AnnoTone: Record-time Audio Watermarking for Context-aware Video Editing

Ryohei Suzuki **Daisuke Sakamoto** **Takeo Igarashi**
Department of Computer Science, The University of Tokyo
rsuzuki@is.s.u-tokyo.ac.jp, {d.sakamoto, takeo}@acm.org

ABSTRACT
We present a video annotation system called "AnnoTone", which can embed various contextual information describing a scene, such as geographical location. Then the system allows the user to edit the video using this contextual information, enabling one to, for example, overlay with map or graphical annotations. AnnoTone converts annotation data into high-frequency audio signals (which are inaudible to the human ear), and then transmits them from a smartphone speaker placed near a video camera. This scheme makes it possible to add annotations using standard video cameras with no requirements for specific equipment other than a smartphone. We designed the audio watermarking protocol using dual-tone multi-frequency signaling, and developed a general-purpose annotation framework including an annotation generator and extractor. We conducted a series of performance tests to understand the reliability and the quality of the watermarking method. We then created several examples of video-editing applications using annotations to demonstrate the usefulness of Annotone, including an After Effects plug-in.

Author Keywords
Video annotation; audio watermarking; video editing interface; context awareness; metadata

ACM Classification Keywords
H.5.2. Information Interfaces and Presentation (e.g., HCI): User Interfaces

INTRODUCTION
In recent years, shooting video footage has become an everyday activity for a large number of people, and the purpose and style of shooting has diversified significantly. For example, various kinds of lecture videos on topics from university-level mathematics to household machine repairs and cooking have been created and shared. Many events, shows, and live concerts are recorded and stored in online archives to provide people all over the world with entertainment. The appearance of compact, low-cost, and high-performance video cameras and smartphones has led to the further widespread use of video recording. Small-sized action cameras such as GoPro are frequently used for recording first-person view of the players of extreme sports such as skydiving and motocross to provide exciting movie contents. These recorded videos may be edited and uploaded to video-hosting services such as YouTube and Vimeo. The recording and sharing of video is an entertainment experience that is available to a great number of end users.

This new form of entertainment has led to the desire to rapidly create attractive movie content from recorded videos, and the demands of non-professional video-editing applications have grown considerably. One problem here is that it is often time-consuming to edit the recorded video, particularly when the editing is dependent on contextual information that is only available during recording. For example, if the user wishes to overlay a map showing the geographical location of the current video frame, they must separately take notes and manually combine this with the video. Even if the needed contextual information can be obtained by watching the video, it is tedious to repeatedly review the video to complete the editing. Our goal in this study was to facilitate such an editing process by allowing the user to embed contextual information directly into the video while it is being recorded, and to use the embedded information to support editing the video after recording.

In this paper, we describe a system called "AnnoTone", which allows the user to embed annotations into video footage during real-time recording as audio watermarks, and to later extract them from the annotated videos during editing. The user simply places a smartphone on top of a video camera, and the watermarking signal is transmitted from the loudspeaker of the smartphone and recorded as inaudible background audio, as shown in Figure 1. The annotation data can be either manually specified by controlling the smartphone, or automatically generated based on sensor data, which may be a global positioning system (GPS) receiver or an external sensor. Our technique can also be used while recording a video with a smartphone. By running a background application that transmits the watermark signals from the loudspeaker on a smartphone, the user may use any camera application on the phone to record a video with embedded annotations.

We evaluated the performance of the watermarking scheme, including limitations regarding the data rate, the distance that the watermark signal can travel in air, the response to audio format conversions, and the influence of the signal on the sound quality of the audio track in the finished video. The results showed that AnnoTone can embed data into a video at a

CHI 2015, April 17–23, 2015, Seoul, Republic of Korea.

http://dx.doi.org/10.1145/2702123.2702358

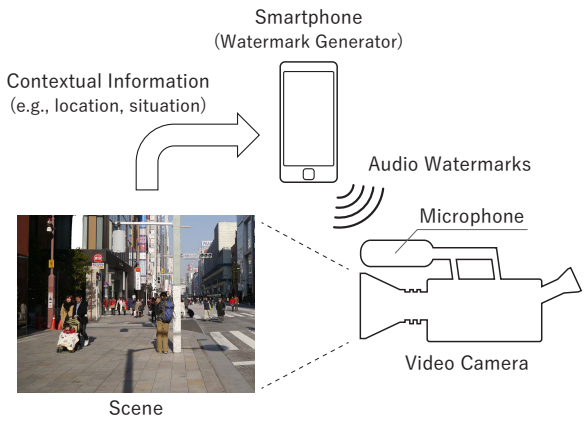

Figure 1. An image showing how annotations are recorded.

bit rate of 400 bps, with adequate reliability in various recording situations, and that the embedded annotations are robust to conversion between several major audio formats. The embedded watermarks do not lead to noticeable loss of listening impression, and they can be removed completely by applying a low-pass filter to the audio signal in the finished recording.

We also investigated several examples of video-editing applications using annotations to demonstrate the possible usages scenarios of AnnoTone. The first application enables automatic segmentation of a recorded video by splitting it into shorter movie clips, which are classified as either success or failure at the positions annotated by the user. The second application utilizes annotations that describe the states of a chess game at each moment to automatically overlay a video with generated graphics to help the watcher understand the game. The third application uses a series of annotations indicating the geo-locations of a moving camcorder to show the path of the camera on a map. These applications demonstrate the potential of annotations in movie editing to reduce the editing workload, and to help create richer content. We also designed a way to integrate the watermark extractor with off-the-shelf video editing programs, and confirmed that AnnoTone can facilitate existing movie-creation workflow by, for example, adding effects automatically based on annotations.

The major contributions of this work can be summarized as follows:

- The creation of a novel video-editing workflow with real-time annotation using audio watermarking.
- Implementation of a prototype system using Dual-Tone Multi-Frequency (DTMF) method.
- Performance evaluation exploring the capabilities and limitations of the method.
- Three example applications that facilitate the AnnoTone and a script to export annotation to video editing software.

RELATED WORK

Digital Audio Watermarking

Digital audio watermarking techniques have been studied mainly for protecting music and movies from piracy by including copyright information in the content. Audio watermarking was first developed in the late 1990s. In 1996, Bender et al. described the basic techniques, including low-bit coding, phase coding, spread spectrum, and echo hiding [5]. Following their pioneering work, many researchers have reported enhanced audio watermarking techniques. An advanced spread spectrum technique, with improved robustness and detection capabilities, which employs a large range of frequencies was reported by Cox et al. [8, 7].

Most watermarking schemes embed audio watermarks into existing audio or video content by analyzing and manipulating the audio signals, so it is not usually possible to add watermarks during real-time recording. Recently, technologies termed *real-time watermarking*, which embed watermarks into audio signals as they are recorded, have been developed. Tachibana described a real-time watermarking scheme named Sonic Watermarking, for live musical performances [23], the aim of which was to prevent surreptitious recording of concerts. In this scheme, the musical performance and watermark signals are played separately from two speakers, and mixed in the air; therefore, audio watermarking is realized without manipulation of the recorded audio signals of the performance. Our method uses a similar technique to mix audio watermarks and environmental sounds in the air to annotate video data without requiring any modifications of the recording instruments.

Audio watermarking techniques have also been used to transmit information to devices to enhance the user experience in entertainment and advertising applications. Matsuoka reported a watermarking framework termed "acoustic orthogonal frequency division multiplexing" (acoustic OFDM) [17] to convey text data, such as a uniform resource locator (URL) to consumer devices during audio broadcasting. Listeners of an annotated audio program can retrieve related information such as song titles using a dedicated smartphone application that can receive watermark signals. Museum installations and live concerts have demand for such communication because they can enhance the interactivity of exhibitions and performances without installing additional devices except for an ordinary loudspeaker and smartphones. Gebbensleben et al. developed an audio guide system for museums and exhibitions that automatically detects objects which the user is seeing using audio watermarking signals, and plays an appropriate audio content [9]. Hirabayashi developed a system called Cryptone [14], which enables interaction between performers and the audience at venues of musical performances using high-frequency acoustic DTMF signaling, in a similar manner to our method; however, this system can only represent a limited number of identifiers and cannot transmit arbitrary information. In contrast, our method, AnnoTone, focuses on a rich array of information, including geo-location data and contextual information. In addition, whereas existing methods use audio watermarking to facilitate interactions between devices, we use it to facilitate video editing.

Video Annotation

As the advances in camera devices have led the proliferation of digital photographs and videos, audio/video annotation techniques to facilitate collecting, browsing and editing

recorded contents have been proposed. WillCam [24] is a digital camera system that captures and records contextual information including the location, temperature, ambient noise, and facial expressions of the photographer, and uses these data to provide the user with a means to indicate what is relevant to them in the picture. It simply overlays the pictures with visual icons representing associated contextual information, and simultaneously stores sensor data in extended image format (EXIF). ContextCam [19] is a context-aware video camera for creating archives of home movies, which annotates it with the time, location, people present, and additional contextual information associated with the video, using a collection of sensors and machine-learning techniques to infer higher-level information. With ContextCam, annotations are embedded in the recorded video sequence using least significant bit (LSB) encoding on video frames. These systems enable the recording of contextual information during the shooting of a picture or a video footage, as with our method; however, because they both require specialized hardware, the range of applications of these techniques is limited. However much work also have been done for automated or semi-automated photograph/video annotation [15, 20, 26], there is still little work in intensively utilizing these annotations for editing videos. AnnoTone focuses on providing a novel video-editing workflow with recorded annotations for supporting content creation.

Some commercial products that can record contextual information with video data are already commercially available. Sony's HDR-AS30V/B Action Cam [1] is a compact digital video camera equipped with a GPS receiver to record ones location during the shooting of video footage. The videos recorded by it can be played with overlaid information, including the speed at each moment, together with a map showing the path taken when using a special player. Our method makes it possible to add such functionalities to ordinary video cameras and smartphones, without requiring any special hardware.

Managing Contents with Metadata

Numerous user interfaces have been developed to utilize time-synchronized metadata or annotation to browse, analyze and arrange various kinds of contents including videos. DIVA developed by Mackay and Beaudouin-Lafon [16] is an explaratory data analysis tool for videos with various kinds of annotations, designed for analyzing the work practice of air traffic controllers, and discovering an opportunity for improving their user interface. DIVA provides a sophisticated mechanism to handle multiple streams of annotations, such as boolean operation between streams, and a display interface in which the user can look over the correlations between annotations from above, to facilitate the video analysis task.

SILVER [6] is a digital video editor that simplifies the process of video editing by incorporating semantic annotations about videos into the editing interface. It provides a hierarchical timeline view that enables the user to handle videos by multiple levels of semantics, from frame level to shot level to clip level. It also provides a transcript view in which the user can specify a chunk of video based on the selection of transcript. Using these interfaces fully utilizing annotations, the user can easily edit a video concentrating on the semantics of video, without being bothered by too much low-level operations. As these systems suggested, handling large data such as a video could be significantly simplified by using appropriate annotations and user interfaces. Annotations recorded with videos by AnnoTone could be used from such systems to facilitate managing contents.

ANNOTONE

Workflow

Figure 2 shows the workflow of AnnoTone. We designed the annotation method to convert annotation data into sounds and to transmit them from the loudspeaker of a smartphone. We take advantage of the fact that ordinary video cameras have 44.1 kHz audio sampling frequency and can record sound with frequency up to about 22 kHz, though high-frequency sound above 18 kHz is virtually inaudible for human if it is not too loud [4, 22]. Annotations are represented as audio watermarks, are embedded in the high-frequency range of the audio track of a video recording, and can be extracted after the video is imported into a personal computer (PC) for use to support video editing. Because annotations are embedded as sounds that can be recorded using an ordinary microphone, our technique does not require any special equipment other than a video camera and a smartphone. Audio watermarking also has advantages in terms of the portability and synchronicity between the annotation and a timestamp of a video, because annotations are embedded directly into the time series of the content. Unlike annotation methods that employ external metadata, all of the information is contained in the video file, and synchronization between a moment of a video and a watermark is not lost during video editing. Because AnnoTone uses a specially designed watermarking scheme that employs high-frequency audio (in the range 17.6 - 20.0 kHz), the watermarks are not only inaudible, but also can be easily removed from the video recording by applying a digital filter.

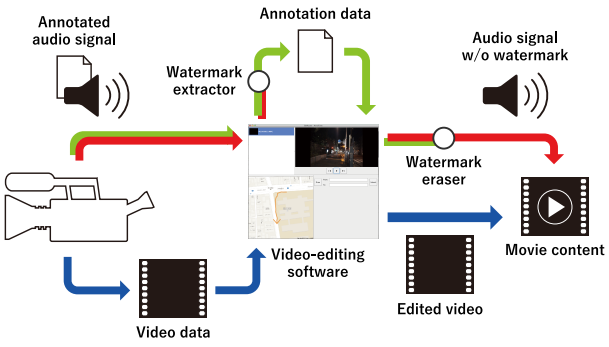

Figure 2. The AnnoTone workflow.

Usage Scenario

The basic usage scenario of AnnoTone is as follows. Users first attach a smartphone to a video camera and then start an application that includes the watermark generator, as shown

in Figure 3. Next, they record a scene using the video camera, controlling the smartphone application to embed the annotations they need. After recording the video, the video file is transferred to a PC, and loaded into an annotation-aware video editing application using the watermark extractor. And, then users can create movie content from the imported video, exploiting the functionality provided by the application. Following editing, the watermark is removed using a filter, so that the high-frequency components are not included in the resulting movie. In addition to the a dedicated video editing application, we provide an extensional script to extract and erase the audio watermark, and export to a commonly used video editing application.

The other scenario is that public event. Instead of attaching a smartphone to a video camera, we can share the loud speaker for embedding the audio watermark into not only the video camera, but the other camera phones and smartphones. This is an optional scenario and is not main target usage scenario of the AnnoTone. However, this will be included in the performance evaluation for investigating future implication of the AnnoTone.

Figure 3. A typical hardware setup to use AnnoTone.

IMPLEMENTATION

The AnnoTone system consists of a watermark generator, extractor, and eraser. The watermark generator is provided as a software library for smartphones. It receives digital data from either the user interface of the application or the sensors on the smartphone, converts them into audio watermark signals, and then transmits the signals from the loudspeaker on the phone. The user should prepare a smartphone application depending on their purpose for the video annotation. The watermark extractor is implemented as a Java program that analyzes video files and extracts the embedded annotation data for use to support video editing. It can be used as a component of a video editing application that provides dedicated user interface and functionalities for specific annotation types, and also can be integrated to existing video editing applications. The watermark eraser is a low-pass filter that removes the high-frequency audio signals (i.e., at frequencies $\geq$ 17 kHz) to eliminate the audio watermarks from the audio track of the finished video.

Watermarking Scheme

The watermarking scheme used with AnnoTone can be considered a derivative of dual-tone multi frequency (DTMF) signaling [21], which is widely used in telecommunications applications. The basics of the DTMF audio watermarking technique can be found in previous works [11, 14, 18]; however, we describe them here briefly for completeness. AnnoTone converts annotation data into modulated sounds that are almost inaudible, at frequencies in the range 17.6 - 20 kHz [4, 22]. A unit signal in DTMF is a composition of two single tones of different frequencies, and the combination of the two frequencies represents the value of the signal. AnnoTone uses seven equally spaced frequencies, and we use 16 combinations of pairs of these frequencies (${}_7C_2 = 21$) to represent four bits per signal. Larger datasets can be sent by rapidly switching the tones. The length of each unit signal was set to 10 ms, giving a gross aggregate bit rate of 400 bps. Figure 4 shows a signal spectrogram of a watermark packet.

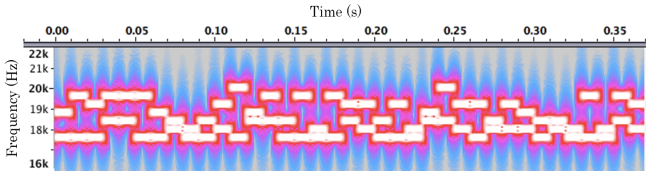

Figure 4. A signal spectrogram of a watermark packet with a 16-byte payload.

Packet Structure

Annotation data are serialized into an array and converted into a structured watermark packet. Figure 5 shows a schematic diagram of the packet structure. Each watermark packet consists of several successive unit signals. The unit signal at the head of a packet is termed the start frame. It is fixed to represent "2", and functions only as an indication of the start position of the packet. The second unit signal is termed the length frame, and specifies the length of the payload region following it to be 2^n bytes, where n is the value of this unit signal. This enables the payload size to vary depending on the annotation. From the third unit signal, the payload region containing the body of the annotation data starts, and continues for the length specified in the length frame. Each byte is encoded using two unit signals: the first represents the lower four bits and the second represents the higher four bits. The unit signal following the end of the payload region represents the annotation type ID, and is a number from 0 to 15, which is used to distinguish the different kinds of annotations, which may be embedded in the same media file. The annotation ID can be determined arbitrarily by the annotation application. Each packet ends with two unit signals representing the cyclic redundancy check (CRC)-8 checksum of the packet for error detection in decoding.

Figure 5. The packet structure used with AnnoTone.

Watermark Generation and Extraction

DTMF watermarks are generated using the following process. First, an annotation of any kind of data are serialized to an array of bytes. Second, a data stream of packets is generated from the array by adding header and footer data, including the CRC-8 checksum. Third, the DTMF pulse-code modulation (PCM) signal is generated as follows:

$$s(t,m) = w(t) \cdot a(m, d(\lfloor \frac{t}{0.01} \rfloor)) \cdot \sin(2\pi t \cdot f(m)) \quad (1)$$

where $s(t,m)$ is the sample value of the m-th sub-carrier at timestamp t (in seconds) counted from the beginning of the watermark packet, $f(m)$ is the frequency (in Hz) of the m-th sub-carrier, $d(n)$ is the n-th 4-bit unit of data in the packet, and $a(m,x)$ is the DTMF encoding function, defined as the (m,x)-th element in the following table.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
7	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0
6	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0
5	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	1
4	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0	1
3	0	1	0	0	0	0	1	0	0	0	0	1	1	1	1	0
2	1	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0
1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0

Table 1. The DTMF encoding function.

$w(t)$ is an envelope filtering function to reduce low-frequency noise at borders between two unit signals:

$$w(t) = (0.5 - 0.5\cos(2\pi \frac{t \bmod 0.01}{0.01}))^{0.8} \quad (2)$$

The watermark packet $x(t)$ is generated by mixing all of the sub-carrier signals as follows,

$$x(t) = \sum_{m=1}^{7} s(t,m) \quad (3)$$

Because DTMF modulation can be considered a variety of frequency-shift keying (FSK), the watermarks that are embedded in media files using AnnoTone can be easily extracted using common demodulation methods for FSK. Watermarks can be removed by simply applying a low-pass filter, because they are contained in a frequency range that is at the upper limit of the frequency response of the human auditory system, and so is not significant for the listening quality. In the performance evaluation described below, we used a 199-tap finite impulse response (FIR) low-pass filter with a cut-off frequency of 17 kHz to remove the watermarks.

PERFORMANCE EVALUATION

To demonstrate the feasibility of the proposed technique, we conducted a series of performance evaluations of our audio watermarking system. These consisted of three objective evaluations to investigate the robustness of the scheme in practical situations, as well as a field test to show that our method can be used without negatively impacting the sound quality of the final product. In these experiments, we used a typical hardware setup consisting of commercially available consumer products, i.e., a Sony NEX-VG30H digital camcorder and a Samsung Galaxy S smartphone. With the exception of the evaluation of the acceptable distance between camera and smartphone, the smartphone was directly attached to the microphone unit of the camera using a rubber band, so that the distance between the loudspeaker of the smartphone and the microphone was approximately 1 cm, as shown in Figure 3. We found that some other video cameras produce video files with 44.1 kHz or 48 kHz audio track, but actually record sounds in much lower sampling rates (e.g. 32 kHz). AnnoTone cannot be used with such video cameras.

Video data were recorded in MPEG-2 format using standard image quality, and sound was recorded using a sampling rate of 44.1 kHz and 16-bit stereo PCM format. The annotation transmitter application was installed on the smartphone. The volume of the smartphone loudspeaker was manually adjusted to the maximum value that did not result in audible distortion. Videos may be recorded in diverse situations, and the acoustic environment may vary considerably depending on the situation in which a video footage is shot. Taking this into consideration, we did recording experiments in four different environmental settings: a silent room, a public space, a rock music concert, and while playing electronic music.

Limitations of Data Rate

To achieve a high data rate, the amount of data sent per unit time must be large; however, shorter lengths of data packets decrease the correct detection rate (CDR), and so there is an optimum data rate that retains a large CDR, together with a large overall data rate. To determine this optimum and show that our watermarking scheme can transmit practicable amounts of data with good reliability, we recorded video footage with embed watermarks with a range of unit signal lengths, and measured the CDR. We varied the unit signal length in the range 6 - 11 ms at 1 ms intervals. A unit signal length of 6 ms corresponds to a data rate of 667 bps, and a unit single length of 11 ms corresponds to 364 bps. For each recording, 100 watermarks with 16-bytes payload were embedded at 3 s intervals, and the correctly detected watermarks were counted using our watermark extractor after recording.

Figure 6 shows the measured CDRs for each of the environmental conditions. This result shows that sufficient reliability was achieved with all of the recording environments when the unit signal length was set to 10 ms (98 % for the electronic music environment, and 100 % for the others). The CDR dropped rapidly as unit signal length decreased below 10 ms in all acoustic environments. From the result, we conclude that 10 ms is the optimum length of the unit signal, and 50 bytes of data can be sent per second. For comparison, the speed of speaking that the range of people can hear comfortably is 150 - 160 words per minute [25], which corresponds to approximately 12 - 13 bytes per second (assuming an average word length of five characters); therefore, 400 bps is sufficient to embed subtitles of a conversation, for example.

Separation of the Camera and Smartphone

In some situations, it may not be convenient to attach the smartphone directly to the video camera, and is desirable that

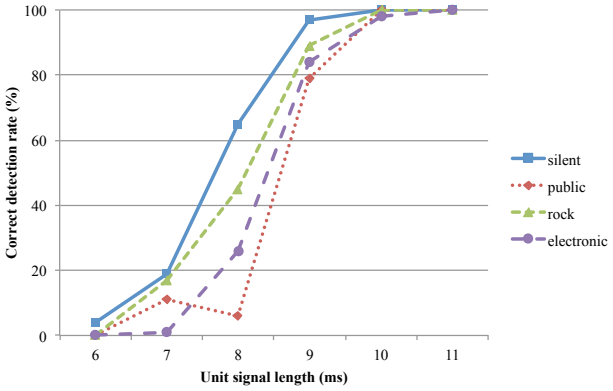

Figure 6. The CDR as a function of the unit signal length.

watermarks are transmitted from a distant device like loud speaker, for example. To investigate the possibility of such usage, we measured the CDR as a function of the distance between the camera and the smartphone.

Figure 7 shows the CDR as a function of the separation between the smartphone and the video camera. A CDR of greater than 80% was obtained when the distance was less than 25 cm. This suggests that watermarking can be achieved without directly attaching a smartphone to the video camera, and annotation is available in various usage scenarios, including, for example, when the camera operator transmits the watermarks from a smartphone in one hand while holding the camcorder in the other hand. However, this distance may be too short for many situations, and so increasing the sensitivity of the watermark detection system is desirable. Improvements in the response of the loudspeaker of the smartphone and the microphone of the video camera at high frequencie range may be expected to enable an increase in this distance.

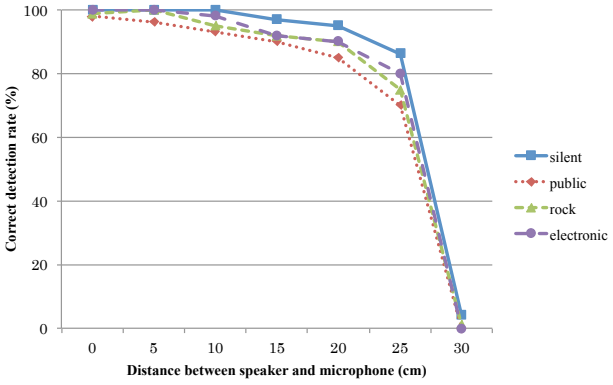

Figure 7. The CDR as a function of the distance between the loudspeaker of the smartphone and the microphone of the camera.

In order to investigate the practicability of diverse use cases, we also conducted a similar experiment using a desktop speaker (28 W Creative GIGAWORKS T20 II) instead of a smartphone, and examined how much distance watermark signals can travel through air if the output volume is much larger than a smartphone. Because of the spatial restriction, we only used a silent room (10 m $\times$ 10 m, 3 m height, concrete wall and floor) as the acoustic environment for this experiment. Figure 8 shows that a CDR greater than 80% was obtained when the distance was less than or equal to 5 m. It suggests that we can record watermark signals into a video camera transmitted from a loudspeaker placed remotely. We give some possible applications with such usage in the discussion section.

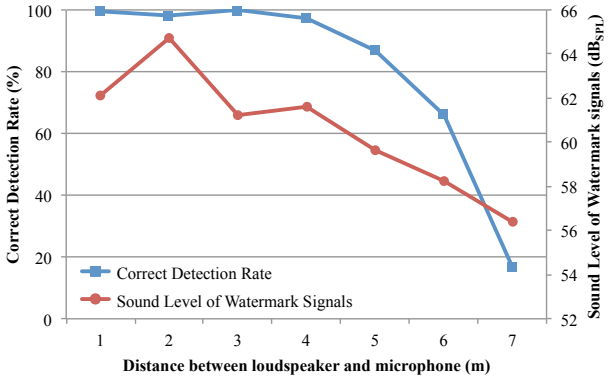

Figure 8. The CDR and the measured sound level of watermark signals at the camera position as functions of the distance between a desktop speaker and the microphone of the camera.

Audio Format Conversion

To assure that the embedded annotations remain throughout the movie-authoring process, it is important to retain the watermarks even when format conversions are conducted. We evaluated the durability of watermarks in regard to format conversions by measuring the proportion of detectable watermarks in videos after applying a variety of audio format conversions. An uncompressed audio file with 10 detectable watermarks was converted into a compressed format, and watermark detection was investigated on the converted file. MP3, Ogg Vorbis, AC-3 (also known as *Dolby Digital*) and AAC formats were used as compression algorithms. For each format, compression was applied with four bit rate settings in the range 128 - 320 kbps to examine the quality requirements necessary to preserve the watermarks with each format. We measured the CDRs of different audio files and calculated the mean CDR for each format and quality setting.

The results are shown in Figure 9; watermarks were preserved almost completely following a format conversion with the Ogg Vorbis and AC-3 formats with a bit rate of $\geq$ 192 kbps. They were also preserved after conversion into AAC format with a bit rate of 320 kbps. This suggests that our method can be used in practical movie-authoring processes because satisfying these format requirements is not difficult. However, the MP3 format did not preserve the watermarks, even at the highest bit rate setting; this is attributed to the characteristics of the compression algorithm, which tends to discard high-frequency components.

Effect of the Watermarks on Audio Quality

We evaluated the audibility of the watermarks via a subjective experiment. Five participants, aged 21, 22, 22, 58, and

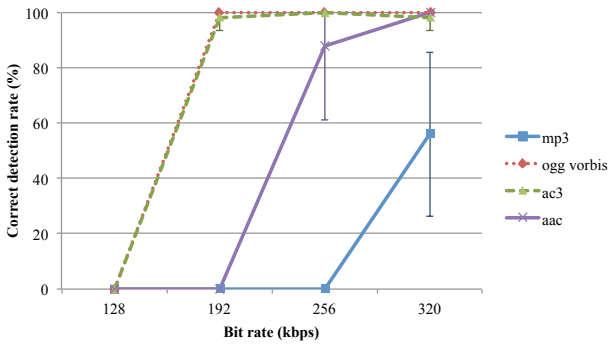

Figure 9. The average CDR following format conversion for a range of compressions.

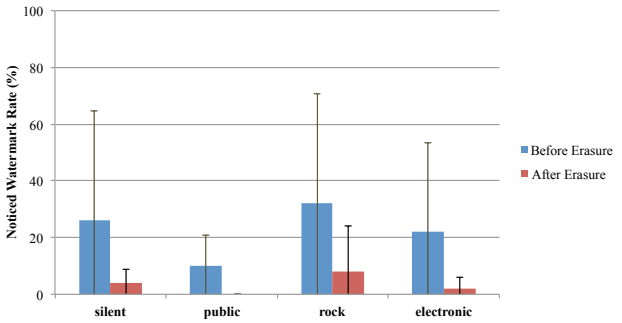

Figure 10. Mean rates of identification of watermarks by listeners for each of the recording conditions.

58 years, were asked to listen to recorded sounds with the embedded watermarks using headphones, and to push a button if they noticed any unexpected or unnatural sounds. The output volume was maintained constant at a volume that was comfortable for the participant to listen to music. The proportion of noticed watermarks was counted for each participant and sound. We investigated whether the watermarks were audible with the four sound settings described above (i.e., silent, public, electronic music, and rock music) to investigate whether the detectability of the watermarks was dependent on the environment of the recording. To verify that the low-pass filter can effectively remove the watermarks from the audio file, we prepared two recordings for each sound setting: one with the filter applied and the other without the filter. In total, eight recordings were used in the experiment. Each recording was 1 min long, and contained 10 watermarks with 16-byte payloads, at randomly selected positions.

The results are shown in Figure 10. The mean rate at which watermarks were detectable by the participants was 30 % prior to applying the low-pass filter in each situation. After applying the filters, the rates at which participants correctly detected the watermarks were negligible, indicating that the watermarks were effectively removed using the filter. Furthermore, the difference in the rate at which participants noticed the watermarks depended on their age, with elder participants being less able to notice watermarks than younger ones. The absolute threshold of hearing is reported to be a sound pressure level (SPL) of 130 dB at 18 kHz in people aged 50 - 59 years, compared to an SPL of 80 dB in people aged 10 - 19 years [22]. Our results are consistent with this age dependence of the frequency response of the human ear.

APPLICATIONS

In this section, we describe three example applications using AnnoTone. Each application is provided as a set of watermark-generating programs running on the camera-mounted smartphone, and a watermark-extracting program running on the PC. Here, we will describe the input to the watermark-generating program and the output of the watermark-extracting program. After that, we show the interoperability of AnnoTone with an off-the-shelf video-editing software, Adobe After Effects.

Record-time Editing

During post-production, a movie creator may carry out a large number of operations, such as deleting scenes, which can be tedious and time-consuming because the user must repeatedly examine the video content. Using AnnoTone, we can embed information required for the editing process into a video during recording, as well as perform automatic editing after the video has been recorded, using that information. We have created a simple video-editing application, which automatically cuts sections containing mistakes from a video, as shown in the right image in Figure 11.

Suppose the user shoots a video lecture, consisting of multiple independent segments. If the lecturer makes a mistake during recording, the segment should be deleted and a new segment needs to be recorded. To facilitate this process, our system allows the camera operator to indicate success or failure of the segment by pressing a button on the smartphone just after recording that segment (Figure 11 left). This success/failure information is recorded as an annotation. After recording, the video-editing application automatically detects the annotations and divides the video into a series of segments according to the annotated timestamps. Then the system automatically removes all of the segments that are marked as failures.

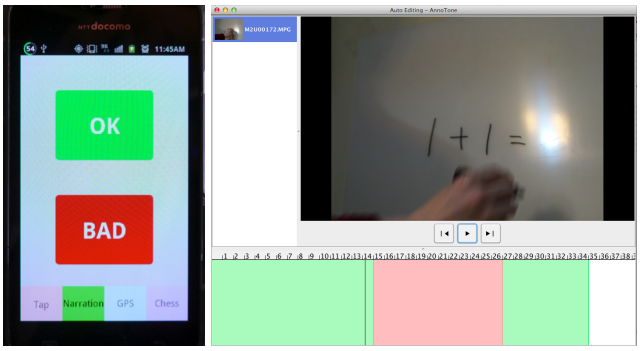

Figure 11. Left: The user interface of the annotation application. Right: The automatic video-cutting application; the good sections are shown in green, the bad sections in red.

Automatic Captions

Much movie content has overlaid captions or images that describe situations and provide supplementary information.

For example, recordings of ballgames typically have a caption containing the score, some television programs related to travel have overlaid maps, and movies of concerts may have captions containing the words of the songs. Creating such captions and overlaying them at the appropriate time is a time-consuming part of the movie-authoring process. In some cases, creating and overlaying captions can be automated by embedding contextual information in videos using AnnoTone.

We developed a system for chess movies, which automatically overlays a video with an animation of a chessboard in sychronization with the game progress using embedded chess notes. The camera operator operates the smartphone application to embed the chess notes in the video. The application provides a simple videogame-style user interface, as shown in the left image in Figure 12. Each chess note is recorded as a line of standard Forsyth-Edwards notation (FEN); therefore, the notes embedded in a video represent all of the information describing the game. The system analyzes the video annotated with the chess notes to determine the state of the chessboard at each timestamp, and generates a series of images to be overlaid as shown in the right image in Figure 12.

Figure 12. Left: The user interface for annotating chess notes. Right: A movie overlaid with a chessboard created by the system.

Geo-location-based Video

Sometimes a video is recorded while the camera operator walks or rides in a vehicle. In such cases, the location at each moment is often significant to the content of the movie. We created a system that can automatically overlay a video showing a corresponding map together with the trail of the camera. The watermarking application embeds the geographical location of the video camera using data from a GPS sensor at intervals of a few seconds. The video-editing application extracts the embedded series of geo-locations associated with the timestamps of the video to obtain a map using the Google Maps API [3], as shown in Figure 13.

Integration with Existing Video-editing Software

So far, we showed three example applications that facilitate the audio watermarking technique. These applications are designed as dedicated video-editing software to provide special functionalities. However, AnnoTone is not limited to this usage, and can be used with off-the-shelf video-editing software to facilitate content creation workflow in general. Some video-editing programs provide end-user scripting mechanisms that allow the user to control effects and objects in the movie by writing a simple program. AnnoTone's annotation data can be used from these mechanisms by implementing an extension or a plugin for the video-editing software that executes watermark extractor and imports extracted data into it. We developed an extension script for Adobe After Effects (AE), a widely used video editing program, to integrate AnnoTone's functionality into it. We employ the end-user scripting mechanism (*expressions*) of AE. The process is simple; just running the extension script once to extract the audio watermark. As a result, a new text layer is generated, and each embedded watermark packet is converted into a key frame of the layer at the corresponding timestamp in the source video. Each key frame contains annotation data represented in the JavaScript Object Notation (JSON) format, which can be accessed from the end-user scripts. User can control many aspects of the behavior of objects and effects in the video using the imported annotation data. For example, users can control the pulse rate of a computer animation of a heart that is overlaid on a sport video, in synchronization with the sequence of output values of a vital sensor that is embedded in the video as annotations (Figure 14).

Figure 13. A video overlaid with a map showing the trail of the camera.

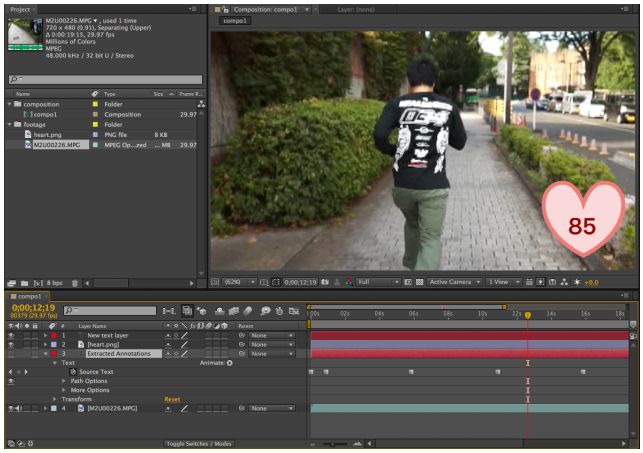

Figure 14. Editing a video of jogger with annotations of his heart rate at each moment in Adobe After Effects. The extension script automatically generates a text layer including keyframes representing the heart rate at each moment as a JSON object. The animation and the number at the lower-right of the movie is associated with the data by end-user scripts.

DISCUSSION

Possible Extensions and Applications

As an extension of AnnoTone, it is possible to change the source of the watermark sound to broaden the use of the technique. As a result of the performance evaluation, we confirmed that audio watermarking from distant place by using 28 watt desktop speaker was successfully embedded information into a video camera. This result suggests that the distance between speakers and a video camera can be longer with a public space-size loud speaker. For instance, using a loud speaker placed in a public space to transmit watermarks containing location-specific data that could allow videos recorded in that space to be used by anyone. Such an installation could offer various kinds of entertainment associated with video recoding, for example, videos recorded at rides and attractions of an amusement park by visitors could be automatically placed on a map after being uploaded to a video-hosting service, and visitors who recorded videos at a specific location could receive promotional material from the park. Publicly transmitted watermarks could also be used to synchronize videos recorded at a sports match or a live performance, and to generate multi-viewpoint video content. According to studies in the past, some kinds of animals such as dogs and cats are sensitive to high-frequency tone [12, 13], therefore publicly transmitted audio watermark signals possibly affect them. Thus, installation of public speakers for transmitting watermark signals should be avoided in places where pets are expected to be brought, such as parks.

Using machine learning or pattern-recognition techniques with the watermark-transmitting program for smartphones could provide more useful applications. For example, annotating an emoticon describing the mood of the camera operator could aid with locating content in a large home video archive [24]. Another possible application of AnnoTone is similar to the Accessible photo album program [10], i.e., supporting people with visual impairments in sharing their memories by recording videos with associated non-visual information that aids in recalling the experiences, such as temperature, textual information describing the scene, and the subjects or camera operators body position.

Meta-data vs. Audio Annotation

An external file called "meta-data" is a common way to add annotation data to videos. For example, Adobe Extensible Metadata Platform (XMP) format [2] is designed to store annotation data with timestamps in the video capturing device, and has the same benefit as AnnoTone from the viewpoint of time-synchronization with the content. Using these formats could be a smarter way when shooting a video with a camera phone. However, in order to generate such meta-data during recording a video, a user should use a specialized camera or a smartphone application that can handle a specific format, thus it is difficult to use them with standard video cameras. AnnoTone can be used with any video camera that does not have a capability of handling digital data except for video, because it only requires a camera to record the high-frequency modulated signals in the audio track of a video. This is advantageous when one wants to record a high-quality video using a dedicated video camera device. Additionally, having time-synchronized information within a single, standard video file is advantageous because one only needs to copy or send a single file, and that one can edit it using standard video editing software. Managing separate files might be trivial for professionals, but it can cause significant confusion and frustration for general users.

The bandwidth of AnnoTone is 400 bps, which is much smaller than other meta-data formats. However, 400 bps = 50 ascii characters/sec, is sufficient enough for storing textual information like subtitles. A combination of location and orientation data, which consists of five or six floating-point values ($\leq$ 4 Bytes $\times$ 6 = 24 Bytes), could be recorded in less than 0.5 sec. This is sufficient for our target application scenarios. If a user wants to record more data than supported by the current bandwidth capabilities, they can record the data separately and use embedded annotations as anchors to them.

Limitations and Future Work

In most cases, the embedded watermarks are imperceptible to human ears; however, the high-frequency sounds are perceptible to some sensitive listeners. Although watermarks can be removed by applying a low-pass filter, these treatments may result in some degradation of sound quality. To overcome this limitation, we could employ more elaborate methods to completely erase the watermarks with less loss of quality using existing noise-reduction techniques. There is also scope for improving the precision and sensitivity of the watermark transmission and receiving. The current watermarking scheme of AnnoTone uses high-frequency DTMF; however, higher data rates and improved reliability may be obtained by introducing a more sophisticated modulation technique, such as spread spectrum audio watermarking.

Finally, our implementation is a proof-of-concept prototype so that AnnoTone requires the user to develop an one-off smartphone application to generate watermarks. Instead, we can provide a pair of applications that generates and extracts watermarks for individual usage scenario. This will be useful and work for the end-users. For the higher level of users, we can provide customizable general-purpose watermarking application that provides basic annotating functions such as sensor logging and detecting touch positions.

CONCLUSION

We presented a video annotating system called AnnoTone, which can annotate a video during recording with little requirements for specific hardware using an audio watermarking technique. AnnoTone is designed to present new video-editing workflow with a handheld device (e.g., smartphone) and a video camera. We demonstrated the use of our annotation system through video-editing applications that utilize contextual information embedded in videos to provide automated video-editing functionality, and demonstrated the feasibility of our watermarking scheme through a series of performance evaluations. We have discussed the limitations of the current implementation of this technique, and identified a number of directions for further development. We believe that our system has considerable potential for creating a new

style of video recording and editing, by the inclusion of annotations and non-audio/visual data.

REFERENCES

1. Action cam with gps - hdr-a30v/b review - sony us. `http://store.sony.com/compact-pov-action-cam-zid27-HDRAS30V/B/cat-27-catid-All-Action-Cam`. (Accessed: 2014-09-22).
2. Adobe extensible metadata platform (xmp). `http://www.adobe.com/products/xmp.html`. (Accessed: 2014-09-22).
3. Google maps api. `https://developers.google.com/maps/`. (Accessed: 2014-09-22).
4. Ashihara, K., Kurakata, K., Mizunami, T., and Matsushita, K. Hearing threshold for pure tones above 20 khz. *Acoustical science and technology 27*, 1 (2006), 12–19.
5. Bender, W., Gruhl, D., Morimoto, N., and Lu, A. Techniques for data hiding. *IBM systems journal 35*, 3.4 (1996), 313–336.
6. Casares, J., Long, A. C., Myers, B. A., Bhatnagar, R., Stevens, S. M., Dabbish, L., Yocum, D., and Corbett, A. Simplifying video editing using metadata. In *Proceedings of the 4th conference on Designing interactive systems: processes, practices, methods, and techniques*, ACM (2002), 157–166.
7. Cox, I., Miller, M., Bloom, J., and Miller, M. *Digital watermarking*. Morgan Kaufmann, 2001.
8. Cox, I. J., Kilian, J., Leighton, F. T., and Shamoon, T. Secure spread spectrum watermarking for multimedia. *IEEE Transactions on Image Processing 6*, 12 (1997), 1673–1687.
9. Gebbensleben, S., Dittmann, J., and Vielhauer, C. Multimodal audio guide for museums and exhibitions. In *Electronic Imaging 2006*, International Society for Optics and Photonics (2006), 60740S–60740S.
10. Harada, S., Sato, D., Adams, D. W., Kurniawan, S., Takagi, H., and Asakawa, C. Accessible photo album: Enhancing the photo sharing experience for people with visual impairment. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, ACM (2013), 2127–2136.
11. Healy, R., and Timoney, J. Digital audio watermarking with semi-blind detection for in-car music content identification. In *Proceedings of the AES 36th International Conference: Automotive AudioSound in Motion* (2009).
12. Heffner, H. E. Hearing in large and small dogs: Absolute thresholds and size of the tympanic membrane. *Behavioral Neuroscience 97*, 2 (1983), 310.
13. Heffner, H. E., and Heffner, R. S. High-frequency hearing. *Handbook of the senses: Audition* (2008), 55–60.
14. Hirabayashi, M., and Shimizu, M. Cryptone: Interaction between performers and audiences with inaudible dtmf sounds. In *SIGGRAPH Asia 2012 Emerging Technologies*, SA '12, ACM (2012), 5:1–5:4.
15. Lavrenko, V., Feng, S., and Manmatha, R. Statistical models for automatic video annotation and retrieval. In *Acoustics, Speech, and Signal Processing, 2004. Proceedings.(ICASSP'04). IEEE International Conference on*, vol. 3, IEEE (2004), iii–1044.
16. Mackay, W. E., and Beaudouin-Lafon, M. Diva: exploratory data analysis with multimedia streams. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Press/Addison-Wesley Publishing Co. (1998), 416–423.
17. Matsuoka, H., Nakashima, Y., Yoshimura, T., and Kawahara, T. Acoustic ofdm: Embedding high bit-rate data in audio. In *Advances in Multimedia Modeling*. Springer, 2008, 498–507.
18. Nakayama, A., Machino, T., Kitagishi, I., Iwaki, S., and Okudaira, M. Rich communication with audio-controlled network robot. proposal of "audio-motionmedia". In *Proceedings of the 11th IEEE International Workshop on Robot and Human Interactive Communication*, IEEE (2002), 548–553.
19. Patel, S. N., and Abowd, G. D. The contextcam: Automated point of capture video annotation. In *Proceedings of the 6th International Conference on Ubiquitous Computing*, Springer (2004), 301–318.
20. Sarvas, R., Herrarte, E., Wilhelm, A., and Davis, M. Metadata creation system for mobile images. In *Proceedings of the 2nd international conference on Mobile systems, applications, and services*, ACM (2004), 36–48.
21. Schenker, L. Pushbutton calling with a two-group voice-frequency code. *Bell System Technical Journal 39*, 1 (1960), 235–255.
22. Stelmachowicz, P. G., Beauchaine, K. A., Kalberer, A., and Jesteadt, W. Normative thresholds in the 8 to 20khz range as a function of age. *The Journal of the Acoustical Society of America 86*, 4 (1989), 1384–1391.
23. Tachibana, R. Audio watermarking for live performance. In *Electronic Imaging 2003*, International Society for Optics and Photonics (2003), 32–43.
24. Watanabe, K., Tsukada, K., and Yasumura, M. Willcam: A digital camera visualizing users. interest. In *CHI '07 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '07, ACM (2007), 2747–2752.
25. Williams, J. R. Guidelines for the use of multimedia in instruction. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 42, SAGE Publications (1998), 1447–1451.
26. Yi, J., Peng, Y., and Xiao, J. Exploiting semantic and visual context for effective video annotation. *Multimedia, IEEE Transactions on 15*, 6 (2013), 1400–1414.