

# Software Synchronization of Projector and Camera for Structured Light 3D Body Scanning

Tomislav PETKOVIĆ<sup>\*1</sup>, Tomislav PRIBANIĆ<sup>1</sup>, Matea ĐONLIĆ<sup>1</sup>, Nicola D'APUZZO<sup>2</sup>  
<sup>1</sup> University of Zagreb, Faculty of Electrical Engineering and Computing, Zagreb, Croatia;  
<sup>2</sup> Hometrica Consulting, Ascona, Switzerland

DOI: 10.15221/16.286    <http://dx.doi.org/10.15221/16.286>

## Abstract

Current 3D body scanners based on structured light principle are expensive and somewhat bulky machines which limits their wider spread. One of the factors driving the scanner cost up is the requirement to synchronize projector and camera which requires either expensive system components a costly customized hardware synchronization solution. We propose a software solution to the projector-camera synchronization problem which enables construction of low-cost 3D scanner using common commercial off-the-shelf components only: a projector, a camera, and a computer. We also propose a simple calibration procedure for precise measurement of delay time which is necessary to achieve proper synchronization. We have developed a prototype system for 3D human body scanning using the proposed approach which achieves projector-camera synchronization and enables data acquisition at 30 FPS.

**Keywords:** projector-camera synchronization, software synchronization, structured light, 3D body scanning

## 1. Introduction

Structured light 3D scanners are comprised of two main components: a projector and a camera. The task of the former is to project one or more patterns onto the scanned object whereas the role of the later is to acquire the corresponding images. By combining commercial off-the-shelf camera and projector one should therefore be able to quickly assemble a low-cost 3D scanner based on the structured light principle. For this to be possible the problem of projector-camera synchronization [1] must be solved first: a projected structured light (SL) pattern must not change during image acquisition. Additionally, the projector-camera synchronization must be implemented in a way which enables fast acquisition rates. For most applications, 3D body scanning included, both requirements are essential. Hence, solving the synchronization problem is necessary for successful 3D scanning [2].

Approaches to the synchronization problem in structured light scanning may be divided into three distinct groups: (a) hardware-based synchronization approaches [2,3,4,5,6,7,8,9,10,11,12,13,14,15]; (b) software-based synchronization approaches [1,16,17,18]; and (c) no-synchronization approaches [19,20,21,22,23]. (Most notably, some works which stress the importance of projector-camera synchronization omit the details about how the synchronization is implemented [17,18,24].)

Hardware-base solutions to the synchronization problem use hardware to trigger the camera synchronously to the pattern projection. In [2,3,4] a standard analog video-genlock [25] is used; such solutions are limited to standard broadcasting acquisition rates of 25 or 30 FPS. In [5,6,7,13,14] a camera having an external trigger input is used; the camera is then triggered by a synchronization pulse output by the projector or an external synchronization/triggering circuit. Acquisitions speeds of such custom solutions are not limited to broadcasting rates; in [7] a 3000 FPS for 3D data is reported. Beside these somewhat standard solutions custom hardware designs which achieve high acquisition speeds are also proposed: S. Zhang et al. present a custom solution for synchronizing DLP projectors reporting 120 FPS image acquisition rate (40 FPS for 3D data) in [8,9]; this is later extended first to 240 FPS (120 for 3D) in [10] and finally to 2000 FPS (667 for 3D) in [11]. Overall, hardware-based solutions are proposed often as they enable reliable synchronization and offer fast acquisition rates at the cost of additional synchronization/triggering circuits. We note that synchronizing the camera to the video signal alone using an external triggering circuit disregards the inherent input lag [26] of the projector which therefore must be precisely measured to truly synchronize the acquisition system. Therefore, the proper hardware synchronization requires a specialized projector as nowadays most common commercial off-the-shelf projectors (especially low-cost ones) do not have input/output connections for easy synchronization with other devices.

\* [tomislav.petkovic.jr@fer.hr](mailto:tomislav.petkovic.jr@fer.hr); +385 1 6129563; [www.fer.unizg.hr](http://www.fer.unizg.hr)

Software-based solutions to the synchronization problem are not often described in the literature despite the fact that there exists a large body of work devoted both to aligning unsynchronized video streams and to synchronizing multiple cameras. Software-based approaches described in the literature [16,1,17,18] utilize a personal computer or a microcontroller to drive both the projector and the camera, which are then synchronized in software only. The simplest possible scheme is described in [16] where the system uses prolonged wait intervals between projecting and acquiring an image; although simple such an approach is flawed as it significantly extends the duration of the acquisition process. In [1] Jaeggli et al. propose a simple method for coarse measurement of projector-camera delay time and describe a better scheme where the camera acts as a synchronization source. Both [17,18] state the projector and camera are synchronized in software but omit the implementation details. Surprisingly, software based projector-camera synchronization in structured light applications has never been investigated up to a performance comparable to the hardware-based solutions.

No-synchronization approaches impose certain limitations which make the projector-camera synchronization unnecessary [19,20,21,22,23]. The simplest approach is a static SL pattern; examples include Microsoft Kinect [19], Google Project Tango [20], and Structure Sensor [21]. Static pattern also enables fast acquisition rates: in [22] it is used to achieve 1000 FPS. Drawbacks of a static pattern are reduced spatial resolution and precision compared to non-static patterns. Furthermore, the projector must project a constant image, therefore low-cost DLP projectors which use time-multiplexing projection are inherently ill-suited for such approaches. A different approach is proposed by Moreno et al. [23] where an imaging model of a rolling-shutter camera is used to retrieve the timing information from the distortions of the recorded Gray-code pattern; such an approach is quite complex as it requires substantial computational power to retrieve the 3D data. Due to necessary limitations which are imposed on system components by no-synchronization approaches there are of interest only in applications where fast acquisition is more important than reduction in spatial resolution and precision.

Nowadays general purpose computers and microcontrollers have the capabilities to measure time intervals with precision and accuracy both in the order of microseconds, which is more than sufficient for software-only projector-camera synchronization. We propose a software-based projector-camera synchronization approach which utilizes common commercial off-the-shelf components, a projector, a camera and a desktop PC. The proposed approach uses a vertical blank (VBLANK) interrupt as a master timing signal which controls both the projector and the camera. We also propose a simple measurement scheme for high-precision measurement of projector-camera delay. A developed prototype using the proposed synchronization approach achieves acquisition speed of 30 FPS, which is typical for non-custom hardware solutions, while removing one of the components of such hardware solutions, a separate synchronization/triggering board. Compared to previous works on software synchronization [16,1,17,18]: (1) we propose to use a VBLANK interrupt as a master timing signal, which is a natural choice and simplifies the implementation; and (2) we propose a measurement scheme for high-precision measurement of projector-camera delay which requires no additional equipment. To the best of our knowledge, this is a first study which reports software-synchronization in SL which achieves acquisition rates comparable to hardware solutions.

This article is organized as follows: In Section 2 we describe the software-based synchronization approach. In Section 3 we describe the developed prototype, present obtained experimental results and provide discussion. We conclude in Section 4.

## 2. Software Synchronization

During 3D scanning a SL pattern is projected by the projector; each of projected pattern images must then be synchronously captured by the camera. When implementing a software-based synchronization one of the system components, i.e. a projector, a camera, or a computer, must be chosen as a synchronization source to which all other components are then slaved. A natural choice of the master timing signal is a computer (or a microcontroller) which controls both the projector and the camera.

A key insight required to synchronize the camera to the projector using software only is a fact that almost all video display adapters generate a VBLANK interrupt which is normally used to synchronize graphics rendering. Therefore, if the video display adapter drives the projector then the generated VBLANK interrupts are always synchronized to the projected images (disregarding a constant input lag) and may be used to trigger the camera in software only. We note that using VBLANK for synchronization is in principle identical to external triggering/synchronization circuits which trigger the camera synchronously to the vertical synchronization pulse (VSYNC) of the video signal.

## 2.1. High-Resolution Time Measurement

A necessary prerequisite for software-only synchronization is the ability of the controlling unit, a computer or a microcontroller, to precisely measure time intervals: the controlling unit must have a high-resolution timer. All modern processors have this ability, e.g. a time stamp counter on x86/x86-64 processors [27] and constant frequency counters of Arm processor family, to name but a few. Therefore, time interval measurement with microsecond precision in software only is possible without external timekeeping devices.

The proposed software synchronization solution performs high-resolution time measurements as follows: (1) VBLANK interrupts are always counted and are used for coarse timing tasks, and (2) high-resolution timers are used to precisely trigger the camera some predetermined time after the VBLANK interrupt.

A crucial requirement for software synchronization to work is: the synchronizing computer must have sufficient processing power for all crucial tasks it must perform. The tasks are: (a) pattern rendering; (b) image acquisition; and (c) precise timing of all actions using VBLANK interrupts and high-resolution timers. For example, if projector and camera operate at 60 FPS then the computer must be capable of preparing one pattern image and of processing one captured image in time which is significantly shorter than 16.67 ms (1/60 s), especially if a non-real time operating system is used. This requirement is easily fulfilled by most of modern personal computers.

## 2.2. Camera and Projector Delays

Although video display hardware generates VBLANK interrupts synchronously with the projected SL pattern images there exists a problem of an unknown delay between the VBLANK interrupt and the time the pattern image is projected by the projector; let  $t_p$  denote this delay of the projector. There is also an unknown delay between the camera software trigger and the start of frame integration; let  $t_c$  denote this delay of the camera. Frame acquisition must be started as soon as possible after the pattern image is projected, therefore the software trigger must be sent to the camera no sooner than  $t_D$  after the corresponding VBLANK interrupt, where  $t_D = t_p - t_c$ . In other words, camera software triggers must be scheduled for execution at  $t_D$  time after corresponding VBLANK interrupts.

Delay times  $t_p$  and  $t_c$  are difficult to measure; measuring them requires expensive measuring equipment. However, we note that the delay  $t_D$  is sufficient to fully synchronize the acquisition system. We propose a convenient approach for measuring the delay  $t_D$  using the same projector, camera, and computer together with a simple white board making expensive time-measuring equipment unnecessary. The only requirement is that the camera exposure time  $t_E$  can be set to a value larger than  $t_p + t_V$ , where  $t_V$  is the display refresh time (inverse of projector's display refresh rate). To measure  $t_D$  only three images of the whiteboard illuminated by the projector have to be recorded by projecting patterns as shown in Fig. 1. For the first image we project an all-white pattern and compute the average white intensity  $I_W$  for the illuminated area of the whiteboard only. Then we project an all-black pattern and compute the average black intensity  $I_B$  for the same area. For the last acquisition we record the transition from all-black to all-white pattern as follows: (1) prior to the VBLANK interrupt an all-black pattern is projected; (2) at the same VBLANK interrupt the camera is triggered; and (3) at the VBLANK interrupt the pattern is switched from all-black to all-white. Denote by  $I_{BW}$  the average image intensity observed for the transition from all-black to all-white pattern for the same area. Then, under the assumption of constant delay times  $t_p$  and  $t_c$ , the observed intensities and frame integration times satisfy the following equation:

$$I_{BW} = \frac{t_D}{t_E} I_B + \frac{t_E - t_D}{t_E} I_W. \quad (1)$$

The delay time  $t_D$  is then obtained using

$$t_D = t_E \frac{I_W - I_{BW}}{I_W - I_B}. \quad (2)$$

Note that Eq. (2) measures the delay time  $t_D$  indirectly by measuring intensities; as modern cameras usually have much more than one million pixels which are averaged in  $I_W$ ,  $I_B$ , and  $I_{BW}$  the delay time  $t_D$  is effectively measured with a microsecond resolution.

Once the delay  $t_D$  is known triggering the camera exactly  $t_D$  after the VBLANK interrupt effectively synchronizes the acquisition system.

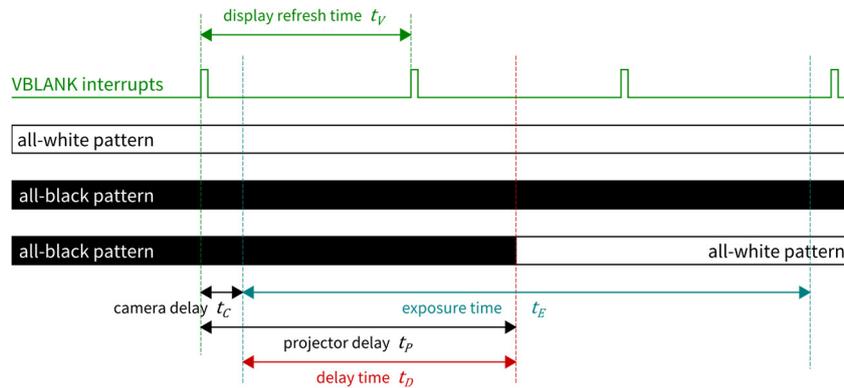


Figure 1. Measuring delay time  $t_D$  of the acquisition system.

### 2.3. Software Architecture

The tasks which need to be executed in software are: (1) rendering pattern images; (2) presenting pattern images; (3) triggering the camera; (4) transferring the data from the camera; and (5) pattern decoding and 3D reconstruction. Many of these tasks may be executed concurrently so we propose a multithreaded design using several threads with carefully assigned thread priorities as listed in Table 1. The *3D reconstruction thread* may be omitted if the acquired data is analyzed off-line. The *image transfer thread* is almost always implemented by a camera SDK. The threads which must be implemented and which we will describe in more details are the *image rendering thread* and the *image acquisition thread*.

Table 1. Threads, tasks, and priorities.

ID	Name	Tasks	Thread Priority
1	<i>image rendering thread</i>	<ul style="list-style-type: none"> <li>– counting VBLANK intervals</li> <li>– rendering pattern images</li> <li>– presenting pattern images</li> </ul>	highest/time critical
2	<i>image acquisition thread</i>	<ul style="list-style-type: none"> <li>– triggering the camera</li> </ul>	highest/time critical
3	<i>image transfer thread</i>	<ul style="list-style-type: none"> <li>– image data transfer</li> </ul>	above normal
4	<i>3D reconstruction thread</i>	<ul style="list-style-type: none"> <li>– pattern decoding and 3D reconstruction</li> </ul>	below normal

We have implemented the logic of the *image rendering* and the *image acquisition* threads using simple finite state machines which operate as illustrated by the timing diagram in Fig. 2. The *image rendering thread* executes its tasks at each VBLANK interrupt as follows: (a) if the remaining time to the next camera trigger is shorter than  $t_V$  then alert the *image acquisition thread*; (b) immediately after one frame is presented render the next frame; and (c) present the next frame at the first possible VBLANK only after the *image acquisition thread* sends the notification that previous acquisition succeeded. The *image acquisition thread* waits to be alerted by the *image rendering thread* and executes its tasks as follows: (a) if the next frame is presented then continuously measure time using high-resolution timer and trigger the camera  $t_D$  after VBLANK; and (b) once the frame is acquired notify the *image rendering thread*.

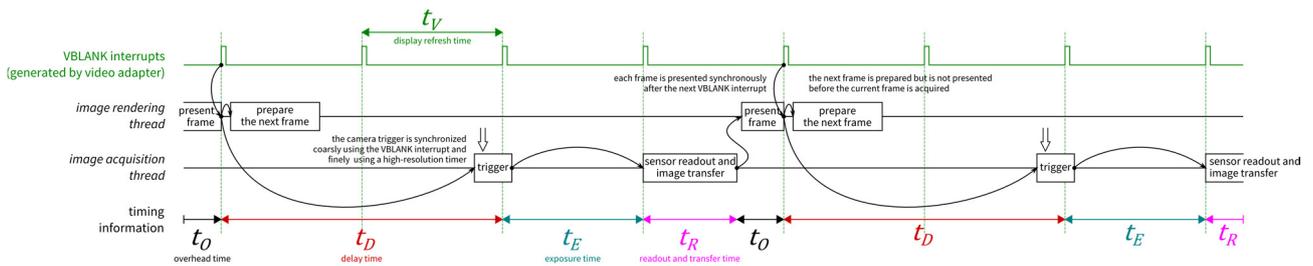


Figure 2. System timing diagram showing the order in which the tasks are executed.

The maximum achievable acquisition frame rate when using the proposed software architecture is

$$\text{FPS}_{\text{MAX}} = \frac{1}{t_D + t_E + t_R + t_O}, \quad (3)$$

where  $t_D$  is the delay time,  $t_E$  is the camera exposure time,  $t_R$  is the frame readout and data transfer time, and  $t_O$  is the unavoidable overhead time to the next VBLANK interrupt (see Fig. 2).

Most common operating systems today are not real-time systems and may lack an application programming interface which offers sufficient precision to execute the camera trigger exactly  $t_D$  after the corresponding VBLANK interrupt. As no such problem exists with the VBLANK interrupt we use the *image rendering thread* to alert the *image acquisition thread* at the VBLANK interrupt exactly preceding the moment when camera must be triggered. Then the *image acquisition thread* spinlocks to precisely determine the moment when the trigger must be sent (note that using a spinlock is not allowed if a single-core processor is used).

Due to inherent constraints of non real-time operating systems the proposed architecture is by design causal: if any of the tasks which are shown in Fig. 2 take more than the anticipated time then the data acquisition will not fail but will simply be prolonged for the unexpected delay. Therefore, the proposed acquisition scheme is robust in the sense that frames can never be dropped. Taking this into consideration Eq. (3) gives the expected frame rate in normal operating conditions only; in some exceptionally rare circumstances the actual frame rate may be lower.

## 2.4. Equipment Limitations

Three major components of the system are projector, camera and computer. Here we list most important limitations when selecting from among common commercial off-the-shelf components.

Low-cost projectors today most commonly use one of DLP, LCD or LCoS technologies. To achieve color projection DLP uses time-multiplexing while LCD and LCoS use spatial multiplexing. The use of time-multiplexing in DLP projectors imposes a condition on camera exposure time: the camera exposure time  $t_E$  must always be set to a multiple of projector's display refresh time  $t_V$ , otherwise the acquired image will exhibit the unwanted rainbow effect. No such limitations exist for LCD and LCoS projectors. The proposed software synchronization approach is applicable to all three types of projectors if the exposure condition is satisfied for the DLP projectors.

Selected camera must have the capability to be triggered using software only. This requirement rules out the cheapest cameras which all have a fixed internal trigger.

Selected computer or microcontroller must be capable of measuring time in high-resolution, i.e. in order of microseconds or better. In-built time measuring devices are implemented as counters; however, one must carefully select a device in which the counter counts ticks of a constant frequency source. We have already mentioned the TSC counter of the x86/x86-64 processor family; some older processor models have TSC which uses a non-invariant frequency source and therefore cannot be used for reliable time measurements with microsecond resolution (see [27] for details).

## 3. Results and Discussion

### 3.1. System Prototype

We have constructed a low-cost 3D human body scanner comprised of off-the-shelf components: a short-throw DLP projector (Mitsubishi EW230U-ST), two USB3 Vision cameras (PointGray Grasshopper3 23S6C) equipped with wide-angle lenses (Kowa LM8HC), and a personal computer (Intel Core i5-4590@3.30GHz running Microsoft Windows). The DLP projector and cameras are mounted on a simple vertical pole placed on a wheel-cart to make the scanner easy to use (Fig. 3).



Figure 3. Prototype of a low-cost 3D human body scanner:  
 (a) side view; (b) top camera; (c) bottom camera; (d) back view showing a calibration board.  
 Arrow indicators: cameras are blue, projector is red, and calibration board is green.

We have implemented the proposed software based synchronization for the Microsoft Windows operating system. Microsoft's DirectX Graphics Infrastructure (DXGI) [28] is used to wait on the VBLANK interrupt via `IDXGIOutput::WaitForVBlank` method; this provides coarse timekeeping. Windows QPC API is used to access the TSC register of x86/x86-64 processors for high-resolution timekeeping which is implemented using a spinlock mechanism.

The developed system may use any structured light pattern sequence, however, we have implemented one of the most popular SL strategies for the 3D reconstruction, Gray code and phase shifting combination (GC+PS) [29]. The implementation offers both 3D reconstruction and visualization and is therefore a complete 3D human body scanning solution which demonstrates the feasibility of the proposed approach.

The system was calibrated using the modified method of Moreno and Taubin [30] where the Gray code SL pattern was replaced by the combination of Gray code and phase-shifted fringe and where a square calibration pattern was replaced with a circle pattern (see Fig. 3d). The delay time  $t_D$  of the system was measured using procedure described in Section 2.2. at 16.804 ms. Note that this quite fast delay is achieved only when the projector is set to 3D frame sequential 1280x800@119.909Hz mode; using a non-3D mode raises the delay time to a standard 30-42 ms depending on the refresh rate used. The camera exposure time  $t_E$  was set to 8.340 ms, which corresponds to one rotation of the color wheel of the DLP projector in 1280x800@119.909Hz mode. The readout and data transfer time  $t_R$  is shorter than 7 ms so  $t_D + t_E + t_R$  is 32.144 ms, which is less than the limit of 33.333 ms required for 30 FPS. The developed prototype acquires images with the rate of 30 FPS.

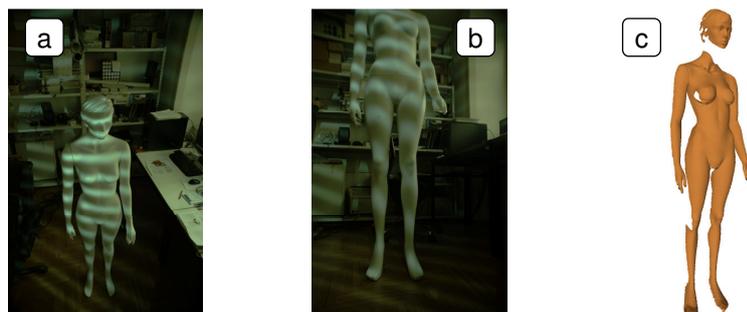


Figure 4. Reconstruction of a mannequin: (a) anterior view using top camera; (b) anterior view using bottom camera; and (c) anterior surface reconstruction.

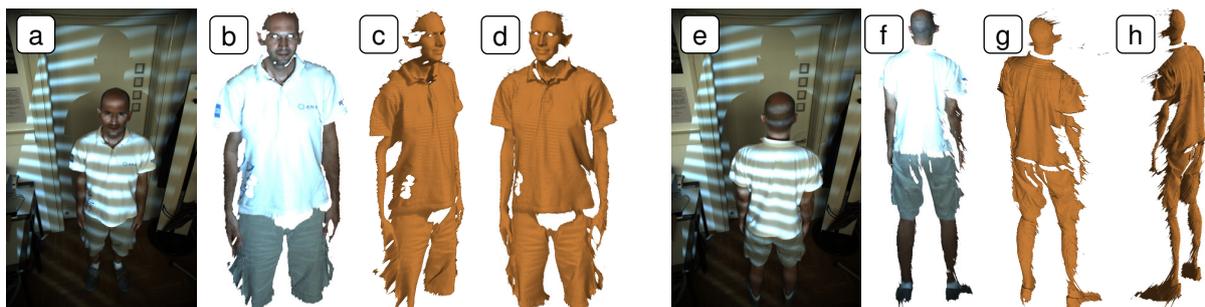


Figure 5. Reconstruction of a human male: (a) anterior view using top camera; (b) texture mapped anterior reconstruction; (c,d) two views of anterior surface; (e) posterior view using top camera; (f) texture mapped posterior view; and (g,h) two views of posterior surface.

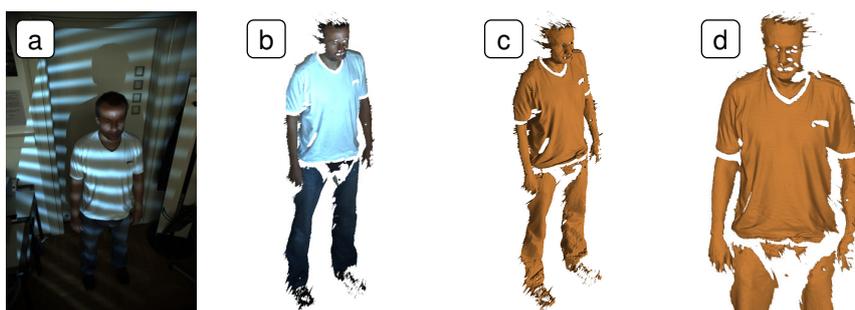


Figure 6. Reconstruction of a human male: (a) anterior view using top camera; (b) texture mapped anterior reconstruction; (c) anterior surface; and (d) close-up of anterior surface.

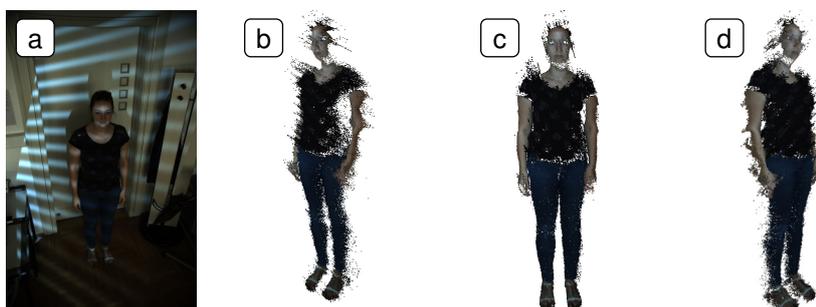


Figure 7. Reconstruction of a human female:  
 (a) anterior view using top camera; and (b,c,d) three texture mapped views of anterior reconstruction.  
 Note how dark clothes significantly affect the reconstruction which is quite noisy.

### 3.2. Experimental Recordings

We have recorded a mannequin and human volunteers using a GC+PS pattern. We have used 8 images (shifts) of a sine pattern for the phase shifting part and 8 images for the Gray code part making a total of 16 images. Considering the projector-camera synchronization is achieved successfully at 30 FPS, the entire scanning of 16 images using GC+PS takes approximately half a second.

Reconstruction of a mannequin is shown in Fig. 4 while reconstructions of human volunteers are shown in Figs. 5, 6, and 7. Note that the mannequin is pure white and is therefore easy to reconstruct while human volunteers have clothing ranging from bright to dark colors. The reconstructions shown in Figs. 4, 5, 6, and 7 are typically for GC+PS pattern. For a successfully 3D human body scanning the subject should wear bright clothing to obtain best possible reconstruction results; wearing dark clothes significantly impacts reconstruction results as is visible in Fig. 7.

### 3.3. Discussion

In the proposed software synchronization scheme the VBLANK interrupt is used as the master timing signal. Hence, using a projector with high refresh rates is a critical prerequisite to enable fast acquisi-

tion rate. The Mitsubishi EW230U-ST projector has a native refresh rate of 120 Hz; if every projected image was captured then the acquisition rate would be 120 images per second, however, the causal design requires 4 VBLANK interrupts per one acquisition cycle and wastes 3 VBLANK interrupts. This overhead was purposefully introduced to allow for non-real time operating systems: the acquisition of previous frame must succeed before the next structured light pattern image is projected. The advantage gained is that any unexpected stalls during the program execution require no special action whatsoever. When an unexpected stall occurs in practice the affected frame simply takes some additional VBLANK interrupts which users cannot notice. We note that using a true real-time operating system effectively eliminates the possibility of unexpected stalls and enables a more efficient pipelined acquisition architecture where the camera is triggered before the sensor readout and data transfer of previous frame are complete. Such pipelined architecture effectively eliminates the delay time  $t_D$  from Eq. (3) and therefore enables faster acquisition rates.

The VBLANK interrupt is the master timing signal, hence the projector must be as fast as possible; the camera only needs to achieve the frame-rate indicated by Eq. (3). A critical limiting factor for the overall acquisition rate is the input lag [26] of the projector which is the major factor of the delay time  $t_D$  (camera delay  $t_C$  is usually much smaller than 1 ms). Therefore, when designing a SL scanner using the proposed approach one should first select a fast projector with smallest possible input lag and then match all remaining components to the projector.

We have used a SL pattern comprised of 16 images to demonstrate the feasibility of the proposed software-only synchronization: the achieved 30 FPS for the image data is reduced to 1.9 FPS for the 3D data if continuous acquisition is required. Naturally, shortening the SL pattern would increase the 3D acquisition speed. The shortest possible pattern of one image would allow the same acquisition rate for both image and 3D data. A good single-image SL pattern should use color and should enable dense 3D reconstruction; a good choice is the recently proposed one-image colored SL pattern [31].

#### 4. Conclusion

We have proposed a simple software only approach to projector-camera synchronization for structured light scanning which simplifies 3D scanner construction as it makes a separate external synchronization/trigging circuit unnecessary. We have also proposed a simple method to precisely measure the delay time of the acquisition system which does not require any additional hardware.

To demonstrate the feasibility of the proposed approach we have developed a prototype system for 3D human body scanning which uses common commercial off-the-shelf components: a projector, a camera, and a desktop PC. The developed system achieves acquisition rate of 30 FPS which is comparable to standard hardware solutions.

For future work we will investigate possibilities of a non-causal software architecture for synchronization and will combine it with a pipelined projection and acquisition. When combined together those extensions could offer faster acquisition speeds than the currently achieved 30 FPS at the cost of reduced system reliability due to frame drops.

#### Acknowledgment

This work was supported in parts by the Croatian Science Foundation under Project IP-11-2013-3717 and by the Croatian-French program “Cogito”, Hubert Curien Partnership, under Project “Three-dimensional reconstruction using smartphone”. We are grateful to Simone Gasparini, ENSEIHT Toulouse, France, whose suggestions improved the content and presentation of this paper.

#### References

- [1] Tobias Jaeggli, Thomas P. Koninckx, Luc Van Gool, “Online 3D Acquisition and Model Integration”, in International Workshop on Projector-Camera Systems (PROCAMS 2003), Nice, France, October 12<sup>th</sup> 2003.
- [2] Marcelo Bernardes Vieira, Luiz Velho, Asla Sá, Paul Cezar Carvalho, “A Camera-Projector System for Real-Time 3D Video”, in 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05) – Workshops, San Diego, CA, USA, June 2005, <http://dx.doi.org/10.1109/CVPR.2005.385>

- [3] Olaf Hall-Holt, Szymon Rusinkiewicz, "Stripe Boundary Codes for Real-Time Structured-Light Range Scanning of Moving Objects", in Proceedings of Eighth IEEE International Conference on Computer Vision (ICCV 2001), Vancouver, BC, Canada, Jul 7-17, 2001, pp. 359-366, <http://dx.doi.org/10.1109/ICCV.2001.937648>
- [4] Szymon Rusinkiewicz, Olaf Hall-Holt, Marc Levoy, "Real-time 3D model acquisition", in Proceedings of the 29<sup>th</sup> annual conference on Computer graphics and interactive techniques SIGGRAPH '02, San Antonio, Texas, USA, July 21-26, 2002, pp. 438-446, <http://dx.doi.org/10.1145/566570.566600>
- [5] Ramesh Raskar, Greg Welch, Matt Cutts, Adam Lake, Lev Stesin, Henry Fuchs, "The Office of the Future: A Unified Approach to Image-Based Modeling and Spatially Immersive Displays", in Proceedings of the 25<sup>th</sup> annual conference on Computer graphics and interactive techniques SIGGRAPH '98, Orlando, Florida, USA, July 19-24, 1998, pp. 179-188, <http://dx.doi.org/10.1145/280814.280861>
- [6] Idaku Ishii, Kenkichi Yamamoto, Kensuke Doi, Tokuo Tsuji, "High-speed 3D Image Acquisition Using Coded Structured Light Projection", in Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, CA, USA, October 29-November 2, 2007, pp. 925-930, <http://dx.doi.org/10.1109/IROS.2007.4399180>
- [7] Joji Takei, Shingo Kagami, Koichi Hashimoto, "3000-fps 3-D Shape Measurement Using a High-Speed Camera-Projector System", in Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, CA, USA, October 29-November 2, 2007, pp. 3211-3216, <http://dx.doi.org/10.1109/IROS.2007.4399626>
- [8] Song Zhang, Peisen S. Huang, "High-resolution, real-time three-dimensional shape measurement", in Optical Engineering, Vol. 45, No. 12, December 2006, <http://dx.doi.org/10.1117/1.2402128>
- [9] Song Zhang, "High-resolution, Real-time 3-D Shape Measurement", Doctoral Dissertation, Stony Brook University, May 2005
- [10] Song Zhang, Peisen Huang, "High-resolution, Real-time 3D Shape Acquisition", in Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW'04), Washington, D.C., USA, June 27-July 2 2004, <http://dx.doi.org/10.1109/CVPR.2004.86>
- [11] Song Zhang, Daniel Van Der Weide, James Oliver, "Superfast phase-shifting method for 3-D shape measurement", in Optics Express, Vol. 18, No. 8, April 2010, pp. 9684-9689, <http://dx.doi.org/10.1364/OE.18.009684>
- [12] Sam Van der Jeught, Joris J. J. Dirckx, "Real-time structured light profilometry: a review", in Optics and Lasers in Engineering, 2016, <http://dx.doi.org/doi:10.1016/j.optlaseng.2016.01.011>
- [13] Kai Liu, Yongchang Wang, Daniel L. Lau, Qi Hao, Laurence G. Hasebrook, "Dual-frequency pattern scheme for high-speed 3-D shape measurement", in Optics Express, Vol. 18, No. 5, March 2010, pp. 5229-5244, <http://dx.doi.org/10.1364/OE.18.005229>
- [14] Yongchang Wang, Kai Liu, Qi Hao, Daniel L. Lau, Laurence G. Hasebrook, "Period Coded Phase Shifting Strategy for Real-time 3-D Structured Light Illumination", in IEEE Transactions on Image Processing, Vol. 20, No. 11, November 2011, pp. 3001-3013, <http://dx.doi.org/10.1109/TIP.2011.2155072>
- [15] Song Zhang, Shing-Tung Yau, "High-resolution, real-time 3D absolute coordinate measurement based on a phase-shifting method", in Optics Express, Vol. 14, No. 7, March 2006, pp. 2644-2649, <http://dx.doi.org/10.1364/OE.14.002644>
- [16] K. Herakleous and C. Poullis, "3DUNDERWORLD-SLS: An Open-Source Structured-Light Scanning System for Rapid Geometry Acquisition", arXiv:1406.6595v1 [cs.CV]
- [17] Oline Vinter Olesen, Rasmus R. Paulsen, Liselotte Højgaard, Bjarne Roed, Rasmus Larsen, "Motion Tracking for Medical Imaging: A Nonvisible Structured Light Tracking Approach", in IEEE Transactions on Medical Imaging, Vol. 31, No. 1, January 2012, pp. 79-87, <http://dx.doi.org/10.1109/TMI.2011.2165157>
- [18] Thomas P. Koninckx, Luc Van Gool, "Real-Time Range Acquisition by Adaptive Structured Light", in IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 28, No. 3, March 2006, pp. 432-445, <http://dx.doi.org/10.1109/TPAMI.2006.62>
- [19] Z. Zhang, "Microsoft Kinect Sensor and Its Effect", in IEEE MultiMedia, Vol. 19, No. 2, pp. 4-10, Feb 2012., <http://dx.doi.org/10.1109/MMUL.2012.24>
- [20] Project Tango [Online], Available: <https://get.google.com/tango/>
- [21] Structure Sensor [Online], Available: <http://structure.io/>

- [22] Ryusuke Sagawa, Ryo Furukawa, Hiroshi Kawasaki, "Dense 3D Reconstruction from High Frame-Rate Video Using a Static Grid Pattern", in IEEE Transactions of Pattern Analysis and Machine Intelligence, Vol. 36, No. 9, September 2014, pp. 1733-1747, <http://dx.doi.org/10.1109/TPAMI.2014.2300490>
- [23] Daniel Moreno, Faith Calaki, Gabriel Taubin, "Unsynchronized structured light", in ACM Transactions on Graphics (TOG) – Proceedings of ACM SIGGRAPH Asia 2015, Vol. 34, No. 6, November 2015., pp. 178:1-178:11, <http://dx.doi.org/10.1145/2816795.2818062>
- [24] Yueyi Zhang, Zhiwei Xiong, Zhe Yang, Feng Wu, "Real-Time Scalable Depth Sensing With Hybrid Structured Light Illumination", in IEEE Transactions on Image Processing, Vol. 23, No. 1, January 2014, pp. 97-109, <http://dx.doi.org/10.1109/TIP.2013.2286901>
- [25] Kovacs J., "An Overview of Genlock", Application Note No.5, Microlmage Video Systems, [Online], Available <http://www.mivs.com/documents/application-notes/an005-application-note/>
- [26] Bill Livolsi, "Input Lag and Projectors for Gaming: What it is and why you should care?", [Online], Available <http://www.projectorcentral.com/projector-input-lag.htm>
- [27] Microsoft Corporation, "Acquiring high-resolution time stamps", [Online], Available [https://msdn.microsoft.com/en-us/library/windows/desktop/dn553408\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/dn553408(v=vs.85).aspx)
- [28] Microsoft Corporation, "DXGI Overview", [Online], Available [https://msdn.microsoft.com/en-us/library/windows/desktop/bb205075\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/bb205075(v=vs.85).aspx)
- [29] J. Salvi, S. Fernandez, T. Pribanić, X. Lladó, "A state of the art in structured light patterns for surface profilometry", in Pattern Recognition, Vol. 43, No. 8, August 2010, pp. 2666-2680, <http://dx.doi.org/10.1016/j.patcog.2010.03.004>
- [30] D. Moreno and G. Taubin, "Simple, accurate, and robust projector camera calibration", in 2012 Second International Conference on 3D Imaging, Modeling, Processing, Visualization Transmission, pp. 464-471, Oct 2012, <http://dx.doi.org/10.1109/3DIMPVT.2012.77>
- [31] T. Petković, T. Pribanić, M. Đonlić, "Single-Shot Dense 3D Reconstruction Using Self-Equalizing De Bruijn Sequence", in IEEE Transactions on Image Processing, Vol. 25, No. 11, November 2016, pp. 5131-5144, <http://dx.doi.org/10.1109/TIP.2016.2603231>