mashup discussion*: The discussion concerns the selection of audio with questions such as: When does one need to switch from one audio source to another? What kind of audio do you think is better? Are there any differences between video mashup and audio mashup? How can we concatenate audio fragments from different sources together?

B. Results of video mashup

Switching of shots. The two editors said that there should be a lower and upper bound to the duration of shots. Too short a video is incomprehensible, while too long will be boring. The bounds are not constant. Shrestha *et al.* choose a minimum of 3 seconds and a maximum of 7 seconds for concert videos [2]. However, the bounds may not be so strict. Longer shots can be used for moving shots or establishing shots. Moving shots bring new content to viewers, thus avoiding boringness. In establishing shots, a complete view with rich content is shown and boringness is not a concern either.

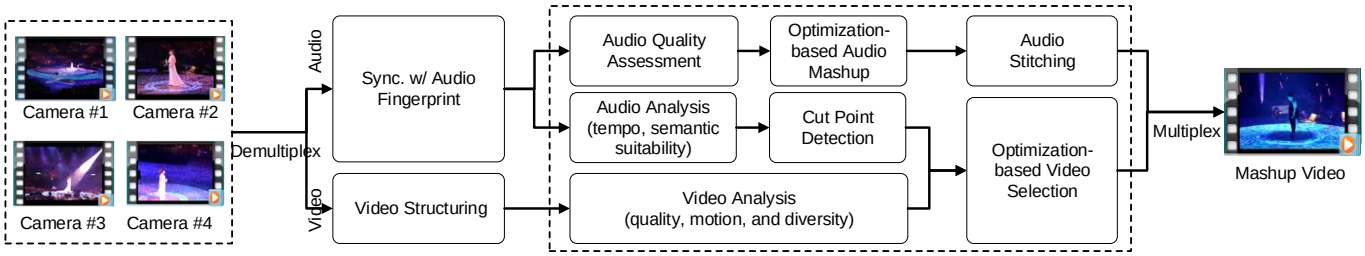


Fig. 2. Framework of the proposed mobile video mashup system. The system consists of two main parts: video mashup and audio mashup.

Switching frequency of videos depends on both audio and video. Frequent switching is preferred for fast-paced audio, intense object motion, and rapid change in brightness. Smooth shots should be paired with less switching. There is no definite relationship between switching frequency and these factors. Whether it is a linear or non-linear correlation is up to the directing style.

Suitable switching should occur near speaking/singing intervals, beginning of speaking/singing captures viewers' attention, and ending releases it. To avoid interruption of the viewers' attention, switching near speaking/singing intervals is often a good choice.

Selection of cameras. Computational factors related to video shot selection include: video quality, diversity, camera motion, and semantic completeness.

- *Video quality*. Video quality ensures clarity and a pleasant viewing experience. Observations include: too dark or too bright frames should be excluded; blurred frames should be avoided; occluded frames will damage viewers' interest; tilting makes viewers uncomfortable, though it can bring some special effects in certain cases. The editors especially made a point of reminding us that bad effects caused by erratic and unprofessional camera motions (hand-shaking, rapid movement of the camerawork, etc.) must be excluded.
- *Diversity*. Diversity means an enriched and entertaining view of an event. According to the study, there are no definite principles to guide the selection of camera positions. Different editors may have different editing styles and thus different choices if many candidate camera positions are available. However, there exist some rules that should not be violated. For example, a core guideline is to avoid "*Jump Cuts*", which means that two sequential shots of the same subject are taken from camera positions that vary only slightly. The *30 degrees rule* indicates that there should be at least 30 degrees' difference between shooting angles to avoid a noticeable portion of overlap between adjacent shots. The editors recommend that if camera positions are unknown, frame difference should be large enough to present some fresh content to viewers.
- *Camera motion*. The change of camera motion between neighboring shots should be smooth for a comfortable viewing experience. Unexpected camera motion can incur annoying visual impact. Some common principles include: (1) static shots should be connected with static shots; (2) moving shots are less placed near each other; (3) visual impact due to the connection of static and moving shots can be alleviated by the slowing down of camera motion, object motion in static shots, or bridging shots.
- *Semantic completeness*. The two editors mentioned some semantic concerns about video mashup. Each recording is comprised of many self-contained semantics (subshots as we called them later). These self-contained semantics should not be interrupted before viewers get a basic knowledge of them. Otherwise, the switching will be quite obtrusive.

C. Results of audio mashup

Unlike video mashup, switching between audio recordings should be as minimal as possible, which we note as *Less Switching Principle*. There is no monotony problem if there are no or little switching between different audio sources. Conversely, stitching audio shots between different sources will create more inconsistency problems due to variances in volume and tone, even in the case that the shots are synchronized exactly. The inconsistency can be caused either by the microphones or a recorder's surroundings, degrading the overall quality of the mashup audio. A high quality audio generated from one or a few sources is more preferred. Considering the mobile scenario, clamorous audio should be avoided. Selected audio or audio fragments are expected to be clear and clean. Audio mashup should distinguish good audio fragments from other noisy ones.

D. Computational filming principles

We summarize some computational filming principles from the focus study. These principles will be the foundations of the proposed mobile video mashup system.

- For video cut point detection, the following standards should be met: 1) there should be a lower and upper bound on shot duration; 2) switching frequency of videos should be consistent with audio tempo; 3) switching should take place near speaking/singing intervals.
- Selected video shots should fit the following criteria: 1) selected video shots should be clear and stable (without blurriness, occlusion, shakiness, etc.); 2) frame difference of adjacent shots should be large to avoid *Jump Cuts*; 3) camera motion around cut points should be smooth and natural; 4) each selected shot should be complete in terms of semantics.

- Selected audio fragments should be: 1) clear and clean, and 2) with the *Less Switching Principle*. That is, switching between audio fragments should be as minimal as possible.

IV. MoVieUp System

In this section, we describe the framework of the MoVieUp system based on the filming principles summarized above. Note that in later sections, video and audio denote the visual and aural content respectively. We use recording to denote both of them.

A. Overview

As analyzed before, the MoVieUp system consists of the generation of both audio mashup and video mashup. To clearly present the system framework, the following terms are clarified for video mashup:

- *Shot*: a shot is a fragment of a video that is selected as part of the mashup video.
- *Subshot*: a subshot is the basic unit of video. It contains consistent camera motion and self-contained semantics [20].
- *Cut point*: a cut point is the time point where the mashup video switches from one source to another. Note that at each cut point, there can be multiple candidate video sources.

1) Framework: Fig. 2 shows the framework of the proposed system. Given a list of recordings, MoVieUp is able to demultiplex them into audio and video, synchronize the recordings, generate the mashup video and audio according to the summarized filming principles, and multiplex them into the final video-audio stream. For audio mashup, we select audio fragments based on quality under the *less switching principle*. Video mashup is comprised of two steps: cut point detection and optimization-based video shot selection. The system detects cut points by matching switching frequency with audio tempo and avoiding speaking/singing interruption [6]. Video analysis is performed at the granularity level of subshots. Given the detected cut points, video shot selection is formalized as maximizing video quality and diversity under constraints of camera motion consistency. Results of video mashup are fine-tuned with semantic completeness. Video stabilization is presented as an optional step to further improve the viewing experience. After the system finishes video and audio mashup, the two separate results are multiplexed into the final output.

2) Notations: Suppose there are N source recordings $\mathcal{R} = \{r_1, r_2, \ldots, r_N\}$. Each recording r_i is demultiplexed into audio a_i and video v_i. r_i starts at time $t_i^{(s)}$ and ends at time $t_i^{(e)}$ (so is v_i and a_i). Suppose there are M^a and M^v shots in the mashup audio and video, respectively. We denote the *j-th* selection by s_j^a and s_j^v (th *j-th* shot). The superscript a or v is to distinguish audio and video. The mashup audio and video are described as:

$$\begin{aligned}\mathcal{M}^a &= (s_1^a, s_2^a, \ldots, s_{M^a}^a)\\ \mathcal{M}^v &= (s_1^v, s_2^v, \ldots, s_{M^v}^v).\end{aligned}$$

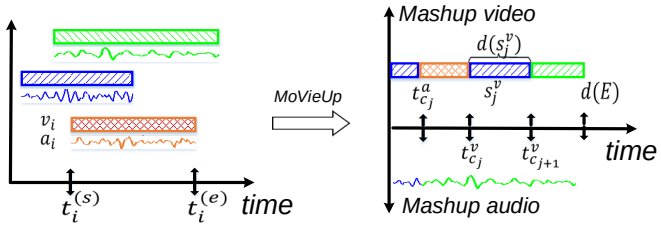


Fig. 3. Illustration of notations. Note that we use $t_{c_j}^a$ to denote the switching time of audio for convenience. s_j^v denotes the *j-th* shot.

The duration of the *j-th* selection s_j is denoted by $d(s_j)$.

For video mashup, there are an upper bound d_{max} and a lower bound d_{min} for the duration of each selection: $d_{min} \leq d(s_j^v) \leq d_{max}$. A cut point c_j is the time point where the video switches from s_{j-1}^v to s_j^v. We denote this time point by t_{c_j}. The beginning of an event is regarded as a cut point for convenience, thus $t_{c_1} = 0$. Figure 3 illustrates the meaning of these symbols.

B. Pre-processing

Before processing, the audios are sampled to 8kHz for synchronization and quality assessment. The video frame rate is normalized to 25 frames/second. The resolution of frames is resized to the same(640×360 for example).

Synchronization. Input recordings lie on different periods of an event. We need to synchronize them for further processing, that is, to determine the start time $t_i^{(s)}$ and end time $t_i^{(e)}, \forall r_i \in \mathcal{R}$. The basic assumption is that there exists at least one candidate recording anytime during the whole event. In our system, we adopt the synchronization method with audio fingerprints presented in [21], [22]. We first extract audio fingerprints and compare them to calculate the time offsets for each pair of audio. A voting scheme is then performed to mutually determine the time offsets of all the recordings.

Both video and audio mashup must follow a *synchronization constraint*: the starting time of the selected item must be earlier than the current cut point, and the end time later than the next cut point.

$$t_{s_j}^{(s)} \leq t_{c_j} \leq t_{c_{j+1}} \leq t_{s_j}^{(e)} \tag{1}$$

If a candidate audio/video does not satisfy this constraint at a switching time point, it will be excluded from consideration.

Video Structuring. Operation on videos can be performed on three temporal layers: frame, subshot, and shot. As talked in [23], frame layer operation is not only time-consuming but also difficult for further content analysis, since frame is not the most informative semantic unit of videos. Shot is a physical video structure resulting from the users' start and stop operation. It usually lasts a relatively long period of time and contains inconsistent content. In our mashup system, subshot is the basic unit of video mashup, as it contains consistent camera motion and self-contained semantics [20]. We apply the color and motion threshold based algorithm as described in [20] to structure the videos.

C. Audio mashup

Audio mashup is the process of combining all the audio recordings into a single but complete sound recording during the whole event. As we stated, each audio may only records part of the whole event. It works by assessing audio quality and selecting the highest one under *less switching principle*.

According to the results in our focus study, the main goal of audio mashup is to maximize the overall quality of the selected audio fragments $Q(\mathcal{M}^a)$. Recall that $Q(\mathcal{M}^a)$ does not simply equal the sum of all selected audio fragments, due to the fact that the switching of audio will degrade the overall quality. We studied the mobile audio carefully and found that the quality does not fluctuate dramatically and frequently for each audio recording. Based on this observation, we embody the *less switching principle* to a hard constraint when switching to another audio source. Let $q_{s_j^a}(t)$ represents the quality score of s_j^a at time t. Switching is only performed when the quality of some other audio is much better than the current one.

$$q_{s_{j+1}^a}(t) \quad > \gamma \cdot q_{s_j^a}(t), \tag{2}$$

where γ is to penalize the switching and is set to 1.2 in our experiments.

We adopt a greedy search strategy to generate the mashup audio by checking the quality of candidate audios every a second. Switching to a new audio source happens when the current one ends or another audio is much better than the current one (equation (1) and equation (2)).

1) Quality assessment: In the above solution, we need to evaluate audio quality. As far as we know, few works have been focused on non-intrusive audio quality assessment, except for some on speech signals [24]. Li *et al.* model non-reference audio quality assessment as a learning-to-rank problem [25]. It seems to be the first approach to music audio as they claim. This approach is not applicable in our scenario, as we expect meaningful quality scores in equation (2).

In our system, we use P.563, a non-intrusive speech quality assessment algorithm [26], to assess audio quality. Such an approximation is based on the assumption that mobile *multi-camera recordings* are often captured at events where speaking or singing is the major signal. We assess audio quality on a five second sliding window with a time step of one second.

To further verify whether P.563 works in our settings, we randomly select four mobile concert audio recordings and download the corresponding music, which we call as the reference audio. We evaluate the quality of these two types of audio with P.563 respectively. The quality scores are shown in Fig. 4. In audio mashup, we want to select audio fragments that have good quality and last long, like the beginning of Audio 2 in Fig. 4.

2) Audio stitching: In the final step, we need to stitch the audio fragments to the mashup audio. Audio is different from video in that people are sensitive to sudden changes. As input audio recordings are recorded in different locations and different surroundings, directly concatenating these audio fragments will result in the sudden change due to variant volume and tone. To overcome this problem, we first apply DC offset correction to adjust audio volume gain. To alleviate the seam effect in concatenating audio, an image blending algorithm with a laplacian pyramid is adopted to stitch audio fragments together.

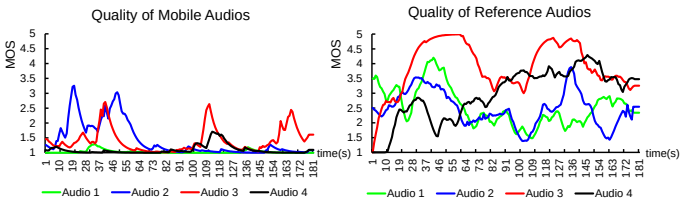


Fig. 4. An example of audio quality assessment. We randomly select four mobile concert audio recordings and download the corresponding music, which we call as reference audio. We evaluate the quality of these two types of audio with P.563 respectively. The left plot is the quality of the four mobile audio recordings. The right plot is the quality of the reference audios. The horizontal axis is the timeline. And the vertical is the quality. MOS is short for Mean Opinion Score. We can find that score of mobile audio is much lower than that of reference audio, which verifies the efficacy of P.563 on mobile audios.

D. Video cut point detection

Video mashup is to select *shots* from the input videos and combine them together for a single yet enriched and professional looking video stream. It is divided into two steps: cut point detection and video shot selection. In this section, we talk about cut point detection which is to determine when the mashup video switches from one video source to another.

1) Formulation: According to the filming principles, video cut point detection is a comprehensive concern of both audio (tempo, speaking/singing interval) and video (camera motion, subshot integrity). The system first detects candidate cut points from the mashup audio. As to video, we require motion consistency and semantic completeness at the candidate cut points later. We propose two suitability scores for cut point detection: tempo suitability $S^T(t)$ and semantic suitability $S^S(t)$. Tempo suitability controls switching frequency with respect to audio tempo. Semantic frequency avoids interruption of speaking/singing. A new cut point is detected on the assumption that cut points earlier than that have already been determined. The problem is formulated as:

$$\begin{aligned} t_{c_j} = \underset{t}{\operatorname{argmin}} \big\{ S^T(t|t_{c_{j-1}}) + S^S(t|t_{c_{j-1}}) \big\}, \quad s.t. \\ d_{min} \leq t - t_{c_{j-1}} \leq d_{max}. \end{aligned} \tag{3}$$

2) Tempo suitability $S^T(t|t_{c_{j-1}})$: Video switching frequency should be consistent with audio tempo. Fast-paced audio should be paired with frequent switching. Less switching is preferred in smooth events. We use the interval between audio onsets to approximate audio tempo [17], [27]. As we discussed in Section III, there is no definite relationship between audio tempo and switching frequency. We map the tempo $b(t)$ at time t to the expected duration $d(b(t))$ linearly as:

$$d(b(t)) = d_{max} - \frac{d_{max} - d_{min}}{b_{max} - b_{min}} (b(t) - b_{min}), \tag{4}$$

where b_{max} and b_{min} are the maximum and minimum tempo of the audio.

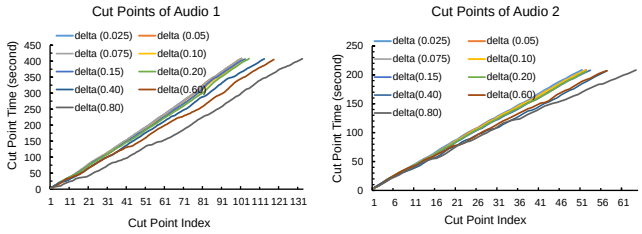


Fig. 5. Comparison of video cut points detection with different values of δ on two randomly selected audios. The values are in terms of seconds. It can be observed that detected cut points are similar when δ is less than 0.2.

The tempo suitability at time t is then measured by

$$S^T(t|t_{c_{j-1}}) = |\int_{t_{c_{j-1}}}^{t} \frac{1}{d(b(t))}dt - 1|. \tag{5}$$

3) Semantic suitability $S^S(t)$: Semantic suitability $S^S(t)$ means that we should avoid switching which attracts user attention. We achieve this goal by selecting points where audio energy is low, like the singing or speaking intervals. Semantic suitability is thus measured by the audio energy $e(t)$, normalized to $[0, 1]$.

$$S^S(t) = e(t). \tag{6}$$

4) Problem solution: It is hard to give a closed-form solution to the continuous objective function in (3), as the tempo suitability and semantic suitability are not smooth functions. Instead, the timeline is discretized into a step of δ seconds. The system enumerates all candidate time points in the time range $[d_{min}, d_{max}]$ from the previous cut point. The problem solution is then:

$$\begin{aligned} t_{c_j} &= t_{c_{j-1}} + K * \delta t, \\ where \quad K &= \underset{K}{\operatorname{argmin}} \left\{S^T(K) + S^S(K)\right\}, \\ S^T(K) &= |\sum_{k=1}^{K} \frac{\delta t}{d(b(k))} - 1|, \\ S^S(K) &= e(K) = e(t_{c^v_{j-1}} + K * \delta t), \\ b(k) &= b(t_{c^v_{j-1}} + k * \delta t). \end{aligned} \tag{7}$$

In the ideal case, the value of δ should be small enough to approach the continuous objective function. To select an appropriate value for δ, we randomly select some audios and detect cut points with different values of δ. Fig. 5 shows the example of two randomly selected audios. We find that when δ is less than 0.2 seconds, the detected cut points are very similar. As a result, we set δ to 0.1 seconds in our experiments. Let $d(e)$ denote the duration of the event, number of candidate time points to enumerate in the above function is $d(e)/\delta$, which is often in thousands of scale for a few minutes of video and costs little time in the cloud.

E. Video shot selection

Video shot selection is to determine which video source to switch to at each cut point, given the detected cut points above. In this section, we formulation video shot selection into an optimization problem constrained by camera motion consistency.

1) Formulation: As observed in focus study, video shot selection is a comprehensive consideration of video quality, diversity, camera motion and semantic completeness. As semantic completeness depends on selected shots, we will consider it in post-processing. Video shot selection is formalized as maximizing video quality and diversity under the constraints of camera motion consistency. Specifically, we do not allow camera motion around cut points to simplify the camera motion consistency constraint. Such a setting can: (a) alleviate visual impact caused by the connection of moving shots; (b) smoothen the change of camera motion as the constraints do not cause any interruption in camera motion.

We denote the camera motion vector at the left and right side of the *j-th* selected shot s^v_j by $\mathbf{m}^-(s^v_j)$ and $\mathbf{m}^+(s^v_j)$. The motion consistency constraint is represented by:

$$\mathbf{m}^-(s^v_j) = \mathbf{m}^+(s^v_j) = 0, \tag{8}$$

Let $Q(\mathcal{M}^v)$ represents the quality of the mashup video, and $D(\mathcal{D}^v)$ the diversity. The formulation of video shot selection is:

$$\begin{aligned} \mathcal{M}^v = \underset{(s^v_1,\ldots,s^v_{M^v})}{\operatorname{argmax}} &\left\{Q(\mathcal{M}^v) + D(\mathcal{M}^v)\right\}, \quad s.t. \\ &\mathbf{m}^-(s^v_j) = \mathbf{m}^+(s^v_{j-1}) = 0, \forall j \in [2, M^v]. \end{aligned} \tag{9}$$

2) Video quality assessment: Unlike audio, video quality does not degrade with switching. The overall video quality is calculated as:

$$Q(\mathcal{M}^v) = \sum_{j=1}^{M^v} Q(s^v_j), \tag{10}$$

where $Q(s^v_j)$ is the quality score of shot s^v_j at period $[t_{c_j}, t_{c_{j+1}}]$.

We employ a non-reference video quality assessment technique to measure mobile video quality. Video quality is measured based on six aspects in two categories: temporal and spatial [23]. Temporal factors, including *unstableness* and *jerkiness*, are caused by erratic camera motion, whereas spatial factors (*infidelity*, *brightness*, *blurring*, and *tilting*) are due to poor capturing environment. The six factors of video shot s^v_j, denoted by $u^v_{j,i} \in [0, 1], i \in [1, 6]$, are evaluated as unsuitability scores at the granularity of subshot. Overall unsuitable score U and quality score Q of a subshot are combined in two ways:

- Pre-filter bad quality candidate subshots. Any subshot with quality score of any factor lower than its threshold is regarded as unacceptable, with the unsuitability score set to a large value (1,000 in our experiment) to ensure that the shot is not selected, except in some cases where the video is still the best choice.
- An objective term in the optimization formulation. Overall score is calculated with a rule-based method [23]:

$$\begin{aligned} U(s^v_j) &= E(u^v_j) + \frac{1}{10 + 6\gamma} \sum_{i=1}^{6} (u^v_{j,i} - E(u^v_j)), \\ Q(s^v_j) &= 1 - U(s^v_j), \end{aligned} \tag{11}$$

where $E(u^v_j)$ is the average of the six quality scores of shot s^v_j. γ is a predefined constant which controls the amount of difference between $u^v_{j,i}$ and $E(u^v_j)$ and is set to an empirical

value of 0.20 as the same to [23]. Quality of a video shot $Q(s_j^v)$ is the minimum subshot quality in the shot.

3) Diversity: The overall diversity is calculated as the cumulative diversity of all the selections:

$$D(\mathcal{M}^v) = \sum_{j=1}^{M^v} D(s_j^v), \tag{12}$$

where $D(s_j^v)$ is the diversity score of the jth selection. It is mostly related to the content that viewers have seen and is still fresh in their memory. We measure diversity by how much a candidate selection will recall such content in memory. $D(s_j^v)$ is calculated as:

$$D(s_j^v) = D(s_j^v, s_{j-1}^v). \tag{13}$$

Similar to the memory model in [12], recall and diversity between two shots a and b are calculated as:

$$\begin{aligned} R(a,b) &= s(a,b) \cdot f_a \cdot f_b \cdot (1 - \frac{\Delta t}{T_m}), \\ D(a,b) &= 1 - R(a,b), \end{aligned} \tag{14}$$

where T_m is the memory size. $s(a,b)$ is the similarity of two shots, which we measure by the SSIM [28] distance of keyframes of the two shots. f_a and f_b are the ratio of shot length to the memory size T_m. Δt is the time difference between two subshots.

4) Problem solution: To solve the optimization problem in (9), constraints are represented by an indicator function $I(s_j^v)$, which is zero when constraints are satisfied. Otherwise, if the camera motion constraint is not satisfied, it is a larger penalty (1,000 in our experiment) to ensure the shot is not selected, except when it is the only choice. The problem is thus formulated as:

$$\mathcal{M}^v = \underset{(s_1^v,\ldots,s_{M^v}^v)}{\operatorname{argmin}} \left\{ \sum_{j=1}^{M^v} \{U(s_j^v) + I(s_j^v)\} + \sum_{j=2}^{M^v} R(s_j^v, s_{j-1}^v) \right\}. \tag{15}$$

The above optimization problem can be defined recursively. Let $f(s_m^v : s_n^v)$ denote the optimal value of the above objective function from cut point m to n (m is not included), as equation (16) shows.

$$\begin{aligned} & f(s_m^v : s_n^v) = \\ & \min_{(s_{m+1}^v,\ldots,s_n^v)} \left\{ R(s_m^v, s_{m+1}^j) + \sum_{j=m+1}^{n} \Big(U(s_j^v) + I(s_j^v) \Big) \right. \\ & \qquad \left. + \sum_{j=m+2}^{n} R(s_{j-1}^v, s_j^v) \right\} \\ & = \min_{s_{m+1}^v} \Big\{ U(s_{m+1}^v) + I(s_{m+1}^v) + R(s_m^v, s_{m+1}^j) \\ & \qquad + f(s_{m+1}^v : s_n^v) \Big\}, \\ & where \qquad 1 \le m \le n \le M^v. \end{aligned} \tag{16}$$

The recursive equation (16) has the optimal substructure property. In other words, to optimize $f(s_m^v : s_n^v)$, we have to optimize $f(s_{m+1}^v : s_n^v)$ for every possible choice of s_{m+1}^v. Taking advantage of the optimal substructure property and the recursive function, we can solve the optimization problem with dynamic programming to approach the global optimization.

We can assume there exists a virtual s_0^v which is unique and satisfies all the constraints. The original optimization problem (15) is then solved by optimizing equation (17) with dynamic programming and the recursive equation (16).

$$\begin{aligned} \mathcal{M}^v &= \underset{(s_1^v,\ldots,s_{M^v}^v)}{\operatorname{argmin}} f(s_0^v : s_{M^v}^v) \\ U(s_0^v) &= I(s_0^v) = 0, R(s_0^v, s_1^v) = 0 \end{aligned} \tag{17}$$

5) Post-processing: As we observed in the focus study, each recording is comprised of many self-contained semantics—subshots. The previous procedures do not incorporate the constraints of such semantic completeness. Selected subshots next to the cut points may be too short to be comprehensible. Besides, shakiness caused by erratic camera motion should be further reduced for better viewing experience.

As a result, we apply two post-processing steps: semantic completeness and video stabilization to the output of video shot selection. For semantic completeness, we require another duration constraint on the subshots next to cut points. Specifically, if a subshot lasts less than one second, we will tune the corresponding cut point a little to satisfy this duration constraint. For stability, we apply video stabilization to alleviate the shakiness in mobile videos [29].

V. Experiments

In this section, we evaluate the performance of the proposed mobile video mashup system. Fig. 6 shows an example of mobile video mashup. Along the timeline are the candidate videos captured by mobile devices. The system selects different shots of these candidates, as the green borders show.

The main goal of the experiments is to evaluate the systems based on three criteria: 1) The first criteria is whether audio mashup improves audio quality. A comparison is conducted with the approach in the *Virtual Director* system [2], which selects the audio fragment of each selected video shot. 2) The second is to evaluate whether the proposed cut point detection works compared with manually labelling and the learning algorithm [5]. 3) The third is whether the proposed video mashup algorithm achieves better viewing experience. The baselines are the two previous mashup systems: *Virtual Director* [2] and *Jiku Director* [5]. Evaluation focuses on video quality, diversity, stability, and overall viewing experience.

A. Dataset

We collect 46 mobile recordings of six events from Youtube[1], which is the largest dataset in existing works. Each recording contains both audio and video streams. These recordings are all captured by non-professionals during concerts using mobile devices. Quality issues mentioned previously are common in the dataset. 14 recordings of the first three events are the same as those used in *Virtual Director*. We use the output videos provided by the authors[2] for comparison. The remaining 32 recordings are submitted to *Jiku Director* for

[1] http://www.youtube.com

[2] Test videos: http://www.youtube.com/AutomaticMashup

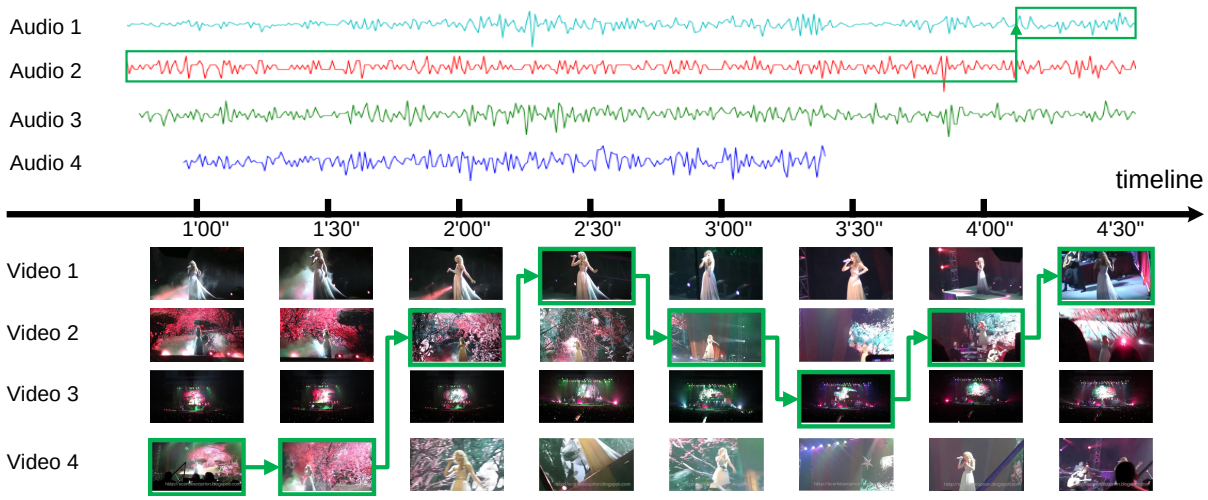


Fig. 6. An example of mobile video mashup. Recordings are captured by mobile devices. Above the timeline is audio mashup. The waves represent the audio energy. Only Audio 2 and Audio 1 are selected for audio mashup. Video mashup is shown below the timeline. Each recording is represented by frames from time 1'00" to 4'30' sampled every 30 seconds. Selection of cameras at these sampling points are represented by the green borders and lines.

TABLE II
DETAILS OF THE DATASET AND THE OPTIMIZATION TIME

Event	#Recordings	Duration	Sync.	Audio Mashup	Cut Point Detection	Video Mashup
E1	5	4'37"	1'44"	0.11"	1.58"	26.19"
E2	5	7'01"	2'38"	0.14"	2.26"	36.26"
E3	4	5'15"	27"	0.11"	1.90"	10.26"
E4	8	6'25"	2'55"	0.10"	2.11"	2'49"
E5	11	3'32"	6'07"	0.11"	1.36"	6'26"
E6	27	5'22"	33'27"	0.19"	1.90"	19'24"

the mashup videos. Table II shows the details of the dataset. We process the recordings in a Windows server which features 8 CPU cores with 3.40GHz speed and 16GB RAM. Averagely, for a one minute audio, it takes 45 seconds to evaluate audio quality. For a one minute video, it takes 240 seconds for video structuring, 65 seconds for motion analysis, and 13 seconds for quality assessment. Optimization time of the six events are reported in Table II.

B. Experimental settings

For audio mashup evaluation, we ask users to listen to pairs of audio recordings generated by the proposed method and the *Virtual Director* approach that stitches audio fragment of the selected video shots. We follow the evaluation scheme of MOS (Mean Opinion Score) [30]. The score choices are: 1 (Bad), 2 (Poor), 3 (Fair), 4 (Good), 5 (Excellent). The methods used to create the audio mashups are not disclosed to users. The order of the two audio mashups in each pair is randomized for different users.

To evaluate the cut point detection, we invite another two professional video editors (different from those in focus study) to assess the cut point suitability, as average users are less sensitive to the quality of cut points. We also ask the editors to give their comments to the cut points. The assessment is around two aspects:

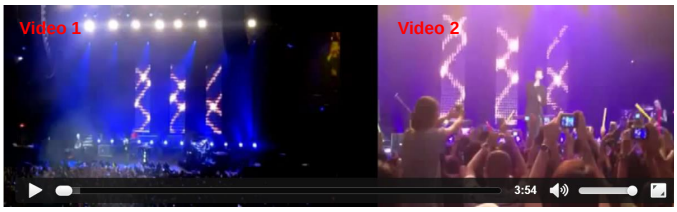


Fig. 7. Video mashup evaluation. Two videos created by two methods are placed on the left and right side respectively.

- Is the switching frequency of the video appropriate?
- Do the cut points appear at the right time?

Mashup videos created by *Virtual Director*, *Jiku Director*, and ours (12 videos in total) are shuffled. The editors are asked to watch them and give scores to the above two questions one by one. The score choices and their meanings are the same to those of audio mashup evaluation, ranging from 1 to 5.

For video mashup, we conduct online user studies to compare the viewing experience. Before the study, an instruction page is shown to guide the evaluation. The page shows four major factors users should consider: diversity, image quality, stability, and overall rating. We ask users a question for each aspect as follows:

- **Diversity**. Does the video give a rich overview of the event? This question determines whether users will be bored with the video due to the monotonous view.
- **Visual Quality**. Is the visual quality of the video good? Visual quality here is mainly about the spatial factors mentioned before.
- **Stability**. Is the video shown stable? We highlight stability since shakiness is a major quality issue in mobile videos, especially in videos captured by amateurs.
- **Overall Rating**. Do you think the video is well edited?

Users are asked to answer the above four questions for each video, with scores ranging from 1 (No, I do not agree at all) to 7 (Yes, I completely agree).

We ask an user experience (UX) designer to help design

TABLE III
NUMBER OF SWITCHING TIMES OF MASHUP AUDIO

Method	Setting 1			Setting 2		
	Audio 1	Audio 2	Audio 3	Audio 4	Audio 5	Audio 6
Mashup	**1**	**2**	**1**	**2**	**1**	**4**
VD	5	5	4	86	23	76

TABLE IV
SUBJECTIVE EVALUATION (MOS) OF AUDIO MASHUP

Method	Setting 1			Setting 2		
	Audio 1	Audio 2	Audio 3	Audio 4	Audio 5	Audio 6
Mashup	**2.56**	**2.28**	**3.56**	**3.00**	**3.81**	**3.0**
VD	2.33	1.28	3.5	1.75	2.81	2.38

the layout of the evaluation web pages. Unlike previous experiments where videos are shown one after another [2], [5], the designer suggests that we show two mashup videos simultaneously. One mashup video is placed on the left side and the other on the right, as Figure 7 shows. Placement of videos is random. Users give scores to both the left and right video, without knowing the methods that create them. The benefits of such a setting include:

- It is easy for users to notice the difference between two videos. In previous settings, users are expected to keep all the information in memory.
- Users will not feel confused when watching two videos simultaneously. Too many videos playing simultaneously will distract users' attention.

According to the above setting, the proposed system is compared with *Virtual Director* on the first three events and *Jiku Director* on the latter three events respectively.

C. Evaluation of audio mashup

In evaluating audio mashup, we conducted two experiments with different settings. In the first experimental setting, we evaluate whether the proposed stitching method works. Due to the *less switching frequency* principle, mashup audio generated by our system switches just a few times (see Table III). Hence we select 30 seconds of the three mashup audio fragments in which switching between audio sources happens often. The counterparts of audio recordings generated by the *Virtual Director* approach are selected.

We invited 18 users, including 14 males and 4 females, to attend the user study. Users' ages range from 22 to 26 years old. Among them, frequency of watching video recordings (concerts, competition, etc.) are distributed as follows: 2 rarely, 3 monthly, 6 weekly, and 7 daily. The results are shown in Table IV (Setting 1). The proposed stitching method is able to alleviate the aural impact caused by switching. The benefit is even obvious in cases where source audio fragments differ much from each other on volume and tone(Audio 2).

In the second setting, we present the whole mashup audio recordings to participants. The purpose of this setting is to find out whether the proposed audio mashup scheme can improve the quality of mashup audio recording. We invited another 15 users, with ages ranging from 20 to 33 years old to evaluate another three pairs of mashup audios. Among the users, there are five females and 11 males. The frequency with which they watch video recordings is distributed as follows: 2 rarely, 1 monthly, 10 weekly, and 3 daily. As we can see from Setting 2 in Table IV, audio fragments generated by the mashup system are much better than those generated by *Virtual Director*.

According to the results of the above two settings, our system generates better mashup audio. The improvement is three-fold: 1) The proposed algorithm selects better audio fragments at each switching; 2) The proposed algorithm reduces switching frequency significantly; 3) The proposed algorithm stitches shots from different sources smoothly to reduce the obtrusive aural effect.

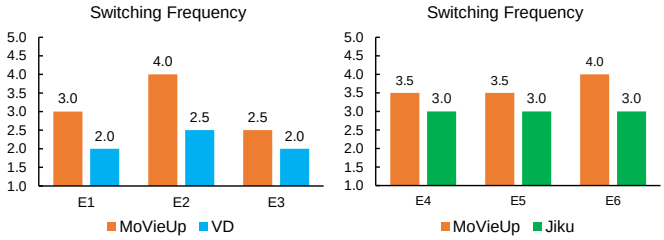


Fig. 8. Evaluation of switching frequency. E1 to E6 are the six events in Table II.

D. Evaluation of Cut Point Detection

In this section, We compare our proposed cut point detection method with *Virtual Director* and *Jiku Director*. Note that cut points in *Virtual Director* is manually annotated by listening to the highest quality audio among all recordings. While *Jiku Director* and *MoviMashup* select automatically.

Fig. 8 shows the comparison result of the three systems on switching frequency. We can easily observe that MoVieUp is much better than *Virtual Director*. This is a little surprising as the cut points in *Virtual Director* are manually annotated. To investigate the reasons, we talked with the editors and found that there are too much meaningless switching in videos generated by *Virtual Director*. Such meaningless switching is often due to some bad effects in video (like occlusion and shakiness). Though our system also selects candidate cut points from audio, it works better on video quality to avoid such meaningless switching. The editors remind us that no switching is even better if no appropriate video is available. *Jiku Director* gets a *Fair* score on all the three events. This is an average score and accords with its non-content-based learning approach. MoVieUp, with consideration of tempo suitability, semantic suitability, shot quality, and motion consistency on cut points, achieves the best performance.

As to the cut point suitability comparison shown in Fig. 9, *Virtual Director* gets poor scores due to the same meaningless switching problem analyzed above. The results verify that cut point selection is dependent on both audio and video. According to the scores, the proposed MoVieUp is slightly better than *Jiku Director*. In our focus study, editors suggest switching videos at speaking/singing intervals. We also find literature that switch at the music beats [17]. The editors themselves even cannot tell a definite rule to select cut points.

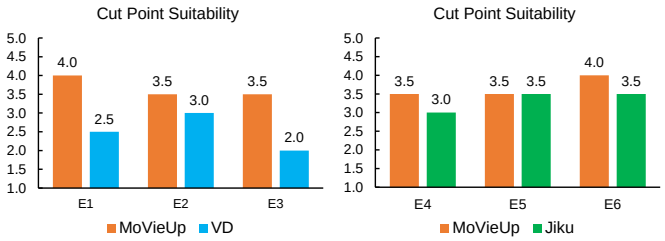


Fig. 9. Evaluation of cut point suitability. E1 to E6 are the six events in Table II.

Even though, the experimental results can still show that MoVieUp provides a practical way to detect cut points as it performs better than *Fair* or even achieves the *Good* level.

E. Evaluation of Video Mashup

We evaluate the viewing experience of our proposed system—MoVieUp, to *Virtual Director* [2] and *Jiku Director* [4]. Since *Jiku Director* is an online system, we mainly focus on the viewing experience of the videos, rather than their application scenarios. Note that we do not apply video stabilization in MoVieUp during the experiments to make a fair comparison with the two existing systems.

1) Comparison with Virtual Director: The 18 users in the first setting of audio mashup attended this evaluation study. Results are shown in Fig. 10.

We find that our method is better than *Virtual Director* on diversity, but the advantage is not as significant as with the other three factors. This is reasonable since both methods select different video sources at each cut point to avoid the problem of monotony. The difference is that we consider temporal issues, compared to *Virtual Director* which measures diversity with only adjacent frames. Additionally, we analyze these videos and find that in each event there are only four to five source videos. They are captured from different viewing angles and distances. This also alleviates the monotony problem.

Our method performs much better on visual quality. *Virtual Director* considers four quality factors: *blockiness, blurriness, brightness*, and *shakiness*. We employ more spatial and temporal video quality factors (*tilting, infidelity*, and *jerkiness*). The pre-filtering step helps to filter out very bad video shots. To further study the reasons, we invited three professional editors to label the shot quality of mashup videos to be "*Good*" or "*Bad*" from the aspects of *Shakiness*, *Darkness*, and *Infidelity* (including those caused by occlusion or polluted by strong lighting). *Tilting* rarely appears in our experiments and *blurriness* is not labeled individually as it is closely related to *Shakiness*. The results are as Table V shows. Among 46 shots selected by *Virtual Director* in the first event, 24 shots are too dark (more than half of the screen are black). Correspondingly, there are only seven dark shots in the 27 shots selected by our system. In the second event, 14 of the 55 shots selected by *Virtual Director* suffer irregular camera motion, resulting in the jerkiness effect observed in the mashup video, while camera motion of only two shots are erratic in our system. We take a further look at the dataset and find that four out of

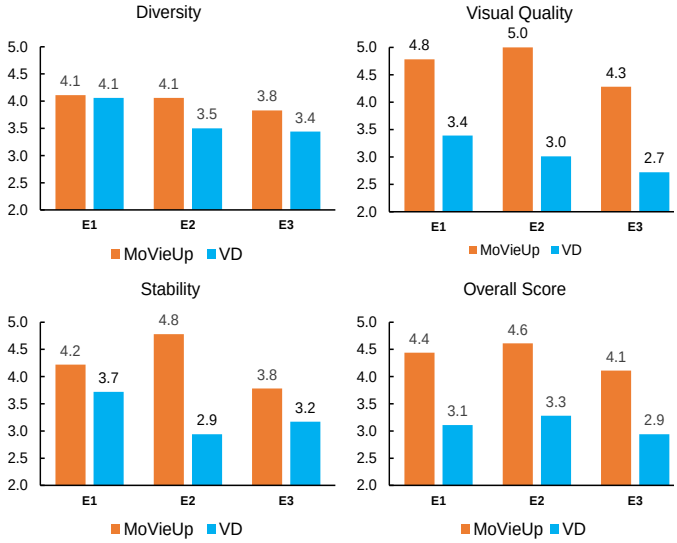


Fig. 10. Comparison of the proposed method with *Virtual Director*. E1, E2, E3 are the three events in Table II. These videos are the same to those used by *Virtual Director* [2].

TABLE V
VISUAL QUALITY COMPARISON OF MOVIEUP AND VIRTUAL DIRECTOR

Factors	MoVieUp			VD		
	Event 1	Event 2	Event 3	Event 1	Event 2	Event 3
Shot	27	27	29	46	55	43
Shaky	0	2	1	2	14	8
Dark	7	0	0	24	1	0
Occluded	0	2	5	0	5	8

the five videos suffer from a shakiness problem. The seven unstable shots are possibly selected for diversity. Besides, infidelity is also a big concern. There are five shots of *Virtual Director* polluted by strong lighting (half of the screen). The number is only two in our system. In the third event, *Virtual Director* selected eight out of 43 unstable shots. In our system, only one shot are unstable. According to the resulted mashup video and user feedback, we concluded that our system is able to select high quality shots from both spatial and temporal aspects.

The proposed mashup system outperforms *Virtual Director* on all three specific factors in all the three events. It verifies the efficacy of the employed video diversity and quality assessment methods. It also shows that the proposed algorithm can achieve a better optimization of diversity, image quality, and thus better viewing experience. This is why users give higher overall scores to all three mashup videos generated by our system.

2) Comparison with Jiku Director: The 15 participants in the second setting of audio mashup attended the comparison study with *Jiku Director*, Evaluation results are shown as Fig. 11.

Our system generates better results than *Jiku Director* on diversity, but like the comparison with *Virtual Director* this is also not by a significant amount. Both MoVieUp and *Jiku Director* employ key frame similarity. The level of interest in

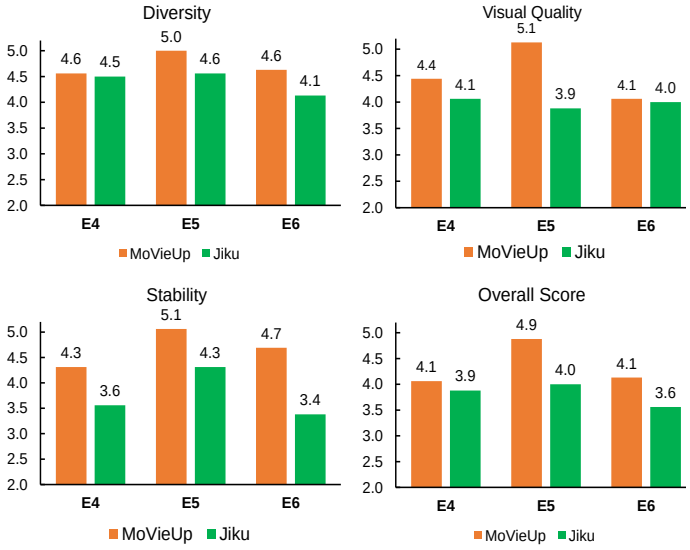


Fig. 11. Comparison of the proposed method with *Jiku Director*. E4, E5, E6 are the three events in Table II. Mashup videos are generated by Jiku Director and MoVieUp respectively.

content for both systems decreases over time. The difference is that *Jiku Director* selects viewpoint (shooting distance and angle) first. Only videos in the selected viewpoint will be considered further. Such a strategy may reduce diversity when consecutive viewpoints are selected close to each other. In contrast to this approach, our method selects shots with a memory model.

For visual quality, the two systems are similar in Event 4 and Event 6. Both systems consider spatial and temporal quality factors. We also evaluate the shot quality as described in comparison with *Virtual Director*. The results are as Table VI shows. Though our system selects fewer unstable shots than *Jiku Director* in Event 4, the occluded shots may be the reason of the comparable scores in Fig. 11. In Event 5, MoVieUp performs much better. We find that the video generated by *Jiku director* is affected by occlusion, strong lighting, and erratic camera motion. As to Event 6, the editors respond that both videos suffer from blurriness. According to Fig. 11 and Table VI, we can still conclude that MoVieUp performs better than *Jiku Director* in terms of visual quality.

The overall rating again shows that the proposed system is able to provide a better viewing experience. It can achieve better diversity and video quality. The stability improves by an especially large margin over previous methods.

F. Video Mashup Example

Due to the fact that the stability of mobile videos is not guaranteed, we apply video stabilization as a post-processing step to the mashup video for better quality. Some examples of the final output of our mashup system are shown online[3]. Generally, viewing experience is further improved compared with non-stabilized videos.

[3]http://www.youtube.com/user/AutoMoVieUp/videos

TABLE VI
VISUAL QUALITY COMPARISON OF MOVIEUP WITH JIKU DIRECTOR

Factors	MoVieUp			Jiku Director		
	Event 4	Event 5	Event 6	Event 4	Event 5	Event 6
Shot	49	21	37	49	26	49
Shaky	3	0	0	11	3	3
Dark	1	0	0	0	0	1
Occluded	5	1	1	7	3	4

VI. CONCLUSION AND DISCUSSION

In this paper, we present a fully automatic mobile video-audio mashup system that works in the cloud to generate both mashup audio and mashup video from *multi-camera recordings* uploaded from various handheld clients. The system achieves viewing experience superior to state-of-the-art video mashup techniques. The system is based on the filming principles concluded from a focus study. To generate high quality mashup audio, we evaluate the quality of the sources and select the best one under the *less switching principle*. We detect video cut points by measuring tempo and semantic suitability from audio. Motion consistency is considered to make smooth switching. To ensure the quality of video, we consider both spatial and temporal quality factors. To enrich video content, we use a memory model to resolve the problem of monotony. Video mashup is formulated as a constrained optimization problem.

One limitation of this work is that we assume that the quality of mobile recordings can be assessed by speech quality assessment algorithms. This may limit the applications in some mobile scenarios. As far as we know, quite few works have been done on non-intrusive quality assessment of general audio. Another limitation of this work is that it cannot reconstruct the 3D position and orientation of the cameras, which hinders the employment of film grammars like the 30 degrees rule, avoiding *Jump Cuts*, and many others about selecting video shots. 3D reconstruction is time consuming and often fails on mobile videos. Sensor-based analysis is a potential solution but requires the encoding of such 3D information along with video frames, which is still unavailable currently.

There are a number of possible improvements for mobile video mashup. Localization of mobile cameras in the captured event [31] will bring in more computational filming principles and help improve video diversity. Besides, non-intrusive objective audio quality assessment is still a rare involved research topic. Visual factors can be incorporated into the detection of cut points approach more natural switching. Another possibility in future work is to explore more semantic information in videos, so that we can take advantage of object motion, camera motion, and many other elements in the film languages to improve the viewing experience. Furthermore, users are now browsing and searching in the internet with multimodal queries [32]. More interactive user experience between clients and clouds is expected by extending the video editing techniques to more general videos from the internet.

ACKNOWLEDGMENT

The work was partially supported by National Natural Science Foundation of China (No.61371192). Ying-Qing Xu was supported by the National Basic Research Program of China (Grant No. 2012CB725300) and the National Natural Science Foundation of China (Grant No. 61373072).

REFERENCES

[1] J. Sang, T. Mei, Y. Xu, C. Zhao, C. Xu, and S. Li, "Interaction design for mobile visual search," *IEEE Transactions on Multimedia*, vol. 15, no. 7, pp. 1665–1676, 2013.

[2] P. Shrestha, P. H. N. de With, H. Weda, M. Barbieri, and E. H. L. Aarts, "Automatic mashup generation from multiple-camera concert recordings," in *ACM Multimedia*, 2010, pp. 541–550.

[3] S. J. Russell and P. Norvig, *Artificial Intelligence — A Modern Approach*. Pearson Education, 2010.

[4] D.-T.-D. Nguyen, M. Saini, V.-T. Nguyen, and W. T. Ooi, "Jiku director: A mobile video mashup system," in *ACM Multimedia*, 2013, pp. 477–478.

[5] M. K. Saini, R. Gadde, S. Yan, and W. T. Ooi, "MoViMash: online mobile video mashup," in *ACM Multimedia*, 2012, pp. 139–148.

[6] X.-S. Hua, L. Lu, and H. Zhang, "Automatic music video generation based on temporal pattern analysis," in *ACM Multimedia*, 2004, pp. 472–475.

[7] I. Arev, H. S. Park, Y. Sheikh, J. K. Hodgins, and A. Shamir, "Automatic editing of footage from multiple social cameras," *ACM Trans. Graph.*, vol. 33, no. 4, p. 81, 2014.

[8] L. Canini, S. Benini, and R. Leonardi, "Interactive video mashup based on emotional identity," in *Proceedings of European Signal Processing Conf*, 2010, pp. 1499–1503.

[9] D. Cardillo, A. Rapp, S. Benini, L. Console, R. Simeoni, E. Guercio, and R. Leonardi, "The art of video mashup: supporting creative users with an innovative and smart application," *Multimedia Tools and Applications*, vol. 53, no. 1, pp. 1–23, 2011.

[10] H. Sundaram, L. Xie, and S.-F. Chang, "A utility framework for the automatic generation of audio-visual skims," in *ACM Multimedia*, 2002, pp. 189–198.

[11] S. Sharff, *The Elements of Cinema: Toward a Theory of Cinesthetic Impact*. Columbia University Press, 1982.

[12] H. Sundaram and S.-F. Chang, "Computable scenes and structures in films," *IEEE Transactions on Multimedia*, vol. 4, no. 4, pp. 482–491, 2002.

[13] F. Lampi, S. Kopf, M. Benz, and W. Effelsberg, "A virtual camera team for lecture recording," *IEEE MultiMedia*, vol. 15, no. 3, pp. 58–61, 2008.

[14] S. Sumec, "Multi camera automatic video editing," in *Computer Vision and Graphics*, 2006, pp. 935–945.

[15] A. Ranjan, R. Henrikson, J. P. Birnholtz, R. Balakrishnan, and D. Lee, "Automatic camera control using unobtrusive vision and audio tracking," in *Graphics Interface*, 2010, pp. 47–54.

[16] C. Zhang, Y. Rui, J. Crawford, and L.-W. He, "An automated end-to-end lecture capture and broadcasting system," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP)*, vol. 4, no. 1, p. 6, 2008.

[17] X.-S. Hua, L. Lu, and H.-J. Zhang, "Optimization-based automated home video editing system," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 5, pp. 572–583, 2004.

[18] F. Manchel, *Film study: an analytical bibliography*. Fairleigh Dickinson Univ Press, 1990, vol. 1.

[19] D. Arijon, *Grammar of the film language*. Focal Press London, 1976.

[20] J.-G. Kim, H. S. Chang, J. Kim, and H.-M. Kim, "Efficient camera motion characterization for mpeg video indexing," in *IEEE International Conference on Multimedia and Expo*, 2000, pp. 1171–1174.

[21] P. Shrestha, M. Barbieri, H. Weda, and D. Sekulovski, "Synchronization of multiple camera videos using audio-visual features," *IEEE Transactions on Multimedia*, vol. 12, no. 1, pp. 79–92, 2010.

[22] W. Liu, T. Mei, and Y. Zhang, "Instant mobile video search with layered audio-video indexing and progressive transmission," *IEEE Transactions on Multimedia*, vol. 16, no. 8, pp. 2242–2255, 2014.

[23] T. Mei, X.-S. Hua, C.-Z. Zhu, H.-Q. Zhou, and S. Li, "Home video visual quality assessment with spatiotemporal factors," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 6, pp. 699–706, 2007.

[24] D. Campbell, E. Jones, and M. Glavin, "Audio quality assessment techniques — a review, and recent developments," *Signal Processing*, vol. 89, no. 8, pp. 1489–1500, 2009.

[25] Z. Li, J.-C. Wang, J. Cai, Z. Duan, H.-M. Wang, and Y. Wang, "Non-reference audio quality assessment for online live music recordings," in *ACM Multimedia*, 2013, pp. 63–72.

[26] A. W. R. an J. G. Beerends, D.-S. Kim, P. Kroon, and O. Ghitza, "Objective assessment of speech and audio quality — technology and applications," *IEEE Transactions on Audio, Speech & Language Processing*, vol. 14, no. 6, pp. 1890–1901, 2006.

[27] D. Stowell, M. Plumbley, and Q. Mary, "Adaptive whitening for improved real-time audio onset detection," in *Proceedings of the International Computer Music Conference*, 2007.

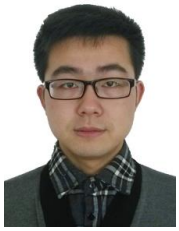
[28] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[29] S. Liu, L. Yuan, P. Tan, and J. Sun, "Bundled camera paths for video stabilization," *ACM Transactions on Graphics (TOG)*, vol. 32, no. 4, p. 78, 2013.

[30] ITU-R Recommendation BS.1284-1, "General methods for the subjective assessmentof sound quality," International Telecommunication Union, Geneva, Switzerland, Tech. Rep., Dec. 2003.

[31] H. Liu, T. Mei, J. Luo, H. Li, and S. Li, "Finding perfect rendezvous on the go: Accurate mobile visual localization and its applications to routing," in *Proceedings of the 20th ACM International Conference on Multimedia*, 2012, pp. 9–18.

[32] Y. Wang, T. Mei, J. Wang, H. Li, and S. Li, "JIGSAW: interactive mobile visual search with multimodal queries," in *Proceedings of the 19th International Conference on Multimedia 2011*, 2011, pp. 73–82.



Yue Wu received his B.E. degree in 2012 from the University of Science and Technology of China (USTC), Hefei, China. He is currently a Ph.D. candidate in the Department of Electronic Engineering (EEIS), USTC. He worked as an intern in Microsoft Research, Beijing, China, from July 2012 to June 2012 and from February 2014 to March 2015, respectively. His research interests include multimedia, computer vision, machine learning, and data mining.

Tao Mei (M'07-SM'11) is a Lead Researcher with Microsoft Research, Beijing, China. He received the B.E. degree in automation and the Ph.D. degree in pattern recognition and intelligent systems from the University of Science and Technology of China, Hefei, China, in 2001 and 2006, respectively. His current research interests include multimedia information retrieval and computer vision. He has authored or co-authored over 100 papers in journals and conferences, 10 book chapters, and edited three books. He holds 11 U.S. granted patents and more than 20 in pending.

Dr. Mei was the recipient of several paper awards from prestigious multimedia journals and conferences, including the IEEE Circuits and Systems Society Circuits and Systems for Video Technology Best Paper Award in 2014, the IEEE Trans. on Multimedia Prize Paper Award in 2013, the Best Student Paper Award at the IEEE VCIP in 2012, and the Best Paper Awards at ACM Multimedia in 2009 and 2007, etc. He received Microsoft Gold Star Award in 2010, and Microsoft Technology Transfer Awards in 2010 and 2012. He is an Associate Editor of IEEE Trans. on Multimedia, Multimedia Systems, and Neurocomputing, a Guest Editor of six international journals. He is the General Co-chair of ICIMCS 2013, the Program Co-chair of IEEE MMSP 2015 and MMM 2013, and the Workshop Co-chair of ICME 2012 and 2014. He is a Senior Member of the IEEE and the ACM.

Ying-Qing Xu is a Cheung Kong Scholar Chair Professor at Tsinghua University, Beijing, China. He received the B.S. degree in mathematics from Jilin University (1982) and the Ph.D. in computer graphics from Chinese Academy of Sciences (1997), separately. His current research interests include human computer interaction, computer graphics, and the e-heritage. Dr. Xu has authored or co-authored over 80 research papers, as well as has had over 20 granted and more pending US patents. He is a member of ACM, ACM SIGGRAPH, and CAA (Chinese Artists Association), a senior Member of IEEE and CCF (China Computer Federation).

Nenghai Yu received the B.S. degree from Nanjing University of Posts and Telecommunications, Nanjing, China, in 1987, the M.E. degree from Tsinghua University, Beijing, China, in 1992, and the Ph.D. degree from University of Science and Technology of China, Hefei, China, in 2004.

He has been on the faculty of the Department of Electronic Engineering and Information Science, University of Science and Technology of China (USTC) since 1992, where he is currently a professor. He is the executive director of the Department of Electronic Engineering and Information Science, and the director of the Information Processing Center at USTC. His research interests include multimedia security, multimedia information retrieval, video processing and information hiding. He has authored or co-authored over 130 papers in journals and international conferences. He has been responsible for many national research projects.

Prof. Yu and his research group won the Excellent Person Award and the Excellent Collectivity Award simultaneously from the National Hi-tech Development Project of China in 2004. He was the co-author of the Best Paper Candidate at ACM Multimedia 2008.

Dr. Shipeng Li joined and helped to found Microsoft Researchs Beijing lab in May 1999. His research interests include multimedia processing, analysis, coding, streaming, networking and communications. From Oct. 1996 to May 1999, Dr. Li was with Sarnoff Corporation. Dr. Li has been actively involved in research and development in broad multimedia areas and international standards. He has authored and co-authored 6 books/book chapters and 280+ referred journal and conference papers. He holds 140+ granted US patents.

Dr. Li received his B.S. and M.S. in Electrical Engineering (EE) from the University of Science and Technology of China (USTC) in 1988 and 1991, respectively. He received his Ph.D. in EE from Lehigh University in 1996. He was a faculty member at USTC in 1991-1992. Dr. Li is a Fellow of IEEE.