2009 Advanced Video and Signal Based Surveillance

Cost-Effective Solution to Synchronized Audio-Visual Capture using Multiple Sensors

Jeroen Lichtenauer, Michel Valstar, Jie Shen
Department of Computing,
Imperial College London
SW7 2AZ, United Kingdom
Email: j.lichtenauer@imperial.ac.uk

Maja Pantic
Dept. of Computing,
Imperial College London, UK /
EEMCS, University of Twente, The Netherlands
Email: m.pantic@imperial.ac.uk

***Abstract*—Applications such as surveillance and human motion capture require high-bandwidth recording from multiple cameras. Furthermore, the recent increase in research on sensor fusion has raised the demand on synchronization accuracy between video, audio and other sensor modalities. Previously, capturing synchronized, high resolution video from multiple cameras required complex, inflexible and expensive solutions. Our experiments show that a single PC, built from contemporary low-cost computer hardware, could currently handle up to 470MB/s of input data. This allows capturing from 18 cameras of 780x580pixels at 60fps each, or 36 cameras at 30fps. Furthermore, we achieve accurate synchronization between audio, video and additional sensors, by recording audio together with sensor trigger- or timestamp signals, using a multi-channel audio input. In this way, each sensor modality can be captured with separate software and hardware, allowing maximal flexibility with minimal cost.**

***Keywords*-Video recording; Audio recording; Multisensor systems; Synchronization;**

I. Introduction

In the past two decades, the use of CCTV (Closed Circuit Television) and other visual surveillance technologies has grown to unprecedented levels. Besides security applications, multi-sensorial surveillance technology has also become an indispensable building block of various systems aimed at detection, tracking, and analysis of human behaviour having a wide range of applications including proactive human-computer interfaces, personal wellness and independent living technologies, personalised assistance, etc. Furthermore, research on sensor fusion - combining video analysis with the analysis of audio, as well as other sensor modalities - is becoming increasingly more common. It is also considered as a prerequisite for increase in accuracy and robustness of human behavior analysis. Although, according to [2], humans tolerate an audio lag of up to 200ms or a video lag of up to 45ms, sensor fusion algorithms may benefit from higher synchronization accuracy. For example, in [5], correction of a 40ms time difference, between the audio and video streams recorded by a single camcorder, resulted in increased performance of speaker identification with audio-visual fusion. Also, Lienhart, Kozintsev and Wehr [4] demonstrated that microsecond accuracy between audio channels is required to achieve a gain from distributed blind signal separation.

With this ever-increasing need for multi-sensorial surveillance systems, the commercial sector is keeping up by offering multi-channel frame grabbers and DVRs that encode video (sometimes combined with audio) in real-time (e.g. see http://www.dvrsystems.net). Although these systems can be the most suitable solutions for current surveillance applications, they may not allow the flexibility, quality, accuracy or number of sensors required for technological advancements in human behavior analysis. The spatial and temporal resolutions, as well as the supported camera types of real-time video encoders is often fixed or limited to a small set of choices, dictated by established video standards. The accuracy of synchronization between audio and video is mostly based on human perceptual acceptability, and could be inadequate for sensor fusion. Even if A/V synchronization accuracy is maximized, an error below the time duration between 2 video frames can only be achieved when it is exactly known how the recorded video frames correspond to the audio samples. Furthermore, commercial solutions are often closed systems that do not allow the accuracy of synchronization that can be achieved with direct connections between the sensors themselves. Time-stamping of sensor data with GPS or IRIG-B modules can provide microsecond accuracy. However, applicability depends on sensor hard- and software. Also, actual accuracy can never exceed the uncertainty of the time lag in the I/O process that precedes time-stamping of sensor data. For PC systems, this can be in the order of milliseconds [4].

Because of such shortcomings of commercially available video capture systems, many researchers have sought custom solutions that meet their own requirements. Unfortunately, this often leads to high complexity and/or cost. Zitnick et al. [10] used two specially built concentrator units to capture video from eight cameras of 1024x768 pixels at 15fps. Wilburn et al. [9] built an array of 100 cameras, using 4 PC's and custom-built low-cost cameras of 640x480 pixels at 30fps, connected through trees of interlinked programmable

978-0-7695-3718-4/09 $25.00 © 2009 Crown Copyright
DOI 10.1109/AVSS.2009.92

processing boards with on-board MPEG2 compression. They used a tree of CAT5 cables between the processing boards to synchronize the cameras with an accuracy of 200 nanoseconds. More recently, a modular array of 24 cameras (1280x1024 pixels at 27fps) was built by Tan et al. [8]. Each camera was placed in a separate special-built hardware unit that had its own storage disk, using on-line video compression to reduce the data. Recorded data was transmitted off-line to a central PC via a TCP/IP network. Svoboda, Hug and Van Gool [7] proposed a solution to synchronous multi-camera capture involving standard PC's. They developed a software framework that manages the whole PC network. Each PC could handle up to three cameras of 640x480 pixels at 30fps, although the software under development was still limited to 10fps. Camera synchronization was done by software triggers, simultaneously sent to all camera's through the ethernet network. This solution could reduce costs by allowing the use of low-cost cameras that do not have an external trigger input. However, the cost of multiple PC's remained. Furthermore, a software synchronization method has a much lower accuracy than an external trigger network. A similar system was presented in [1], which could handle 4 cameras of 640x480 pixels at 30fps per PC. The synchronization accuracy between cameras was reported to be within 15 milliseconds. Hutchinson et al. [3] used a high-end server PC with three PCI buses that provided the necessary bandwidth for 4 FireWire cards and a PCI-X SCSI hard drive controller card connecting 4 hard drives. This system allowed them to capture video from 4 cameras of 658x494 pixels at 80fps.

Recent developments in computer hardware technology have significantly increased the capacity of commercial PCs, allowing for more audiovisual sensors to be connected to a single PC. Commencing audiovisual capture by means of a single PC does not only save the costs of additional PCs, but it also saves on the resources needed to synchronize and connect multiple PCs into a single data-capture system. Our multi-sensorial data-capture PC, is built from low-cost Commercial Off-The-Shelf (COTS) components. It facilitates simultaneous, synchronous recordings of audiovisual data from six cameras having 780x580 pixels spatial resolution and 61 fps temporal resolution, together with eight 24-bit 48 KHz audio channels. The overall data rate that the system records is 160 MB per second. By using six 1TB hard drives built in the capture PC, over 9 hours of continuous recordings can be made. Performance tests suggest that the number of cameras can even be increased to 18, if a temporal resolution of 60fps is used, or 36 cameras at 30fps. In Table I we have compared the estimated capacity of our PC wit respect to the number of cameras used for recordings. A higher number of cameras per PC means that more applications can be solved with only a single computer.

Although the cameras in our setup are triggered externally, no expensive or complicated triggering hardware is required. The trigger output of a master camera forms a self-triggering network, by being connected to the trigger inputs of the remaining cameras. External triggering also allows straightforward synchronization of cameras connected to multiple capture PCs. Using less PCs and no extra hardware shifts the costs from the capture infrastructure towards the quality of the sensors. Contrary to previous work, we also provide a solution to accurately synchronize video with audio as well as additional sensor data. This is a crucial extension for applications of human behaviour analysis and sensor fusion. Furthermore, our solution allows to use separate software for the data capture of each modality. This allows the use of off-the-shelf software, maximizing flexibility with minimal development time and cost.

Table I
ESTIMATED CAMERA SUPPORT OF OUR CAPTURE PC

Spatial Resolution	**Temporal Resolution**	**Rate per Camera**	**Max. Nr. of Cameras**
640x480 pixels	30fps	8.8MB/s	48
1024x768 pixels	15fps	11.25MB/s	36
658x494 pixels	80fps	24.8MB/s	18
780x580 pixels	60fps	25.9MB/s	18
1280x1024 pixels	27fps	33.8MB/s	12

II. SYSTEM DESIGN

This section describes the components of our system and explains the motivations behind the most important design choices that we had to make. The relevant components of our setup are summarized in Table II. We will first describe the camera synchronization, followed by the interface between camera and PC, the storage hardware and the motherboard. Subsequently, we describe our solutions for the synchronization of different sensor modalities.

A. Camera Synchronization

While software-triggering is the most low-cost and simple solution for synchronizing cameras, the architecture of general-purpose computer systems implies uncertainty in the arrival times of triggering messages, resulting in unsynchronized frame capture by different cameras. For some applications, this can still be sufficiently accurate. However, for stereo imaging and analysis of fast events by multi-sensor fusion, hardware-triggering is demanded. Unfortunately, web-cams and camcorders generally do not support external triggering. This means that there isn't any choice but to use industrial cameras, which are generally in a higher price range. However, the limited image quality and capture control of web-cams makes them unsuitable for many applications anyway.

The AVT Stingray cameras provide a trigger input as well as output. This allows a relatively simple synchronization network of up to 7 cameras (limited by the maximal output current of one camera), without any extra trigger-

Table II
SYSTEM COMPONENTS

Sensor Component	**Description**
5 monochrome video cameras	AVT Stingray F-044B, 780x580 resolution, max. 61fps
colour video camera	AVT Stingray F-044C, 780x580 Bayer pattern, max. 61fps
camera interface card	Single-bus IEEE 1394b PCI-E×1, Point Grey
camera interface card	Dual-bus IEEE 1394b PCI-E×1, Point Grey
room microphone	AKG C 1000 S MkIII
head-worn microphone	AKG HC 577 L
external audio interface	MOTU 8-pre Firewire 8-channel, 24-bit, 96kHz
Tobii X120	Eye tracker
Computer Component	**Description**
6 Capture disks	Samsung Spinpoint F1 1TB SATA, 32MB Cache, 7,200rpm
System disk	PATA Seagate Barracuda 160GB 2MB Cache, 7,200rpm
Optical drive	PATA DVD RW
4GB Memory	2GB PC2-6400 DDR2 ECC KVR800D2E5/2G
Graphics card	Asustek GeForce 8800GS PCI-E
Motherboard	Asus Maximus Formula, ATX, Intel X38 chipset
CPU	Intel Core 2 Duo 3.16GHz, 6MB Cache, 1333MHz FSB
ATX Case	Antec Three Hundred
PSU	Corsair Memory 620 Watt
Software Application	**Description**
MS Windows Server 2003	32-bit Operating System
Norpix Streampix 4	Multi-camera video recording
Audacity 1.3.5	Freeware multi-channel audio recording
AutoIt v3	Freeware for scripting of Graphical User Interface control
Tobii Studio	Eye tracking & stimuli data suite
Tobii SDK	Eye tracker Software Development Kit

or amplification hardware. When the trigger output of the master camera is used as the input to the slave cameras, the resulting delay of the slave cameras is approximately 30 microseconds. If more than 7 cameras must be synchronized, either a trigger amplifier/relay must be used, or the output of one of 6 slave cameras can again be used as a trigger for 6 additional slave cameras. However, at each such step in the chain, another 30 microseconds delay is added.

B. Camera Interface

The camera interface has an impact on cost, bandwidth, maximal number of cameras, as well as the CPU load [6]. The three main interfaces for machine vision cameras are FireWire (400 or 800), ‘GigE Vision’ and ‘Camera Link’.

FireWire allows isochronous data transfer (by default up to 74MB/s for IEEE 1394b). Isochronous data can be written directly to a DMA buffer by the FireWire bus controller, with a negligible CPU load. The maximum number of cameras that can be connected to one FireWire bus is typically limited to 4 or 8 (DMA channels), depending on the bus hardware.

’GigE Vision’ is an upcoming camera interface, based on Gigabit Ethernet (GbE), specifically standardized for machine vision. Depending on cameras, network configuration and packet loss, one GbE connection can support up to 100MB/s from multiple cameras. If many GigE cameras are connected to one PC, using multiple GigE connections, the CPU load can become significant. This can be reduced by using a special Network Interface Card/Chip (NIC) driver.

Camera Link (CL) is an interface that is specifically designed for high-bandwidth vision applications. CL is only interesting if a single camera outputs more data than one FireWire bus or GigE connection can handle. CL grabber cards usually support only one or two cameras each, and require a 4x or 8x PCI-E slot. If multiple CL cards are required, it will become increasingly difficult to find a suitable motherboard with the required number of 4x or 8x PCI-E slots. Also, CL components are not made for the consumer market, meaning that cables and hardware are expensive. For these reasons, the Camera Link interface is generally not suitable for surveillance applications.

We have chosen for the IEEE 1394b interface, because GigE Vision technology is still very new, with many uncertainties (e.g. the effective data capacity and the amount of CPU load) and a limited choice of cameras, while IEEE 1394b is massively supported and works with general-purpose interface cards that write image data directly through DMA. However, when GigE Vision becomes more common and NICs are available that keep CPU load low with large amounts of image data from multiple cameras, GigE may be a better alternative for multi-camera capture in the future.

C. Storage

Currently, the Hard Disk Drive (HDD) is the most significant bottleneck of a conventional PC. Capturing to internal memory (RAM) is the best solution for short video fragments. However, many applications require significantly longer recordings than what can be stored in RAM. The fastest consumer Serial-ATA (SATA) HDDs currently start with a data rate of over 100MB/s (at the outside of the platter) and gradually descent to a rate of around 60MB/s at the end of the disk. The decrease in write transfer rate (WTR) of a 1TB Samsung Spinpoint F1 is shown in figure 1.

Most high-end consumer motherboards provide SATA connections for six disks, including hardware RAID support, which will allow a total capture rate of approximately 500MB/s (depending on how much of the disk space is used for capture). Video streams from multiple cameras can be either written to separate HDDs, or to a single RAID0 disk

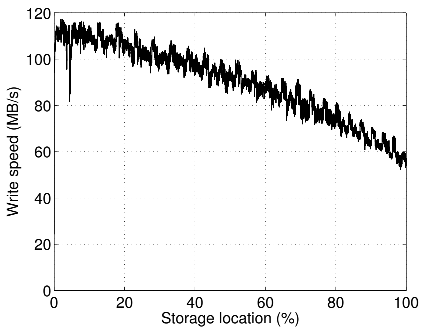

Figure 1. Sequential write transfer rate of Samsung Spinpoint F1 1TB HDD as a function of disk location

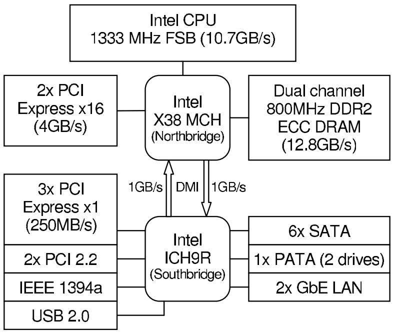

Figure 2. Overview of Asus Maximus Formula motherboard with Intel X38 chipset

that consists of multiple physical member HDDs. A RAID0 disk has a size equal to the number of member disks (N) multiplied by the size of the smallest disk, and a WTR that comes close to $N\times$ the throughput of the slowest disk.

D. Motherboard

After the HDD WTR, the motherboard is often the second most important bottleneck for data capture. Unfortunately, the actual performance of a motherboard is hard to predict, as it depends on a combination of many factors. But, first of all, it should have a sufficient number of storage connections, PCI-E slots and memory capacity.

The most obvious choice is to use a high-end server motherboard, with a chipset such as the Intel 5000 or better, supporting only Intel Xeon CPUs. However, this may be more costly than necessary. Recently, the gaming industry has developed some consumer motherboards that are very well suited for video capture, at a much lower price.

Figure 2 shows the overview of the Asustek 'Maximus Formula' board, used in our experiments, that has an Intel X38 chipset. It supports up to 8GB of ECC DDR2 800MHz RAM and has 6 SATA connections (with RAID support), as well as support for two PATA (IDE) devices. This means that with 6 HDDs for image capture, a system disk and optical drive (for installing software) can still be connected to the PATA interface. The motherboard has two PCI-E$\times$16 slots, that are connected directly to the Northbridge, and three PCI-E$\times$1 slots connected to the Southbridge.

During a video capture process, each FireWire Bus Card (FBC) transfers video data to DRAM, while the video capture application copies received video frames into DRAM frame buffers. From the frame buffers, the data is subsequently formatted (and possibly compressed) and transferred to the HDDs, connected to the Southbridge. The DMI link between North- and Southbridge limits the total HDD WTR to 1GB/s, minus overhead and other southbound data. The rate of northbound video data (coming from the FBCs) can be reduced by placing one or more of the FBCs in a PCI-E$\times$16 slot (compatible with PCI-E$\times$1, -$\times$2 , -$\times$4 and -$\times$8), connected directly to the Northbridge.

If a PCI graphics card is used, five PCI-E$\times$1 cards with dual IEEE 1394b bus could be installed that each support 16 cameras. This totals to 740MB/s of video data from up to 80 cameras. Even more cameras could be connected through the on-board FireWire 400 and/or a PCI IEEE 1394b card.

Other consumer-class motherboards with similar specifications are the Asustek 'Rampage Formula' or 'P5E Deluxe' (which have the newer X48 chipset), the Gigabyte X38 or X48 boards and the Gigabyte EP45 boards with RAID functionality.

E. Cross-Modal Synchronization

When using software to synchronize the capture of video with other sensor modalities, the synchronization accuracy will be limited by the uncertainty in the latency between the sensor measurement and the handling of the data in the software. Depending on sensors, hardware and software, this latency may be anything from a few milliseconds up to more than hundreds of milliseconds. If there is no control over the exact sampling rates, synchronization errors may even accumulate during a recording.

A simple and low-cost solution for high accuracy, is to make use of the synchronization between multiple audio channels of a single audio interface. In the case of a stereo audio input, one channel can be used to record sound, while the second channel can be connected to the camera trigger network. In this way, the trigger pulses that activate the capture of each video frame, are recorded alongside the audio. Our audio input was provided by a MOTU 8Pre external audio interface, connected with a capture PC through an IEEE 1394a connection. The 8Pre can record up to 8 parallel audio channels at 24-bit, 96kHz. The 12Volt camera trigger network was connected to one of the analog audio inputs, via a voltage divider. The pulses in the respective audio track can easily be detected and matched with all the captured video frames, using their respective frame number and/or timestamp. With an audio sampling rate of 48kHz, the uncertainty of synchronization that can be achieved, in this way, is below $50\mu s$.

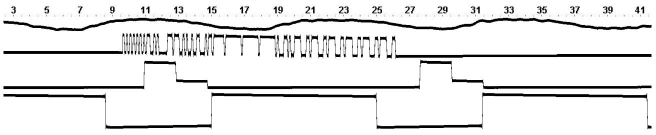

Figure 3. Signals recorded simultaneously. From top to bottom: microphone; serial port timestamp; measured infrared light in front of eye tracker; camera frame integration. Time axis indicates relative milliseconds.

Another advantage of this synchronization method is that it allows to use off-the-shelf software applications for capturing each modality separately. Any type of sensor can be synchronized to the audio data, as long as it produces a measurable signal at the moment of taking a sample and its output data include reliable sample counts or timestamps relative to the first sample.

For sensors that do not have a trigger output, such as the Tobii X120 Eye Tracker, we required an alternative solution. The data of the eye tracker is accurately time-stamped in synchronization with the CPU cycle counter of the computer that runs the Eye-tracking application. However, there is no accurate relation between the CPU counter time and the recorded audio data. To establish this link, we developed an application which periodically transmits the current CPU cycle time through the serial port. This signal is recorded as a separate audio channel, allowing to accurately relate events in the other audio channels (such as the video triggers or sound tracks) with the CPU cycle time. An example of such a timestamp is shown as the 2nd signal in figure 3. A parallel recording of the infrared X120 eye tracker strobe illumination (from a photo diode), allows correction of errors in gaze data synchronization up to 8ms. Recording CPU cycle timestamps of multiple PCs makes it possible to synchronize between capture software running on different PCs and RF transmission of this signal allows for wireless system integration [4]. The delay of the serial port timestamps was in the order of 30 microseconds. By compensation of the average delay, the accuracy was increased even further. However, in the end, accuracy is still limited by the audio sampling rate.

F. Software

Our proposed multi-sensor capture solution does not depend on the specific choice of software. However, when using off-the-shelf components, Microsoft Windows operating systems are currently the most suitable for multi-sensor applications. This is because the support of hard- and software is mostly aimed at these operating systems. This especially holds for low-cost products developed for the consumer market.

The video capture is handled by 'Streampix 4', which can record video to HDD, from multiple sources simultaneously, and in a format that allows sequential disk writing. The latter is essential to reach the full WTR of a HDD. After the recording, the sequences can be processed, exported and compressed by any installed video CODEC.

When each sensor has its own capture software, controlling the starting, stopping and exporting of data recordings quickly becomes unmanageable. Unfortunately, many applications under MS Windows only work by Graphical User Interface (GUI), not allowing for scripting. This problem is solved by the freeware scripting package AutoIt v3, which can switch between applications, read window contents, activate controls and emulate keyboard and mouse actions.

III. Results

The audio data consisted of 8 synchronous channels at 24-bit, 48kHz sampling rate. This amounts to only 1.1MB/s of data, that was streamed to the HDD that also contained the operating system and software. Because the video data rates are orders of magnitude higher, and streamed separately to the six SATA disks (see table II), all our experiments concentrated on video throughput. However, they were always conducted under simultaneous audio capture.

A. Maximum storage throughput

When we streamed 6 cameras at 61.7fps (26.6MB/s per camera) to 3 HDDs (two cameras per disk), the capture runs successfully up to the full 1TB storage capacity of each HDD. During this audio/video capture, the average CPU usage was around 20%. Streamed to 2 HDDs (three cameras per disk), the capturing runs successfully up to 35% of HDD space (333GB). This is due to the reduction of WTR on the inner parts of the HDD platters (see figure 1). This means that at 35% capacity, each HDD has a bottle neck at 79.8MB/s. In line with this result, we found that, at the same 35% storage capacity, we could stream 4 cameras to one HDD, at 46fps, five cameras at 36fps and six cameras at 30fps.

Next, we tested if the motherboard could really capture 480MB/s (from 18 cameras at 61.7fps) to use the full throughput of all six HDDs together. Since we only had six cameras, we used the benchmarking tool 'HD_speed v1.5.4.72' to simulate the additional video data streams. HD_speed either reads from or writes to a single HDD, at the highest rate possible, and shows a live measurement of the achieved rate. We assumed that the input of 8 cameras, connected to the Northbridge chip via two dual-bus PCI-E×1 IEEE 1394b cards in the PCI-E×16 slots, would not have an influence on the performance of the remaining system. Therefore, we only tested the input data from the remaining 10 cameras on the Southbridge, using 6 real cameras plus a reading test on a HDD connected via a PCI-E×1 SATA controller. This reading test should achieve at least 106.5MB/s, to simulate the required video input data. The disk writing data consisted of 160MB/s of real video data to two HDDs, plus three WTR tests on the remaining three disks. These write tests must achieve a combined rate of at least 320MB/s to attain the required throughput.

Although the read test did achieve a rate of 114MB/s, the combined rate of the WTR tests leveled out at around 310MB/s. Furthermore, the video buffers occasionally gathered some writing queues, indicating that the writing limit was approached.

Considering these results, we can conclude that 470MB/s is the maximum video capture throughput of this system. This would be sufficient to capture at 60fps from 18 cameras (resulting in 466MB/s).

B. Discussion

We have found that the system should be capable of capturing 470MB/s of video data, together with 1.1MB/s of audio. We must note, however, that a data stream from a camera input, sent to a DMA buffer, is not fully equivalent to data generated by reading from a hard disk. Also, these experiments were done using a graphics card connected to a PCI-E×16 slot. When all PCI-E slots are needed for data I/O, the graphics card must use a regular PCI slot, connected to the ICH9R. We did not test how this additional load on the ICH9R would effect the data capture process. Furthermore, some headroom is required to ensure robustness of the system. When getting even close to the limit 470MB/s, another motherboard/chipset would be advisable, with a more powerful Southbridge and/or more PCI-E slots connected directly to the Northbridge.

IV. Conclusion

Using commercial off-the-shelf components, we built an audio/video capture PC that was able to simultaneously capture over 9 hours of video from six cameras with resolutions of 780x580 pixels each, at 61.7 fps, together with 8 channels of 24-bit audio at 48kHz sampling rate. This amounts to a data rate of 160MB/s.

When 6 cameras (totalling 160MB/s of data) are streamed to 2 of the 6 video HDDs (three cameras per disk), data capture runs successfully up to 35% of HDD space. This implies that a total of 18 cameras could be supported simultaneously at 61.7fps. This would produce a data rate of 480MB/s. However, the maximum throughput that the system could handle was found to be 470MB/s, limited by the motherboard. The current system could possibly handle up to 18 cameras at 60fps, 16 cameras at 61.7fps, or 36 cameras at 30fps. However, to insure stability, a more powerful motherboard would be advisable when approaching these limits.

The synchronization between audio and externally triggered sensors, such as the video cameras, is realized by recording the trigger signal as one of the audio channels. With an audio sampling rate of 48kHz, this method allows a synchronization error below 50 microseconds. Synchronization with sensors that do not have an external trigger signal, possibly captured with another computer, was solved by a background program that periodically outputs a timestamp signal through the serial port, which can be recorded to an additional audio channel.

Our approach does not require complicated or expensive synchronization hardware, and allows to use separate capture software for each sensor modality, maximizing flexibility with minimal cost.

Acknowledgment

The research of M. Valstar leading to these results is funded in part by the European Community's 7th Framework Programme [FP7/2007-2013] under the grant agreement no 211486 (SEMAINE). The research of the other authors is funded by the European Research Council under the ERC Starting Grant agreement no. ERC-2007-StG-203143 (MAHNOB).

References

[1] X. Cao, Y. Liu, and Q. Dai. A flexible client-driven 3dtv system for real-time acquisition, transmission, and display of dynamic scenes. *EURASIP Journal on Advances in Signal Processing*, 2009. Article No. 5.

[2] K. W. Grant, V. van Wassenhove, and D. Poeppel. Discrimination of auditory-visual synchrony. In *International Conference on Audio-Visual Speech Processing*, pages 31–35, 2003.

[3] T. Hutchinson, F. Kuester, K.-U. Doerr, and D. Lim. Optimal hardware and software design of an image-based system for capturing dynamic movements. *IEEE Transactions on Instrumentation and Measurement*, 55(1):164 – 175, February 2006.

[4] R. Lienhart, I. Kozintsev, and S. Wehr. Universal synchronization scheme for distributed audio-video capture on heterogeneous computing platforms. In *The Eleventh ACM international Conference on Multimedia*, pages 263–266, Berkeley, CA, USA, November 2003. ACM.

[5] M. Sargin, Y. Yemez, E. Erzin, and A. Tekalp. Audiovisual synchronization and fusion using canonical correlation analysis. *IEEE Transactions on Multimedia*, 9(7):1396–1403, November 2007.

[6] S. Sookman. Choosing a camera interface: qualify and quantify. *Advanced Imaging*, May 1 2007.

[7] T. Svoboda, H. Hug, and L. Van Gool. Viroom - low cost synchronized multicamera system and its self-calibration. In *Proceedings of the 24th DAGM Symposium on Pattern Recognition*, volume 2449 of *Lecture Notes In Computer Science*, pages 515 – 522, London, UK, 2002. Springer-Verlag.

[8] S. Tan, M. Zhang, W. Wang, and W. Xu. Aha: An easily extendible high-resolution camera array. In *Second Workshop on Digital Media and its Application in Museum & Heritages*, pages 319–323. IEEE, December 2007.

[9] B. Wilburn, N. Joshi, V. Vaish, E. Talvala, E. Antunez, A. Barth, A. Adams, M. Horowitz, and M. Levoy. High performance imaging using large camera arrays. *ACM Trans. Graph.*, 24(3):765–776, 2005.

[10] C. Zitnick, S. Kang, M. Uyttendaele, S. Winder, and R. Szeliski. High-quality video view interpolation using a layered representation. In *ACM SIGGRAPH*, pages 600–608, Los Angeles, CA, USA, August 2004.