A Study on Fast Three-Dimensional Shape Measurement for Moving Objects
(運動物体の高速三次元形状計測の研究)

by

Jun Chen

Graduate School of Engineering
Hiroshima University
September, 2015

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

1.1.1 High-speed Vision

With the development of digital technology, many vision systems have been applied to various fields, such as robot feedback control, industrial inspection, biomedical, and so on. However, due to the low sampling rate, most of vision systems follows standard video signals (NTSC 30 fps / PAL 25 fps), these systems are not suitable for high-speed phenomena like factory automation high-speed production line, visual servoing, and high-speed object tracking. Many research has been reported recently to improve the working frame rate of vision systems,.

To improve the sampling rate of image information transmission from photo-detectors (PD) to precessing elements (PE), vision chips implemented by integrating sensors and processors compactly has been reported, which can capture image at 1000 fps or more in real time. Bernard et al. proposed an on-chip array of bare Boolean processors with halftoning facilities and developed a 65×76 Boolean retina on a 50 mm^2 CMOS 2 μm circuit for the imager of an artificial retina [1]. Eklund et al. [2] proposed a concept of near-sensor image processing (NSIP) with a description of a method that can implement a two-dimensional (2-D) image sensor array with processing capacity in every pixel. A S^{3}PE(simple and smart sensory processing elements) vision chip architecture with each PE connected to a PD without scanning circuits is proposed by Ishikawa et al. [3, 4] for high-frame-rate image processing. Komuro et al. proposed a dynamically reconfigurable

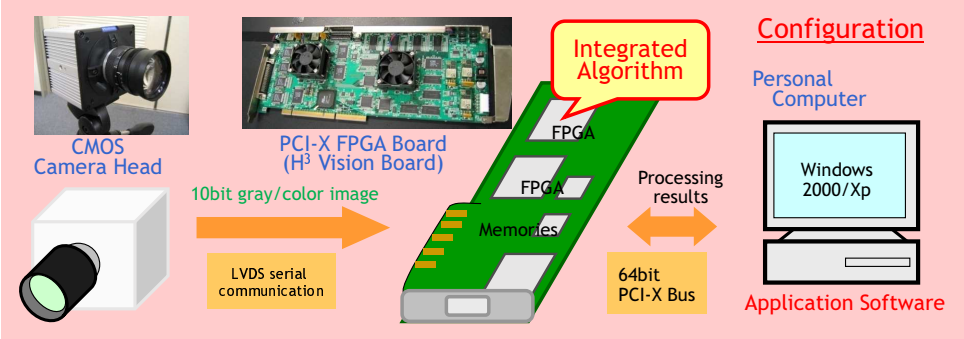


Figure 1.1: Hiroshima Hyper Human Vision

single-instruction multiple-data (SIMD) processor for a vision chip and developed a prototype vision chip based on their proposed architecture, which has 64×64 pixels in a 5.4 mm 5.4 mm area fabricated using the 0.35 μm TLM CMOS process [5]. Ishii et al. proposed a new vision chip architecture specialized for target tracking and recognition, and developed a prototype vision chip using 0.35 μm CMOS DLP/TLM(3LM) process [6].

There are also some research groups attempt to execute high-speed image processing on Field Programmable Gate Array (FPGA) board, which can connect to high-speed camera head and communicate with external device at high speed. Hirai et al. developed an flexibility FPGA-based vision system using the logic circuit to implement the image algorithm [7]. Watanabe et al. [8] proposed a high-speed a high-speed vision system that can measure moving/deforming objects in real time at a rate of 955 fps with a resolution of 256×256. A H^3(Hiroshima Hyper Human) Vision [9], as shown in Figures 1.1, is reported by Ishii et al. which can process a 1024×1024 pixels image at 1000 fps and a 256×256 pixels image at 10000 fps by implementing image processing algorithms as hardware logic on a dedicated FPGA board. In the latest two years, Ishii et al. developed a high-speed vision system called IDP Express, which can execute real-time image processing at a rate from 2000 fps (512×512 resolution) to 10000 fps (512×96 resolution), and high frame rate video recording simultaneously [10].

At present, lots of applications on the basis of high-speed vision system have been reported. Yamaoto et al. proposed a finger-tapping interface called FINGERTAP, which can estimate the contact states and forces on human fingertips in real time by using only

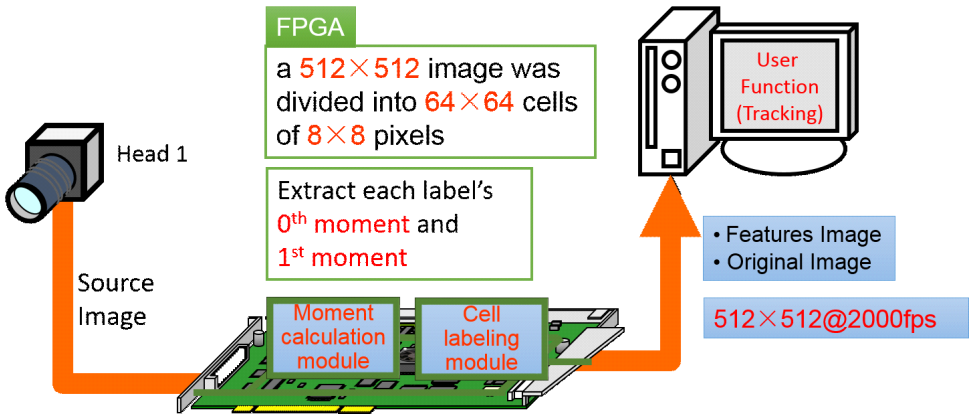


Figure 1.2: HFR multi-object extraction based on cell-based labeling

the fingertip positions measured by high-speed vision cameras [11]. Ishii et. al. proposed a concept for the real-time target tracking of vibrating objects to extract vibration image regions using digital filters at all the pixels in an image, assuming that periodic changes in image intensities exist at pixels around vibrating objects by using 1000 fps vision system [12]. Nakabo et al. developed a 1 ms vision system, which has a 128×128 PD array and an all parallel processor array connected to each other in a column parallel architecture, for 1 ms cycle-time for visual servoing and applied it to high-speed target tracking [13]. Nakabo et al. developed a 3D target tracking/grasping system, which are composed of a 1ms feedback rate using two high-speed vision systems called column parallel vision (CPV) systems and a robot hand arm [14]. Namiki et al. developed a high-speed three-fingered robotic hand controlled by a massively parallel vision system (CPV system) for robot catching [15]. Shiokata et al. proposed a strategy called ”dynamic holding” and developed a experimental robot dribbling using a high-speed multi-fingered hand and a high-speed vision system [16]. Mizusawa et al. used high-speed vision servoing to tweezers type tool manipulation by a three-finger robot hand [17]. Chen et al. reported a novel method for accurate optical flow estimation in real time for both high-speed and low-speed moving objects based on high-frame-rate (HFR) videos [18]. Gu et al. presented a HFR vision system that can automatically control its exposure time by executing brightness histogram-based image processing in real time at a high frame rate [19]. Nie et al. developed a real-time scratching behavior quantification system for laboratory mice

using high-speed vision [20]. Figures 1.2 presents a 2000 fps multi-object feature extraction system based on FPGA implementation of the cell-based labeling algorithm, which is suitable for hardware implementation and only few memory is required for multi-object feature extraction [21].

1.1.2 Vision-Based 3D Shape Measurement

Vision-based 3D shape measurement is a hot topic in computer vision with various applications ranging from industrial inspection for production line and reverse engineering to object recognition, tracking and grasp in robot control. Existing vision-based 3D shape measurement are classified into two different categories: active and passive depending on involving other devices or not.

In the active measurement method, a device is used to emit some kind of light or laser to the surface of the measured scene, while a synchronized camera is used to capture the deformation image. By using the geometric proprieties between camera and emission device and triangulation principle, the 3D shape of target scene can be obtained accurate [22]. Depending on the emission device, many measurement methods, such as time of flight, light section, and structured light have been reported. In the time-of-flight method, a laser is used to emit a pulse of light and the amount of time before the reflected light is seen by a detector is measured [23]. The light section method makes use of the light-section projecting a line-laser beam on the surface of an object The structured light method acquire the 3D shape of a target scene by illuminating it using light pattern with a certain strategy. According to the method to encode the light pattern, single-shot projection method and time-varying projection method have been reported. Figures 1.3 present an example of Gray-code projection method. In this method, a series black and white light pattern with binary information is projected onto the measured scene sequentially, such that each point on the surface of scene processes a unique binary code sequence that different with any other code sequence. Once all the correspondence between the points in light pattern and the points in captured image is established, the 3D shape coordinates can be computed based on the geometric property between camera and projector. In the

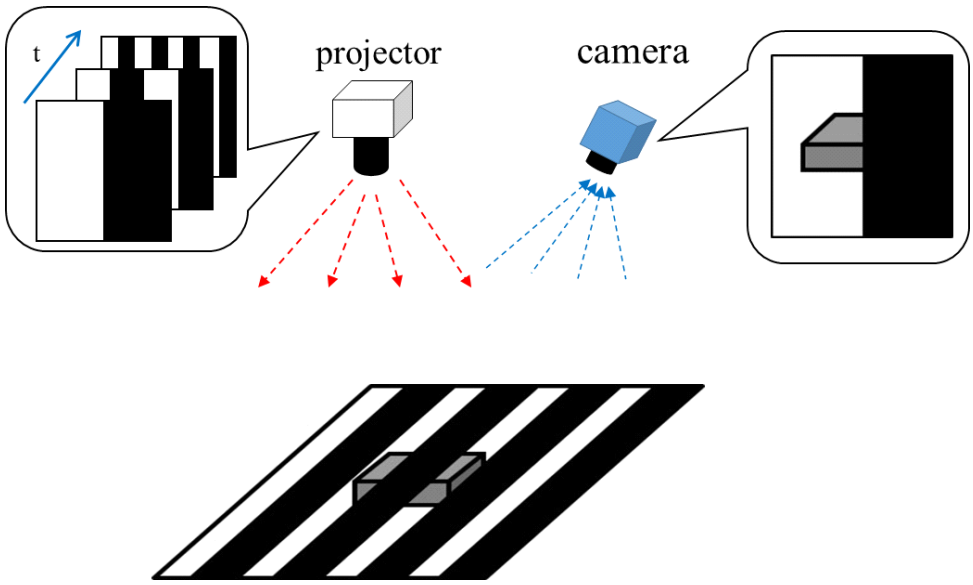


Figure 1.3: HFR multi-object extraction based on cell-based labeling

passive 3D measurement techniques, a point is imaged from various viewpoints and there is a displacement between the image point in the first image and the one in the second images. This kind of displacement is inversely proportional to distance and may therefore be used to compute 3D geometry. Given a correspondence between imaged points from two known viewpoints, it is possible to compute depth by triangulation [24]. This method has the advantages of simplicity and applicability because there is no involvement of complicate instrument [25]. However, the reconstruction quality is still a major issues for passive 3D measurement method, since the channelling existed in the finding accurate correspondence between stereo images [26].

1.2 Related Works

1.2.1 Coded Structured Light Illumination Method

Various methods [37, 30] have been investigated over the past decades to exploit 3D measurement systems focusing on simultaneous 3D extraction and rapid display, yet improving the accuracy and execution speed still pose a major challenge to improve their performance. Through projecting light patterns using certain codification strategies, a

structured light system realizes fast, robust, and accurate measurement with a significantly reduced corresponding complexity [29, 30, 31]. By characterizing them according to the domain in which corresponding features are conducted, structured light systems can be divided into the following correspondence categories [38]: spatial domain, time domain, and spacetime domain.

In the first category, techniques using one-shot projection encode the codeword purely in the spatial domain; correspondence is established by searching similar features in the light patterns and captured images via a spatial window. The Microsoft Kinect [88] is a commercial product that can acquire the 3D shapes of the scenes to be measured at 30 fps by using a spatially coded infra-red light pattern with an assumption of local smoothness. The use of Hamming color code pattern [39] was reported for dense and robust one-shot scanning; its execution time was accelerated up to 12.12 ms by using 2 GPUs. Another real-time 3D sensor based on color pattern that employs local color edge patterns for encoding was presented by Forster [34]. Some other types of light patterns with a space grid [40, 41] or parallel lines [42], where typically the correspondence is established using geometric constraints, were also reported. These systems become unstable when color or texture exists on the surface of object, while they are also not very effective in real-time processing because of their high computational complexity. Multi-spot projection is another type of single-shot projection method that can acquire the 3D shape of moving objects without considering the smoothness of the local surface [43, 44, 45]. However, the extracted spot cannot be distinguished in some exceptional cases, such as where an out-of-camera view or occlusion exists, which degrades the performance of this method.

For the correspondence in time domain, the codeword is generated by successive light patterns, such as a Gray-code light pattern [70] and sinusoidal fringe pattern [47]. In these methods, the decoding procedure is not complete until all the patterns have been projected. Kumagai [48] proposed a 3D measurement system using active stereovision based on light section and phase shift. Gao et al. [67] developed a real-time structured light 3D scanner that can generate 512×512 depth images at 500 fps using 8-bit Gray-code light patterns, which has been expanded to be a real-time projection mapping system [68].

Liu et al. reported a high-speed 3D vision with a motion compensation algorithm [51]. Some real-time 3D shape measurement systems [53, 52, 54, 55] based on phase-shifting method have also been reported recently. Using the sequential projection technique, HFR processing is feasible; however, synchronization errors occur when a moving object is measured.

Spacetime stereo realizes feature correspondence with both spatial and temporal appearance variation [38, 56]. A scanning system [57] based on time-coding boundaries between projection stripes and a boundary tracking algorithm was reported that provides range scanning of moving objects. Zhang [58] proposed a projection technique with a sequence of time-shifted patterns and a small temporal window to locate projection stripe. Weise [59] reported a fast 3D scanning system that combines stereo and active illumination. These systems reduce ambiguity and increase accuracy by exploring the benefits offered by the spatial and time domains, however, the execution time of these systems restricts the moving speed of target object.

1.2.2 Projection Mapping

Projection mapping is the concept of superimposing computer graphics onto the physical world. It has been widely used as a human-computer interface to enhance the viewing experience of the external surrounding environment or offer a natural and intuitive mode. In addition, without a high reliance on equipment restricting the human being's actions, projection mapping gives a better immersion. A number of previous studies have been explored to improve user experiences in various fields, such as surgical operation, entertainment, and industry.

Projector-guided painting [79] is an interactive system for guiding artists to paint using a multi-projector to control the appearance of the artist's canvas. The IllumiRoom reported by B. Jones et al. [80] augments the area surrounding a television with projected computer graphics to enhance gaming experiences. R. Raskar et al. [81] created a simple procedure to render the surface of an object using new calibration and illumination techniques whose effectiveness was evaluated by shader lamps, tracked object illumination,

and so on. H. Yoo and H. Kim [82] reported research on media arts using projection mapping, which has impressive performance with regard to attracting viewers' attention. N. Sakata [83] reported research on mobile interfaces using a body-worn projector and camera for providing awareness and explicit information. Similar studies, such as The Virtual Case [84] and Light Wall [85], were also reported. These above mentioned studies mainly focus on static scenes that are considered as display devices.

For projection mapping on moving objects, D. Kim [86] reported a real-time projection mapping for a flexible object on the musical stage, which projects various textures, patterns, and images onto a moving actor's costume. The Beamatron [87], a steerable AR system, can project graphics onto the surfaces of moving objects in real-time by using a Microsoft Kinetic [88] as a depth sensor. A projector-based augmented-reality system [89] has been reported to offer an intuitive real-time intra-operative orientation in interstitial therapy. In order to reduce the delay time in projection mapping systems for dynamic scenes, a time-delay compensation algorithm [90] and a screen object-tracking algorithm [91] have been presented. The Lumpien system [77, 92], consisting of a projector and a Saccade Mirror [93], can project light patterns onto high-speed moving objects, like a bouncing ball, by using hue-saturation-value (HSV) information to detect the target objects without considering their 3D shapes.

1.3 Outline of Thesis

The thesis is structured as followed.

Chapter 1 introduces the development and applications of high-speed vision, and vision based 3-D shape measurement. The related works of coded structured light illumination method and projection mapping are described in detail.

Chapter 2 presents concepts of blink-spot projection method, motion-compensated stripe projection method, real-time projection mapping system and fast 3D shape scanner.

Chapter 4 describes the algorithm of blink-spot projection method in detail, and the performance of this method is verified by implementation on a real-time, HFR 3D shape measurement system, which consists of a high-speed vision platform and an LCD

projector.

Chapter 3 introduces a motion-compensated stripe projection method and implemented scanner that can remarkably reduce the synchronization errors in the structured-light-based measurement, which are encountered in the projection of multiple light patterns with different timings.

Chapter 5 reports on the development of a projection mapping system that can project RGB light patterns that are enhanced for 3D scenes using a GPU-based HFR vision system synchronized with HFR projectors. The depth image processing is accelerated by installing a GPU board for parallel processing of a gray-code structured light method using infrared (IR) light patterns projected from an IR projector. Using the computed depth images, suitable RGB light patterns to be projected are generated in real time for enhanced application tasks.

Chapter 6 reports on the development of a fast 3D shape scanner that can output 3D video at 250 fps using two HFR camera-projector systems with an implementation of 10-bit Gray co de light pattern encoded in both horizontal and vertical. The 3D data, which is extracted by the two camera-projector systems and accelerated by installing a GPU board for parallel processing of structured light illumination, are registered together to obtain an entire shape.

Chapter 7 will summarize the contributions of this study and discuss the future work. In future work, we plan to combine the blink-spot projection method and motion-compensated stripe projection method to extend its applicability to various applications such as robot feedback control, fast recognition, and industrial inspection.

Chapter 2

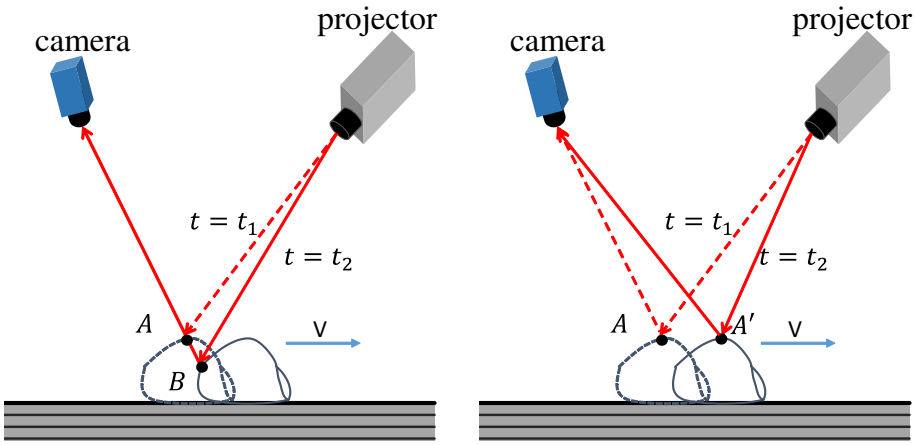
Concepts

2.1 Concept of Motion-compensated Stripe Projection Method

When the 3-D shape of moving scene is observed using the coded structured light approach with multiple light patterns projected at different timings, the synchronization errors are caused by misreading the light patterns projected at different points on the object one as those at the same point. To minimize such synchronization errors, we introduce a motion-compensated coded structured light method for fast and accurate 3-D shapes measurement of high-speed moving scenes, which is based on the following concepts:

(a) Motion-compensated structured light coding

In the conventional coded structured light approach with multiple light patterns with different timings, the 3-D shape is computed by triangulation based on the space code value of structured light patterns, which are projected from the projector, and the pixel position on the image sensor. When there is a relative motion between the camera-projector coordinate system and the object one, synchronization errors would occur in space code acquisition, since the projected light patterns at different multiple points on the object to be measured are misregarded as the same point in space code acquisition. Fig. 2.1(a) illustrates the synchronization errors in space code acquisition when the object to be measured is moving. The light patterns projected at points A and B on the object, which are observed at the same pixel on the image sensor with different timings, are mis-corresponded in space code acquisition; synchronization errors depend on the smoothness of the object surface as well as the apparent motion of the object in the images.



(a) conventional approach (b) our approach

Figure 2.1: Motion-compensated structured light approach.

To minimize such synchronization errors, which are independent on the depth gaps on the object to be measured, our motion-compensated structured light approach introduces an idea that the light patterns projected on the same position on the object are tracked by predicting its corresponding pixel positions with different timings based on the relative motion parameters between the camera-projector coordinate system and the object one. Fig. 2.1(b) illustrates our motion-compensated structured light approach. The light patterns at the points A and A' on the object, which are observed at the different pixel positions predicted with the object's motion, are corresponded in space code acquisition. Here, slight ambiguity in the structured light coding is remained, however, it is not dependent on the smoothness of the object surface even when there are depth gaps in the measured scene.

(b) HFR projection and video processing

When image displacements between frames in camera-projector systems are small, synchronization errors are considerably decreased as reported in [67]; however, a certain time duration is required to project dozens of light patterns at a high frame rate. (e.g. positive and negative 8-bit light pattern projection at 1000 fps requires 16 ms.) Therefore, we introduce the motion-compensated structured light approach to solve synchronization errors during such a certain time duration on HFR camera-projector systems, where the relative motion between the camera-projector coordinated system and the object one can

be linearly predicted, assuming small frame-to-frame image displacement. Moreover, HFR camera-projector systems can assure much wider measurable area in the 3-D structured light measurement when the object to be measured is moving, compared with that on camera-projector systems operating at dozens of frames per second. Here we can acquire space code value at the pixel only where all the projections are observable at multiple frames; the measurable area in the structured light coding corresponds to the overlapped projectable area at different timings, which is depending on the object's motion and the frame interval of the camera-projector system.

2.2 Concept of Blink-spot Projection Method

I propose an improved multi-spot projection method (hereafter referred to as the "blink-dot projection method"). This method simultaneously extracts the ID numbers of the multiple spots projected onto a measured object in a camera view for robust and accurate 3-D measurement by projecting several multi-spot patterns at different timings. Figure 2.2 shows the concept of our blink-dot projection method.

(a) Multi-spot extraction accelerated by hardware logic

In general, the computational complexity of multi-object extraction is of the order of $O(MN)$ for an $M \times N$ pixel image. To accelerate this process for the localization of

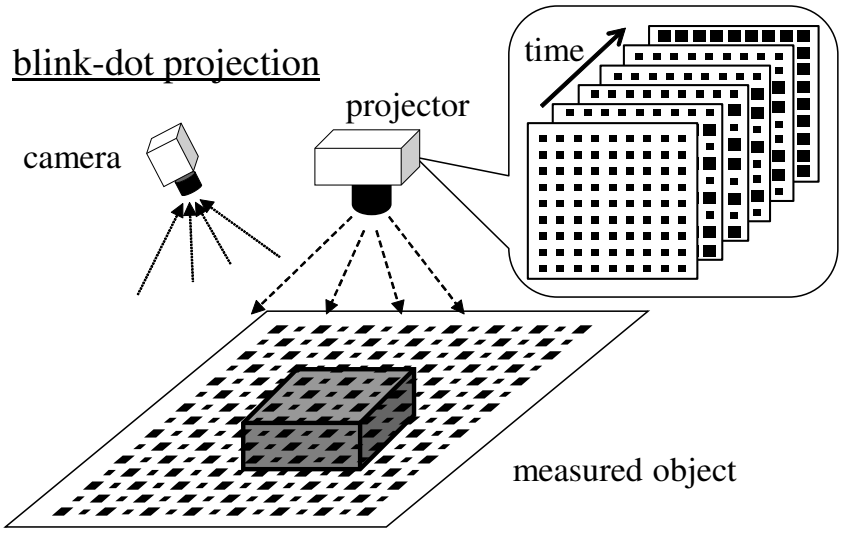


Figure 2.2: Concept of blink-spot projection method.

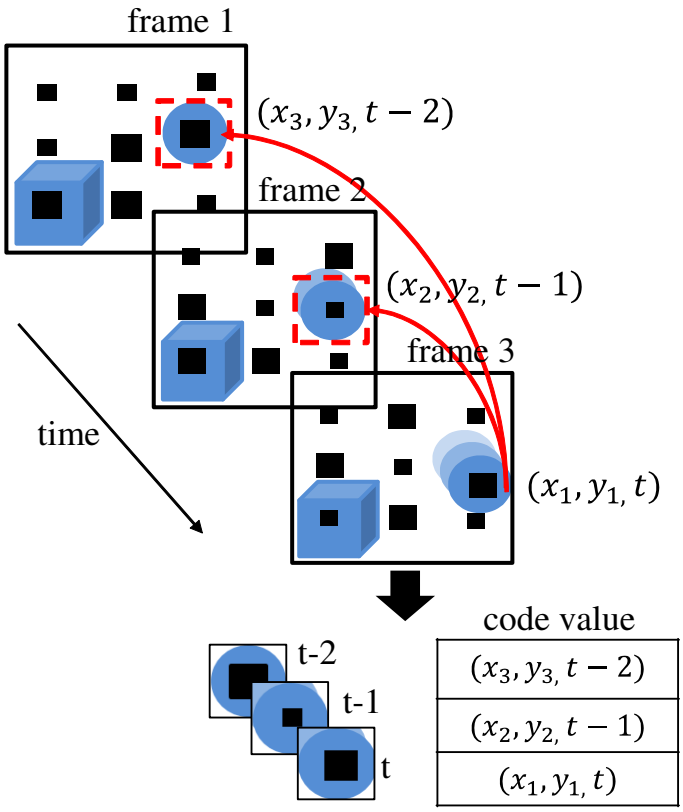


Figure 2.3: Process of frame-to-frame correspondence.

the projected spots, I implement a pixel-level connected component labeling algorithm [35, 36] on a field-programmable gate array (FPGA)-based HFR vision platform. This algorithm is designed such that it is integrated with hardware logic to simultaneously extract the moment features of multiple objects in an image.

(b) Frame-to-frame correspondence with HFR video

I can assume that the displacement of spots between several continuous frames is considerably small in an HFR video. Thus, in the previous frames, spots corresponding to those in the current frame can be searched in smaller areas around the spots extracted in the current frame. The computational complexity of the tracking process has the order of $O(L)$, where L represents the number of spots in the captured image. Fig. 2.3 describes the matching process among three continuous frames.

(c) Robust ID number decoding

In the blink-spot projection method, the multiple spots of different sizes periodically

blink to display their ID numbers, and these ID numbers indicate the directions of the spot beams projected by the projector. A robust ID number decoding algorithm with filters in both the space and temporal domain is proposed to reduce the ambiguity in frame-to-frame spot correspondence.

Chapter 3

Motion-compensated Stripe Projection Method

3.1 Introduction

Vision-based three-dimensional (3-D) shape measurement technology is a hot topic with various applications such as industrial inspection, object recognition and robot perception. Among the 3-D shape measurement techniques, the coded structured light method [61, 62, 63] is a well-known active measurement method based on a camera-projector system; it can acquire the 3-D shape information with pixelwise accuracy by projecting multiple light patterns onto the object surface to be observed and capturing its illuminated scenes. However, in most of the structured light vision systems, the accuracy in 3-D shape measurement becomes worse when there is a fast relative motion between the camera-projector coordinate system and the object one, due to the synchronization errors in projecting multiple light patterns at different timings.

To avoid such synchronization errors in 3-D shape measurement, several one-shot-projection structured light methods have been proposed for observing moving scenes. Guan et al. [64] obtained 3-D shapes by projecting a single composite light pattern combined by multiple slits with random cuts. Sakashita et al. [65] proposed a one-shot projection method that can obtain the 3-D shape of a target object by projecting a single color grid pattern onto the object. Microsoft's Kinect [66] is a commercial product that can acquire 3-D shapes at 30 fps by using a spatially coded infrared light pattern. Most of these one-shot-projection methods should assume that the local surface of the object to be measured is smoothly curved, and these methods cannot guarantee the pixelwise

accuracies in 3-D shape measurement.

Many structured light 3-D shape measurement systems have used image sensors and projectors operating at dozens of frames per second, and synchronization errors in projecting multiple light patterns decrease their accuracies in 3-D shape measurement when moving scenes are observed, because the accuracies are restricted by the frame rate of the camera-projector system. Recently many HFR vision systems are capable of real-time processing at 1000 fps or more have been developed, and several studies have been reported for HFR 3-D shape measurement. Gao et al. [67] developed an HFR camera-projector system that can simultaneously obtain depth images at 500 fps based on the coded structured-light method using multi light patterns, and this system was expanded for real-time projection mapping [68]. These studies indicated that HFR camera-projector systems could overcome the trade-off between accuracy and synchronization errors in 3-D shape measurement, however, there were still synchronization errors when fast-moving objects were observed. Chen et al. [69] proposed a blink-dot projection method that can accurately identify the 3-D positions and ID numbers of blink dots even when rapidly moving scenes are observed, and showed its performance when 15×15 blink-dot projection was conducted at 1000 fps on the HFR camera-projector system, however, its spatial resolution in 3-D shape measurement depended on the number of blink dots.

In this study, we propose a motion-compensated coded structured light method with time-variant light projections that can compute depth images with pixelwise accuracy even when fast-moving scenes are observed; synchronization errors in projecting multiple light patterns are reduced by generating space code images using motion-compensated illuminated scenes, which are captured at different timings. Section 3.2.1 describes the concept of our approach and gives its outline. Section 3.2.2 presents the configuration of our HFR structured light vision system, which is mounted on a 6-DOF manipulator, and outlines the implementation of our method on it for computing 512×512 depth images at 500 fps. In Section 3.3, we performed several experiments when the 6-DOF manipulator moves rapidly so that 3-D shapes of many objects on a plane are sequentially captured, and verified the effectiveness of our motion-compensated approach for robot-mounted structured light system.

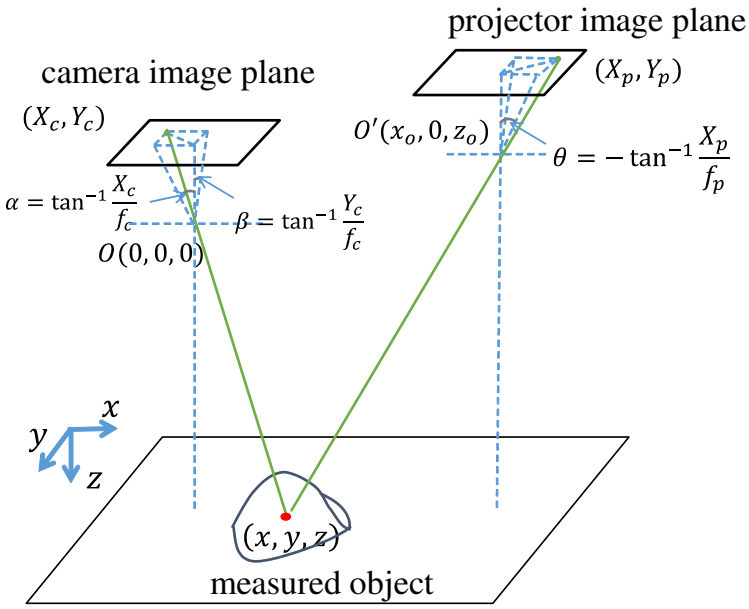


Figure 3.1: Geometry of the camera-projector system.

3.2 Motion-compensated Stripe Projection Method

3.2.1 Algorithm

In this study, we assume that an HFR camera-projector system is mounted on the end of a high-speed robot manipulator, and the camera-projector system on the manipulator observes static scenes when the camera-projector system is rapidly moved on a two-dimensional planar trajectory; there is a large relative motion between the camera-projector coordinate system and the object one.

(a) Definition of coordinate system

The coordinate system of the camera-projector system we considered is shown as Fig. 3.1. The world coordinate system is expressed as the xyz-coordinate system with the center of the camera lens as its origin O. The X_cY_c-coordinate system on the camera image plane is perpendicular to the optical axis of camera lens, and its origin is located at the intersection with the optical axis of the camera lens. The X_c- and Y_c-axes are parallel to the x- and y-axes, respectively. The distance between the X_cY_c-plane and the optical center is f_c. The X_pY_p-coordinate system on the projector image plane is perpendicular to the optical axis of the projector lens, while the distance from the X_pY_p-plane to the optical

center of the projector lens is f_p. The X_p- and Y_p-axes are parallel to the x- and y-axes, respectively. In the xyz-coordinate system, the optical center of the projector lens is at $O'(x_0, 0, z_0)$.

(b) Algorithm for depth images extraction

To minimize synchronization errors in the structured light coding, we describe a motion-compensated structured-light algorithm, which is an improvement for motion compensation over Inokuchi's method [70].

(1) Projection of gray-coded light patterns

N pairs of positive and negative light patterns with the N-bit gray code are projected in the order of $\{g_0, g_1\}, \cdots$, and $\{g_{2N}, g_{2N+1}\}$. The generation of these patterns is expressed as follows:

$$g_{2i}(X_P, Y_P) = \left\lfloor 2^i \cdot Y_P/M + 0.5 \right\rfloor \bmod 2, \tag{3.1}$$

$$g_{2i+1}(X_P, Y_P) = \overline{g_{2i}(X_P, Y_P)} \quad (i = 0, \cdots, N-1), \tag{3.2}$$

where (X_P, Y_P) represents the point on the projector image plane. M describes the unit width of these patterns in the X_P direction. $\lfloor x \rfloor$ denotes the greatest integer less than or equal to x. i denotes the LSB and MSB order in gray code.

(2) Image acquisition of projected patterns

A gray-level image of $I_x \times I_y$ pixels is captured at time $t = k\tau$ as follows:

$$I(X_C, Y_C, k) = \mathrm{Proj}(g_{k \bmod 2N}(X_P, Y_P)), \tag{3.3}$$

where (X_C, Y_C) represents the point on the camera image plane. k indicates the frame number of the captured image. τ is the frame interval.

(3) Binarization with motion compensation

A pair of captured images $I(X_C, Y_C, 2k')$ and $I(X_C, Y_C, 2k'+1)$ are differentiated to avoid ambiguities due to non-uniform brightness at interval of 2τ. Here the differentiation without motion compensation could cause ambiguous frame-to-frame-corresponding when there is an apparent motion in the image. In order to reduce such errors, we obtain

the binarized image with motion compensation by thresholding the two images using θ_b as follows:

$$G(X_C, Y_C, 2k'+1) = \begin{cases} 1 & (I(X_C, Y_C, 2k'+1) - I(X_C - \tau \cdot V_{X_C}^{2k'}, \\ & \quad Y_C - \tau \cdot V_{Y_C}^{2k'}, 2k') > \theta_b) \\ 0 & (I(X_C, Y_C, 2k'+1) - I(X_C - \tau \cdot V_{X_C}^{2k'}, \ , \\ & \quad Y_C - \tau \cdot V_{Y_C}^{2k'}, 2k') < -\theta_b) \\ \phi & \text{(otherwise)} \end{cases} \tag{3.4}$$

where $(V_{X_C}^{2k'}, V_{Y_C}^{2k'})$ is the relative motion between the camera-projector coordinate system and the object one in the X_C and Y_C directions on the image plane at frame $2k'$. We assume that $(V_{X_C}^{2k'}, V_{Y_C}^{2k'})$ is given to the camera-projector system by calculating the camera motion from the joint-angles of the manipulator. ϕ denotes the ambiguous binarization state arising due to occlusion from the camera.

(4) Gray-to-binary conversation with motion compensation

$G(X_C, Y_C, 2k'+1)$ is converted with a pure N-bit binary code by using motion-compensated images $G'(X_C, Y_C, 2(k'-i)+1)$ at the current and previous frames $2(k'-i)+1$ $(i = 0, 1, \cdots, N-1)$ as follows:

$$B(X_C, Y_C, 2k'+1) = \left(\sum_{i=0}^{k' \bmod N} G'(X_C, Y_C, 2(k'-i)+1) \right) \bmod 2, \tag{3.5}$$

where unmeasurable state is considered when there are two or more ambiguous binarizations among the $2N$ frames; 0 is substituted for ϕ in the case of one ambiguous binarization.

Assuming that the relative motion between the camera-projector coordinate system and the object one on the image plane at frame $2k'+1$ is $(V_{X_C}^{2k'+1}, V_{Y_C}^{2k'+1})$, we consider that the light pattern projected on the pixel (X_C, Y_C) at frame $2k'+1$ corresponds to the point projected on its linearly-predicted pixel location at frame $2(k'-i)+1$, $(X_C - \sum_{j=0}^{i-1} 2\tau \cdot V_{X_C}^{2(k'-j)+1}, Y_C - \sum_{j=0}^{i-1} 2\tau \cdot V_{Y_C}^{2(k'-j)+1})$ $(j = 0, 1, \cdots, i-1)$. Thus the motion-compensated

image $G'(X_C, Y_C, 2(k'-i)+1)$ is given as follows:

$$G'(X_C, Y_C, 2(k'-i)+1) = G(X_C - \sum_{j=0}^{i-1} 2\tau \cdot V_{X_C}^{2(k'-j)+1}, Y_C - \sum_{j=0}^{i-1} 2\tau \cdot V_{Y_C}^{2(k'-j)+1}, 2(k'-i)+1), \tag{3.6}$$

where the velocity $(V_{X_C}^k, V_{Y_C}^k)$ is updated at every frame.

(5) Space code generation

A space-code image $C(X_C, Y_C, 2k'+1)$ is obtained by using the N-bit binary images $B(X_C, Y_C, 2(k'-i)+1)$ $(i = 0, 1, \cdots, N-1)$ at the current and previous frames as follows:

$$C(X_C, Y_C, 2k'+1) = \sum_{i=0}^{N-1} 2^{(k'-i) \bmod N} \cdot B(X_C, Y_C, 2(k'-i)+1), \tag{3.7}$$

In this study, a 3×3 median filter is applied to the space code image.

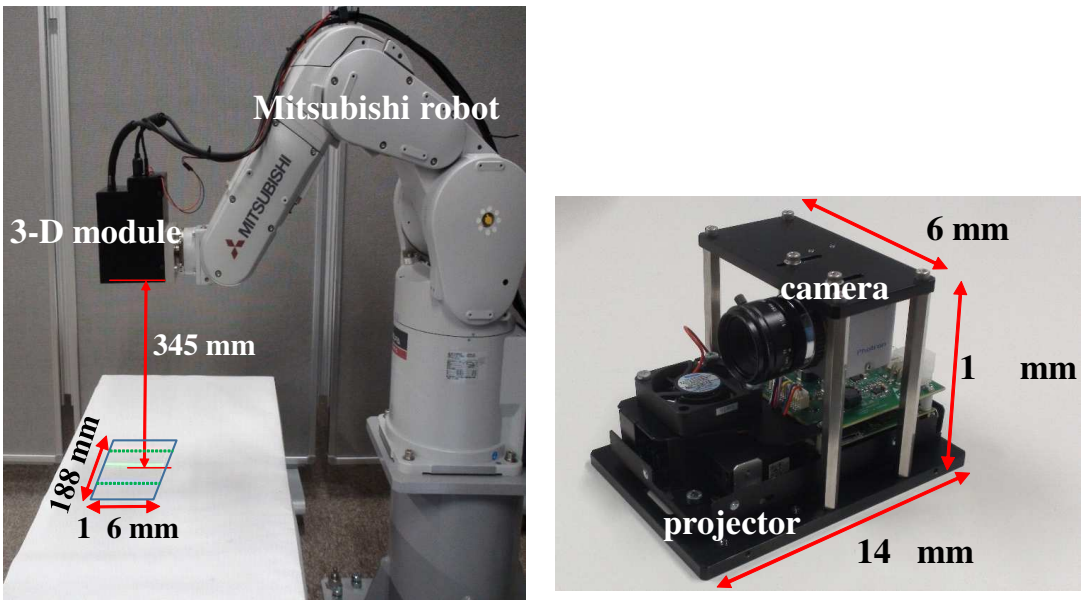
(6) Triangulation

The space code image $C(X_C, Y_C, 2k'+1)$ is transformed to a depth of $z = D(X_C, Y_C, 2k'+1)$ by solving a simultaneous equation with a 3×4 camera transform matrix T_C and a 2×4 projector matrix T_P.

$$H_C \begin{pmatrix} X_C \\ Y_C \\ 1 \end{pmatrix} = T_C \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}, \tag{3.8}$$

$$H_P \begin{pmatrix} C(X_C, Y_C, 2k'+1) \\ 1 \end{pmatrix} = T_P \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}, \tag{3.9}$$

where H_C and H_P are prior calibrated parameters which are obtained by using the calibration method reported in [71].



(a) system setting (b) HFR structured light vision

Figure 3.2: System overview.

3.2.2 Robot-mounted HFR Structured Light Vision

We implemented our motion-compensated structured light algorithm on the HFR camera-projector system, which was used for fast 3-D measurement using blink-dot projection [69]. It includes a 3-D module consisting of an HFR camera head and an HFR projector, an IDP Express board [72], a personal computer (PC) equipped with a GPU board, and a high-speed 6-DOF robot manipulator. Our system can generate 512×512 depth images in real-time at 500 fps by using 8-bit gray code light patterns. Its overview is shown as Fig. 3.2.

The 3-D module is composed of a DLP LightCrafter projector (Texas Instruments Inc., US) and a monochrome camera head (Photron Ltd., Japan); its dimensions and height are 140×96×100 mm and 960 g, respectively. The DLP LightCrafter is a development kit enabling high-speed binary light pattern projection using a DMD device, which can project hundreds of 608×684 binary patterns at 1000 fps or more, in synchronization with an external system. The camera head can capture 8-bit gray-level images of 512×512 pixels at 2000 fps, and the processed image results could be mapped in the PC memory at the same frame rate. The optical center of the projector is at $(x_0, 0, z_0)$ = (50 mm, 0, 20 mm) in the xyz-coordinate system illustrated in Fig. **??**. The focal lengths

Table 3.1: Execution time of 3-D shape measurement.

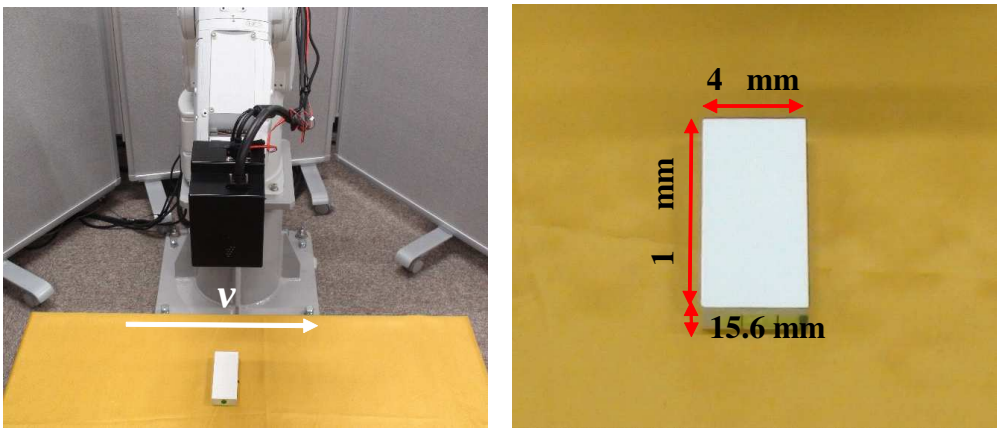
	Time[ms]
(2) Image acquisition of projected patterns	0.03
Transfer to GPU	0.08
(3) Binarization with motion compensation	0.04
(4) Gray-to-binary conversion with motion compensation	0.08
(5) Space code generation	0.02
(6) Triangulation	0.01
Transfer to PC memory	1.21
Total time	1.47

of the camera and projector lenses were $f_c = 16$ mm and $f_p = 12.6$ mm, respectively. The distance from the 3-D module to the level surface is $z = 345$ mm, and the projector forms a 608×684 light pattern in a 188×106 mm square on the level surface. On the level surface, the area of the captured image is a 106×106 mm square, and depth information over this region can be obtained. The 3-D module is mounted on the end of a 6-DOF industrial robot manipulator RV-4F (Mitsubishi Electric Co., Japan).

The IDP Express board is an HFR image capture and processing platform, which has two camera inputs for camera heads, and a user-specific FPGA for hardware implementation of image processing algorithms. The IDP Express board has trigger I/Os for synchronization with external devices.

A GPU board, Tesla C1060 (NVIDIA Co., California, US) is mounted on a PC. The Tesla C1060 is a computer processor board based on the NVIDIA Tesla T10 GPU; it is capable of a processing performance of 933 Gflops/s using 240 processor cores operating at 1.296 GHz and a bandwidth of 102 GB/s for its internal 4 GB memory. The PC has the following specifications: ASUSTeK P6T7 WS main board, Intel Core i7 3.20 GHz CPU, 3 GB of memory, two 16-lane PCI-e 2.0 buses, and Windows XP Professional 32 bit OS.

The subprocesses (2)–(6) are accelerated for HFR depth image processing by executing them in parallel with 512 blocks of 1×512 pixels on the GPU board. Table 6.1 shows the total execution time consumed by our proposed system, which includes the transfer time from the PC memory to the GPU board for a 512×512 input image, and that from the GPU board to the PC memory for processed depth images. The total time is within 1.47 ms, and we confirm depth image processing of 512×512 images in real-time



(a) experimental scene (b) measured cuboid

Figure 3.3: Experimental scene and object.

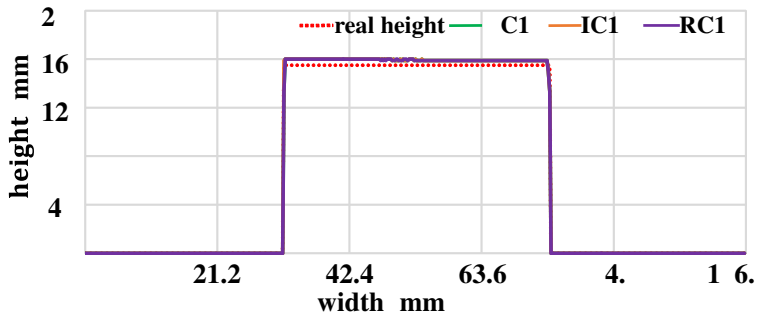
at 500 fps.

3.3 Experiments

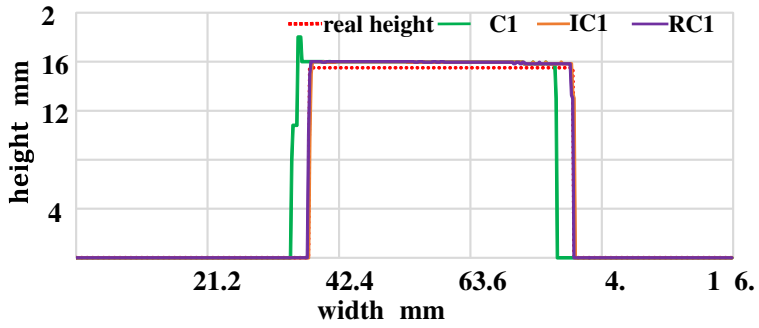
To show the effectiveness of our motion-compensated coded structured light approach, two experiments are conducted with 8-bit gray code patterns using 1000 fps video sequences. In the experiments, “IC1K” and “RC1K” refer to as the motion-compensated coded structured light methods. In “IC1K”, the relative motion between the 3-D module and the measured scene was obtained by calculating the centroid position of the depth image based on image processing; the relative motion is obtained from the joint angle information of the robot manipulator in “RC1K”. “NC1K” refers to as the conventional coded structured light method without motion compensation. “RC1K” was conducted in real time while the results of the other two methods were calculated offline using the input images captured in the experiment of the ”RC1K”. The frame interval and the exposure time were 1 ms and 0.6 ms, respectively; the binarization threshold was $\theta_b = 5$.

3.3.1 Accuracy Verification

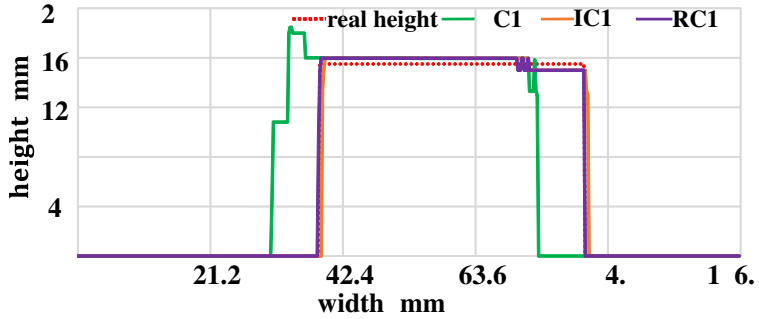
To evaluate the accuracy of the 3-D shape measurement using our motion-compensated coded structured light method, we observed the 3-D shape of a cuboid of 15.6 mm height on the level surface when the 3-D module was manipulated at different speeds of 0.0,



(a) v = 0.0 m/s



(b) v = 0.2 m/s



(c) v = 0.6 m/s

Figure 3.4: Measured depth profiles operating at different speeds.

0.2, and 0.6 m/s. Fig. 3.3 shows the experimental scene and measured object. Fig. 3.4 shows the intersected profiles of the measured 3-D shapes when the 3-D module was operated at different speeds, which was intersected at y = 50 mm. In the figures, the profiles measured using our motion-compensated method ("IC1K" and "RC1K") are plotted, compared with the actual depth profile of the cuboid and the profile measured using the conventional coded structured light method without motion compensation ("NC1K"). In Fig. 3.4(a), it can be seen that all the measured depth profiles, "IC1K", "RC1K", and "NC1K", are similar, and they match with the actual depth profile of the cuboid. This is

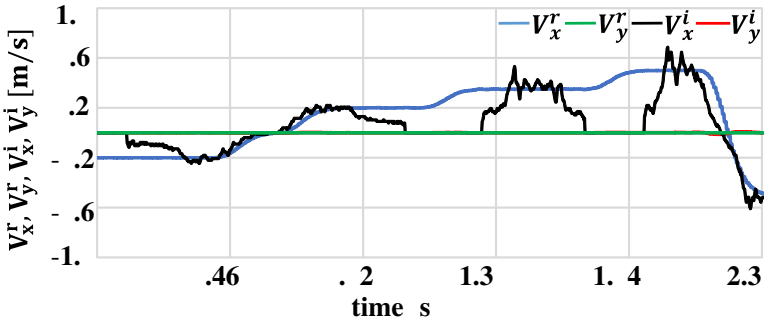


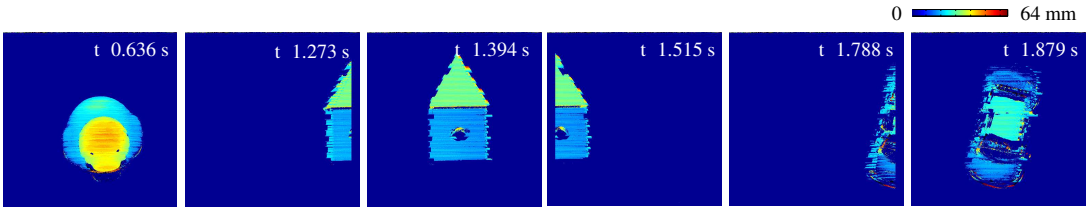
Figure 3.5: Velocity of relative motion.

because there was no synchronization error in structured light coding in all the methods according to no relative motion between the 3-D module and the cuboid. In Fig. 3.4(b) and (c), impulsive noises are observed around the edge of the cuboid in the depth profile of "NC1K", while the depth profiles of "IC1K" and "RC1K" are almost matched with the actual depth profile of the cuboid. The impulse noises observed in "NC1K" become more serious as the relative motion becomes larger, while the motion-compensated coded structured light algorithms can reduce such impulsive noises due to synchronization errors in structured light coding and the measurement errors of "IC1K" and "RC1K" were within 1 mm even when the 3-D module was fast-moving.

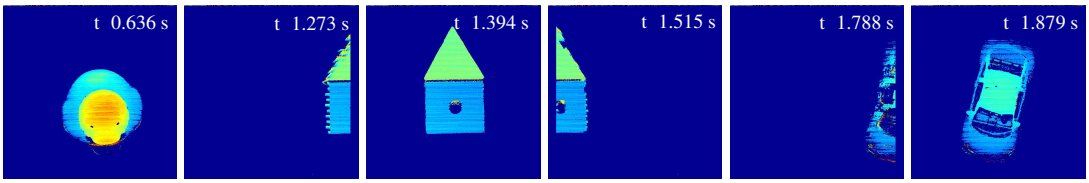
3.3.2 Static scene observation

Next, the 3-D module was horizontally moved at variable speeds of 0.2, 0.35, and 0.50 m/s, in the x direction, and it observed at three objects with different heights on the level plane; a 45-mm-tall toy duck, a 27-mm-tall toy car, and a 31-mm-tall toy house. Fig. 3.5 shows (a) experimental scenes captured by a standard digital video camera, (b) depth images measured with "NC1K", (c) depth images measured with "IC1K", and (d) depth images measured with "RC1K", which were taken at t = 0.636, 1.273, 1.394, 1.515, 1.788 and 1.879 s. Fig. 3.5 shows the relative motion parameters, which are used in the motion-compensated coded structured light methods; (V_x^r, V_y^r) are the motion parameters in "RC1K", which are calculated using the joint-angle information of the robot manipulator; (V_x^i, V_y^i) are the motion parameters in "IC1K", which are obtained by calculating the centroid of the depth image. In Fig. 3.5(b), the depth images measured without mo-

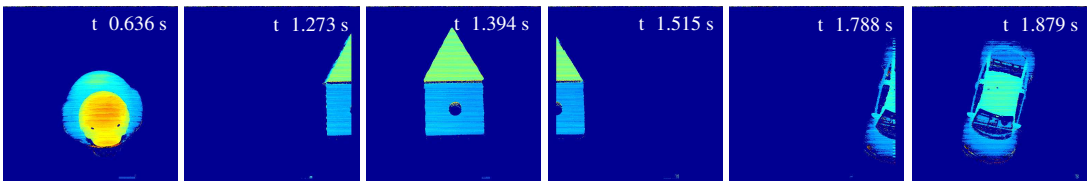
(a) experimental scenes



(b) depth images (“NC1K”)



(c) depth images (“IC1K”)



(d) depth images (“RC1K”)

Figure 3.6: Experimental results of static scene observation.

tion compensation had strong impulsive noises and many unmeasurable pixels around the edge of the objects especially when the speed of the 3-D module became larger. Compared with the depth images estimated without motion compensation, the measurement errors were considerably reduced in those with motion compensation in Fig. 3.5(c) and (d), since the same point on the object could be corresponded in space code acquisition by tracking it in the image. Here certain impulsive noises were observed in “IC1K” at t = 1.273, 1.394, and 1.788 s when the objects were not completely appeared in the image, while there were little noises in “RC1K” at all the times. This is because the centroid information of the depth image did not correctly correspond to the actual position of the object, when the object was not completely appeared in the image, and the relative motion estimated in “IC1K” was quite different from the manipulator’s motion as shown in Fig. 3.5.

3.4 Concluding Remarks

In this study, we reported a motion-compensated coded structured light method for fast and accurate 3-D shape measurement of moving scenes. To evaluate the effectiveness of our method, we implemented it on an HFR camera-projector system that can compute 512×512 depth images in real time at 500 fps. Its real-time performance was verified by showing experimental results when the 3-D module of the HFR camera-projector system was mounted on a fast-moving 6-DOF robot manipulator as a sensing head. In the future, we plan to improve our HFR structured light vision system for more robust and faster 3-D shape measurements, and to extend applications of our motion-compensated coded structured light approach for various human-computer interactions and robotic perceptions.

Chapter 4

Blink-spot Projection Method

4.1 Introduction

With the development of digital technology, real-time three-dimensional (3D) shape measurement is becoming increasingly important for applications in many fields. Much research on the basis of 3D sensors, such as footstep planning for humanoid robots [27] and robotic bin-picking [28], has been reported recently. When the camera view includes fast motion, the extraction of a depth image has to be sufficiently fast to allow more information to be obtained. To achieve high measurement accuracy and reduce time consumption, coded structure light illumination [29, 30, 31], a well-known active 3D shape measurement approach, is widely used. In this method, to reconstruct the 3D scene coded light patterns are projected onto the scene to be measured, while a synchronized camera is employed to capture deformation images from a different perspective. Based on this concept, many scanning systems implemented with a one-shot or sequential projection technique have been reported over the past decades [32, 88, 34]. Systems based on one-shot projection are realized by using a complicated image processing algorithm; however, acceleration of the process presents a challenge. It is feasible that a system based on sequential projection can be effective at a high frame rate (HFR); however, the synchronization errors caused by the light patterns projected at different points being misread as those projected at the same one limit the measurement accuracy.

Our goal is to realize a robust real-time 3D measurement system that reconstructs an accurate 3D scene at hight-speed, without imposing restrictions on the object's motion for

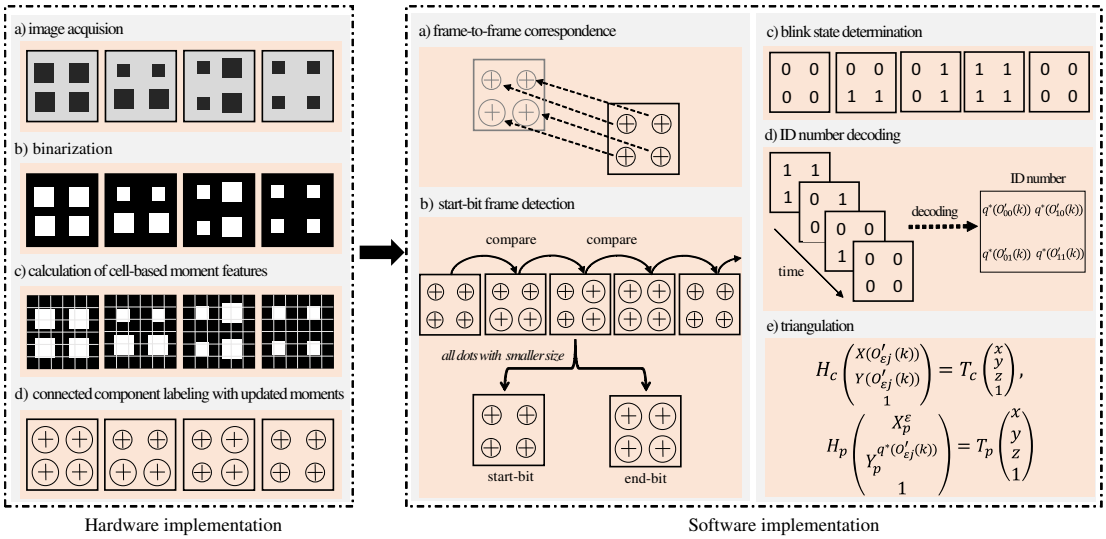


Figure 4.1: Flowchart of blink-spot projection method.

which they are effective. To this end, we propose a novel multi-spot projection method called the "blink-spot projection method". In this method, a certain number of multi-spot light patterns, where large and small spots alternatively blink at different timings corresponding to their ID numbers, is projected by an HFR projector onto the scenes being measured. The image sequences captured by a camera synchronized with the projector are simultaneously processed to extract the spots' ID numbers to achieve robust and accurate 3D shape measurement.

In Section 4.2.1, an outline of our proposed algorithm is given. In Section 4.2.2, the configuration of our HFR camera-projector system, which can obtain the 3D positions of 16×16 spots in real time at 500 fps, the hardware implementation of the image feature extraction algorithm and the evaluated execution time are presented. In Section 4.3, several experimental results for the measurement of moving scenes, the evaluated measurement accuracy of our blink-spot projection method, and the verification of its performance are given.

4.2 Blink-spot Projection Method

4.2.1 Algorithm

To realize real-time 3D shape measurement using blink-spot projection, we implemented the following algorithms: (1) generation of blink-spot projection patterns to determine the positions and blinking states of the spots; (2) moment feature extraction of multiple spots, which is a pixel-level process accelerated through the implementation of an extended cell-based labeling algorithm [36]; and (3) spot tracking and 3D shape calculation, whose computational complexity is determined by the number of spots in the projection patterns. Processes (2) and (3) should be conducted online for real-time 3D shape measurement, while process (1) is performed offline to determine the initial settings. Fig. 4.1 shows the flowchart of processes (2) and (3).

(1) Generation of blink-spot projection patterns

The Q blink-spot patterns $P_i(X_p, Y_p)$ ($i = 0, \ldots, Q-1$) are projected periodically by

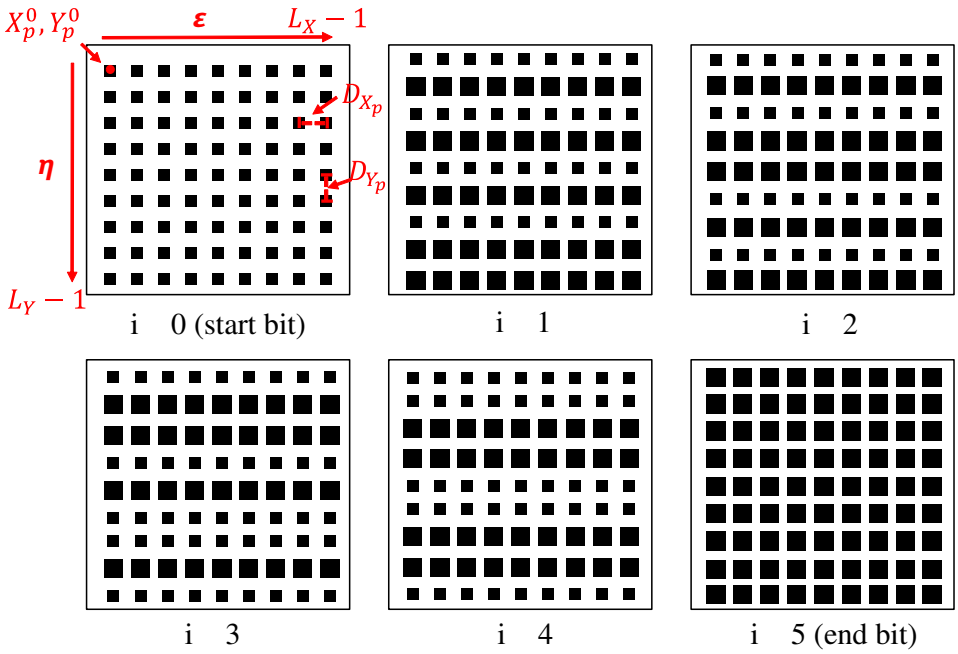


Figure 4.2: Blink-spot pattern sequence ($Q = 6$).

a projector to localize and identify $L_X \times L_Y$ rectangular spots on the illuminated scene:

$$P_i(X_p,Y_p) = \begin{cases} 1 & (|X_p - X_p^{\xi}| \le w_i^{\eta}, |Y_p - Y_p^{\eta}| \le w_i^{\eta}) \\ 0 & (\text{otherwise}) \end{cases}, \tag{4.1}$$

where (X_p^{ξ}, Y_p^{η}) is the center position of the (ξ,η)-th spot in the X_pY_p-space ($\xi = 0,\ldots,L_X - 1; \eta = 0,\ldots,L_Y - 1$):

$$(X_p^{\xi}, Y_p^{\eta}) = (D_{X_p}\xi + X_p^0, D_{Y_p}\eta + Y_p^0), \tag{4.2}$$

where (D_{X_p}, D_{Y_p}) is the interval between two adjacent spots. (X_p^0, Y_p^0) is the center position of the upper-left spot ($(\xi,\eta) = (0,0)$) on the projection pattern. The parameter w_i^{η} that indicates the size of the (ξ,η)-th spot of $P_i(X_p, Y_p)$ is determined as follows:

$$w_i^{\eta} = \begin{cases} w_S & (i = 0) \\ w_L & (i = Q-1) \\ (w_L - w_S)b_i^{\eta} + w_S & (\text{otherwise}) \end{cases}, \tag{4.3}$$

where $2w_S + 1$ and $2w_L + 1$ are the edge lengths of the (ξ,η)-th spot when it is OFF ($b_i^{\eta} = 0$) and ON ($b_i^{\eta} = 1$), respectively. The projection pattern when $i = 0$ is regarded as a start-bit image, where all of the spots to be projected are small. The projection image when $i = Q - 1$ is regarded as an end-bit image, where all of the spots to be projected are large.

Assuming that spots on different column are separable even when they are projected on the camera image plane, the blink state of the (ξ,η)-th spot is provided with the row number η, which is independent of the column number ξ. The blink state b_i^{η} ($\eta = 0,\ldots,L_Y - 1$) of the (ξ,η)-th spot is determined as follows:

$$b_i^{\eta} = \begin{cases} {}^g b_i^{\eta} & (\eta = \text{even}) \\ \overline{{}^g b_i^{\eta}} & (\eta = \text{odd}) \end{cases}. \tag{4.4}$$

Here, ${}^g b_i^{\eta}$ ($i = 1,2,\ldots,Q-2$) is the bit of the ID number for the (ξ,η)-th spot in the column ξ in Gray code notation, where two adjacent numbers differ by 1 bit. Each $(Q-2)$-bit ad-

jacent numbers in the odd-inverted blink state are expressed in Eq. (4.4) as $(b_1^{\eta}, \ldots, b_{Q-2}^{\eta})$ and $(b_1^{\eta+1}, \ldots, b_{Q-2}^{\eta+1})$. These differ by $Q-3$ bits to prevent the misidentification of two adjacent spots during the space code decoding process.

To project $L_X \times L_Y$ blink spots, $Q = \lceil \log_2 L_Y \rceil + 2$ is the minimum number of blink-spot patterns that can determine the row ID numbers for all of the spots, including the start-bit and end-bit images. $\lceil x \rceil$ denotes the least integer greater than or equal to x. Fig. 4.2 illustrates a 9×9 blink-spot pattern sequence generated with our method ($Q = 6$).

(2) Moment feature extraction of multiple spots

a) Image acquisition

When a projector projects the blink-spot patterns of $P_0(X_p, Y_p)$, $P_1(X_p, Y_p)$, $\ldots$, $P_{Q-1}(X_p, Y_p)$ at an interval of τ, the image sensor captures a gray-level image of $M \times N$ pixels at the time $t = k\tau$ as follows:

$$I(X_c, Y_c, k) = \mathrm{Proj}(P_{k \bmod Q}(X_p, Y_p)), \tag{4.5}$$

where k expresses the frame number of the captured image.

b) Binarization

The binary image is obtained by thresholding the captured gray-level image $I(X_c, Y_c, k)$ with the threshold θ_B as follows:

$$B(x, y, k) = \begin{cases} 1 & (I(X_c, Y_c, k) > \theta_B) \\ 0 & (\text{otherwise}) \end{cases}. \tag{4.6}$$

c) Calculation of cell-based moment features

To reduce the number of scanned pixels for labeling without degrading the accuracy of the spatial resolution, the binary image $B(X_c, Y_c, k)$ is divided into $M'N'$ cells Γ_{ab} ($a = 0, \ldots, M'-1; b = 0, \ldots, N'-1$) of $m \times n$ pixels, where $M = mM'$ and $N = nN'$. The zeroth- and first-order moment features are calculated for each cell as follows:

$$M_{pq}(\Gamma_{ab}, k) = \sum_{X_c=am}^{a(m+1)-1} \sum_{Y_c=bn}^{b(n+1)-1} X_c^p Y_c^q \cdot B(X_c, Y_c, k), \tag{4.7}$$

where $(p, q) = (0, 0), (1, 0)$ and $(0, 1)$. Γ_{ab} is expressed as follows:

$$\Gamma_{ab} = \{(X_c, Y_c) \,|\, (am + p, bn + q)\,, 0 \leq p < m, 0 \leq q < n\}. \tag{4.8}$$

d) Connected component labeling with feature calculation

To obtain the moment features of multiple spots in an image simultaneously, the connected component labeling process is accelerated with the cell-based labeling algorithm [35]. This algorithm can reduce the computational complexity and memory consumption on the order of $O(M'N')$ by using the cell-based moment features for $M'N'$ cells: $1/mn$ of the pixel-level complexity on the order of $O(MN)$. The label-domain moment features $M_{pq}(O_l(k))$ are accumulated for the labelled regions $O_l(k)$ $(l = 0, \ldots, L-1)$ in $B(X_c, Y_c, k)$ concurrently with the scanning of the flag map $F(\Gamma_{ab}(k))$ of $M'N'$ cells. $F(\Gamma_{ab}(k))$ is defined by checking $M_{00}(\Gamma_{ab}(k))$ with a threshold θ_L as follows:

$$F(\Gamma_{ab}(k)) = \begin{cases} 1 & (M_{00}(\Gamma_{ab}(k)) \geq \theta_L) \\ 0 & (\text{otherwise}) \end{cases}. \tag{4.9}$$

where $M_{pq}(O_l(k))$ are defined as follows:

$$M_{pq}(O_l(k)) = \sum_{(X_c, Y_c) \in O_l(k)} X_c^p \cdot Y_c^q \cdot B(X_c, Y_c, k). \tag{4.10}$$

The detailed processes of the cell-based labeling algorithm and its evaluation with regard to accuracy, speed, and memory consumption are given in [35].

e) Calculation of spot size and location

The size $S(O_l(k))$ and position $(X(O_l(k)), Y(O_l(k)))$ of the labelled region $O_l(k)$ at the frame k are obtained as those of spots in an image using its zeroth- and first-order moment features:

$$S(O_l(k)) = M_{00}(O_l(k)), \tag{4.11}$$

$$(X(O_l(k)), Y(O_l(k))) = \left(\frac{M_{10}(O_l(k))}{M_{00}(O_l(k))}, \frac{M_{01}(O_l(k))}{M_{00}(O_l(k))} \right) \tag{4.12}$$

(3) Multi-spot tracking and 3D measurement

a) Frame-to-frame correspondence

Based on the geometric relationship of the camera-projector system, $L_X \times L_Y$ spots are projected onto L_X columns on the X_cY_c-plane; these are located at regular intervals parallel to the Y_c-axis independent of the 3D shape of the object to be measured. We categorize the labelled regions $O_l(k)$ into the following L_X sets $\mathcal{A}_\xi(k)$ ($\xi = 0, \ldots, L_X - 1$):

$$\mathcal{A}_\xi(k) = \{O_l(k) \mid |X(O_l(k)) - X_c^\xi| \leq \theta_D\}, \tag{4.13}$$

where $X_c^\xi = D_c\xi + X_c^0$ is the X_c-coordinate of the ξ-th column. D_c is the interval of the L_X columns on the X_cY_c-plane. X_c^0 is the X_c-coordinate value of the zeroth columns on the X_cY_c-plane and corresponds to the spots at $\xi = 0$ in the projection pattern. θ_D is the acceptable deviation in the X_c-direction. The labelled regions $O_l(k)$ that belong to $\mathcal{A}_\xi(k)$ are sorted as $O'_{\xi j}(k)$ ($j = 0, \ldots, L_Y - 1$) with their Y_c-coordinates as follows:

$$O'_{\xi j}(k) = \{O_l(k) \in \mathcal{A}_\xi(k) \mid Y(O'_{\xi j}(k)) > Y(O'_{\xi j'}(k)) \quad (^\forall j' < j - 1)\}. \tag{4.14}$$

The computational complexity of the multi-spot correspondence between frames is reduced by narrowing the search range of the limited number of extracted spots, those belong to $\mathcal{A}_\xi(k - i)$ at the previous frame $k - i$ ($i = 1, \ldots, Q - 1$), as follows:

$$T_{\xi j}(k-i) = \begin{cases} G_{\xi j}(k-i) & (|X(O'_{\xi j}(k) - X(G_{\xi j}(k-i))| < \theta_G) \\ \phi & (\text{otherwise}) \end{cases}, \tag{4.15}$$

$$G_{\xi j}(k-i) = \underset{O'_{\xi j'}(k-i)}{\arg\min} |X(O'_{\xi j}(k)) - X(O'_{\xi j'}(k-i))|, \tag{4.16}$$

where $T_{\xi j}(k - i)$ ($\in \mathcal{A}_\xi(k - i)$, $j = 0, \ldots, L_Y - 1$) indicates the nearest spot region at the frame $k - i$ corresponding to $O'_{\xi j}(k)$ at the frame k. The threshold θ_G determines the distant pairs of the extracted spots to be discarded. ϕ indicates the spot failed to establish

correspondence.

b) Frame order detection

In Eq. (4.3), all of the spots are small at the start-bit frame, and all of the spots are large at the end-bit frame. The frame order $\tau(k)$ of frame k in one projection sequence can be detected by checking the ratio of the total area for all of the spots at the frame k to that at the frame $k-1$ with the threshold θ_S as follows:

$$\tau(k) = \begin{cases} 0 & \left(\dfrac{\sum_l S(O_l(k))}{\sum_l S(O_l(k-1))} < \theta_S \right), \\ \tau(k-1)+1 & \text{(otherwise)} \end{cases} \tag{4.17}$$

c) Blink state determination

The blink state of the spot region $O'_{\xi j}(k)$ is determined by comparing the spot size with that at the previous start-bit frame $k_S(k) = k - \tau(k)$ and end-bit frame $k_E(k) = k - \tau(k) - 1$ as follows:

$$g(O'_{\xi j}(k)) = \begin{cases} 0 & (|S(O'_{\xi j}(k)) - S(T_{\xi j}(k_S(k))| < \theta_I) \\ 1 & (|S(O'_{\xi j}(k)) - S(T_{\xi j}(k_E(k)))| < \theta_I), \\ \phi & \text{(otherwise)} \end{cases} \tag{4.18}$$

where θ_I is a threshold to determine the blink state. Corresponding to $O'_{\xi j}(k)$, $T_{\xi j}(k_S(k))$ and $T_{\xi j}(k_E(k))$ indicate the tracked spot regions at the previous start- and end-bit frames, respectively. ϕ indicates ambiguous binarization state.

d) Spot ID number decoding

When no or one blink state of $O'_{\xi j}(k)$ is empty, the row ID number of $O'_{\xi j}(k)$ can be decoded by accumulating Q blink states as per Eq. (4.4):

$$q(O'_{\xi j}(k)) = \sum_{k'=0}^{Q-1} 2^{Q-2-\tau(k-k')} \cdot L\left(g'(T_{\xi j}(k-k'))\right), \tag{4.19}$$

where $L(x)$ is a function to convert odd-inverted Gray code to binary code. $g'(T_{\xi j}(k))$ is

defined as follows to exclude the blink states at the start-bit and end-bit frames:

$$g'(O'_{\xi j}(k)) = \begin{cases} 0 & (\tau(k-k') = 0 \text{ or } Q-1) \\ g(O'_{\xi j}(k)) & (\text{otherwise}) \end{cases}. \tag{4.20}$$

To discard incorrectly decoded spots due to ambiguous binarization, the decoded ID numbers $q(O'_{\xi j}(k))$ are inspected by considering their reasonable ID numbering in space domain and their consistencies with time domain.

In space domain, the decoded ID numbers $q(O'_{\xi j}(k))$ are conformed by checking whether they are sorted in ascending order with j corresponding to their X_c-coordinates:

$$q'(O'_{\xi j}(k)) = \begin{cases} q(O'_{\xi j}(k)) & (q'(O'_{\xi j'}(k)) < q(O'_{\xi j}(k)), \\ & \quad (\forall j' < j-1)). \\ \phi & (\text{otherwise}) \end{cases} \tag{4.21}$$

To recover the column ID numbers of unidentified spot regions, $q'(O'_{\xi j}(k))=\phi$, they are interpolated using other identified spot regions in their neighbourhood in the Y_c direction:

$$q''(O'_{\xi j}(k)) = \begin{cases} q'(O'_{\xi j}(k)) & (q'(O'_{\xi j}(k)) \neq \phi) \\ q'(O'_{\xi j-1}(k)) + 1 & (q'(O'_{\xi j}(k)) = \phi, \\ & \quad q'(O'_{\xi j-1}(k)) \neq \phi) \\ q'(O'_{\xi j+1}(k)) - 1 & (q'(O'_{\xi j}(k)) = q'(O'_{\xi j-1}(k)) \\ & \quad = \phi, q'(O'_{\xi j+1}(k)) \neq \phi) \\ \phi & (\text{otherwise}) \end{cases}. \tag{4.22}$$

For ID consistency over time, the determining ID number $q^*(O'_{\xi j}(k))$ is computed as the following mode value that appears most often in the decoded ID numbers at the previous K_p frames, the current frame, and the following K_f frames:

$$q^*(O'_{\xi j}(k)) = \text{mode}(q''(O'_{\xi j}(k-K_p)), \ldots, q''(O'_{\xi j}(k+K_f)), \tag{4.23}$$

where $\text{mode}(x_1, \ldots, x_n)$ is a function to determine the mode value in a set of $(x_1, \cdots, x_n)$.

Consequently, the spot region $O'_{\xi j}(k)$ on the camera image plane are identified as the $(\xi, q^*(O'_{\xi j}(k)))$-th blink spots corresponding to which is located at $(X_p^{\xi}, Y_p^{q^*(O'_{\xi j}(k))})$ in the projection pattern as defined in Eq. (4.2).

e) Triangulation

Using the geometric relationship between the spot region projected at $(X(O'_{\xi j}(k)), Y(O'_{\xi j}(k))$ on the camera image plane and its corresponding blink spot at $(X_p^{\xi}, Y_p^{q^*(O'_{\xi j}(k))})$ on the projector image plane, the 3D position (x, y, z) of the $(\xi, q^*(O'_{\xi j}(k)))$-th spot where $q^*(O'_{\xi j}(k)) \neq \phi$ is obtained with triangulation.

4.2.2 System Implementation

We implemented our blink-spot projection method on an HFR camera-projector system for dynamic 3D scene measurement. Our system includes a 3D module that consists of an HFR camera head and HFR projector, a high-speed FPGA-based image processing board (IDP Express board), and a personal computer (PC).

The 3D module consists of a monochrome camera head (Photron Ltd., Japan) and DLP LightCrafter projector (Texas Instruments Inc., US); its dimensions and weight are $140 \times 96 \times 100$ mm^3 and 960 g, respectively. The camera head has a CMOS image sensor with a resolution of 512×512 pixels and can capture 8-bit gray-level images of

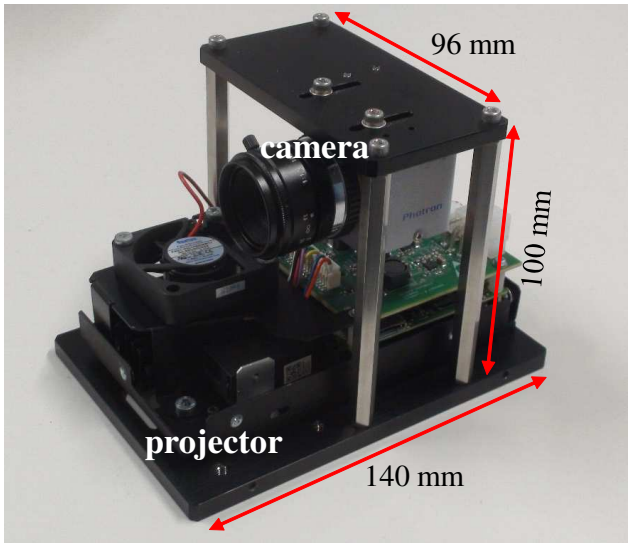


Figure 4.3: Overview of HFR 3D module.

512 × 512 pixels at 2000 fps or more. The physical dimensions of its sensor and the pixel size are 5.12 × 5.12 mm^2 and 10 × 10 μm^2, respectively. DLP LightCrafter is a development kit for high-speed projection using a DMD device; it can project 608 × 684 binary patterns at 2000 fps or more in synchronization with an external device. The focal lengths of the camera and projector are f_c = 17.4 mm and f_p = 10.4 mm, respectively. When the distance from the 3D module to the level surface is set at z = 340 mm, the projector forms a 608 × 684 projection pattern in a 188 × 106 mm^2 rectangular area on the level surface. On the level surface, the area of the captured image is a 106 × 106 mm^2 square. At the center of this region on the level surface, the measurable range of the depth measurement is z = 0∼80.0 mm, and the depth information can be obtained with an accuracy of 1.0 mm or less. The accuracy corresponds to the digitized error on the camera image plane. Fig. 4.3 shows an overview of the 3D module.

The IDP Express board is an FPGA-based image processing board for the high-speed vision platform IDP Express [60]. It consists of two camera inputs for camera heads and a user-specific FPGA (Xilinx XC3S5000) for hardware implementation of the image processing algorithms. The IDP Express board has several trigger I/Os for synchronization with external devices. We used a PC with an ASUSTeK P6T7 mainboard, Intel Core i7 960 3.20 GHz CPU, 3 GB of memory, Windows XP SP3 (32-bit) OS, and two 16-lane PCI-e 2.0 buses.

For the moment feature extraction of multiple spots in parallel, which has a pixel-level computational complexity of $O(MN)$, an extended multi-object feature extraction module [35] was hardware-implemented on the user-specific FPGA on the IDP Express board. Fig. 4.4 illustrates the schematic data flow of the implemented module. An 8-bit 512 × 512 pixel gray-level image (M=N=512) is scanned in parallel in a unit of four pixels from the upper-left to the lower-right using X and Y address signals with a 151.2 MHz clock, and the 512 × 512 pixel image is processed by being divided into 16,384 cells (M'=N'=128) of 4 × 4 pixels (m=n=4). The module consists of a binarization submodule, cell-based feature extraction submodule, connected component labeling submodule, and data selector for FIFO output.

The binarization submodule converts a 512 × 512 pixel gray-level input image

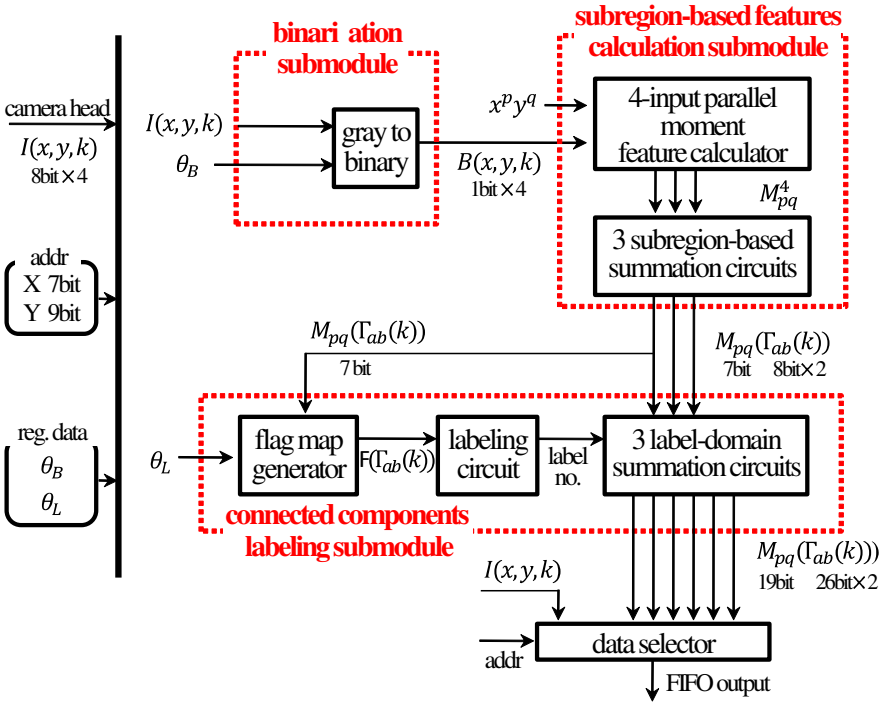


Figure 4.4: Schematic data flow.

$I(X_c, Y_c, k)$ into a binary image $B(X_c, Y_c, k)$ with the threshold θ_B in a unit of four pixels using X and Y address signals with a 151.2 MHz clock.

The cell-based feature extraction submodule calculates the zeroth- and first-order moment features $M_{pq}(\Gamma_{ab}(k))$ ((p, q) =(0, 0), (1, 0), and (0, 1)), for 16,384 cells of 4×4 pixels using a four-input parallel moment feature calculator and three cell-based summation circuits. The four-input parallel moment feature calculator adds the four-pixel data of $B(X_c, Y_c, k)$ in parallel as the zeroth- and first-order moment features for four pixels M^4_{pq} ((p, q) =(0, 0), (1, 0), and (0, 1)) by weighting $X_c^p Y_c^q$ in synchronization with a 151.2 MHz clock. The cell-based moment summation circuit sequentially adds the four-pixel moment features M^4_{pq} for 16,384 cells. The zeroth- and first-order cell-based moment features $M_{pq}(\Gamma_{ab})$ are output immediately after all of the pixels in a cell are scanned.

The connected component labeling submodule calculates the zeroth- and first-order moment features $M_{pq}(O_l(k))$ ((p, q) =(0, 0), (1, 0), and (0, 1)) for 2048 (= L) labeled regions $O_l(k)$ using a flag-map generator, a labeling circuit, and three label-domain summation circuits. The flag-map generator determines active cells among the 16384 cells

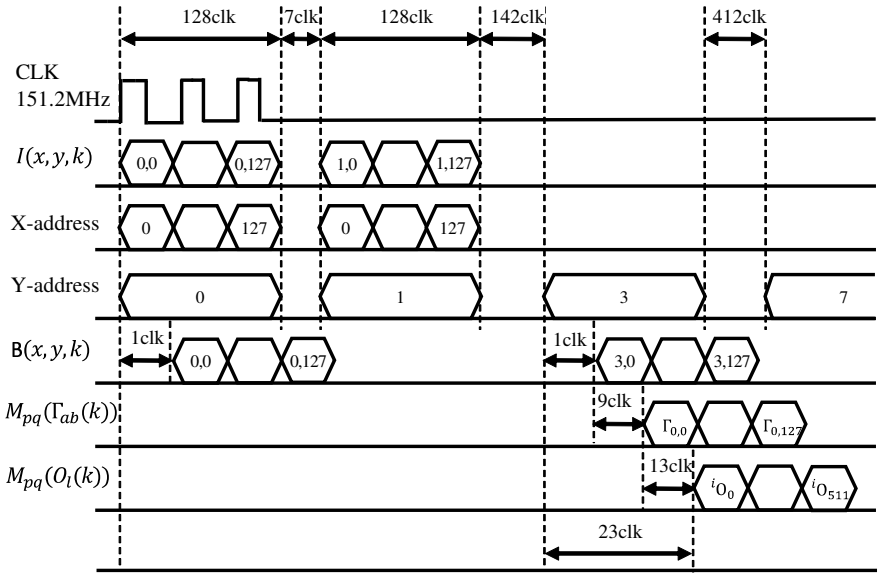


Figure 4.5: Timing chart of control signals.

Table 4.1: Resource consumption of user-specific FPGA.

Device type	Xilinx XC3S5000-4FGG900
Slice	12,608/33,280 (37%)
Slice Flip-Flop	12,926/66,560 (19%)
4-input LUT	16,667/66,560 (25%)
Bounded IOB	229/633 (36%)
Block RAM	74/104 (71%)
MULT 18×18	4/104 (3%)
DCLK	2/8 (25%)

when the cell-based moment features $M_{00}(\Gamma_{ab})$ are greater than the threshold θ_L. The labeling circuit labels all of the active cells as four-connected cells, and the label-domain summation circuit incrementally adds the cell-based moment features of an active cell to the memories of the same label number after the cell-based moment features for a cell In the submodule, the calculated zeroth- and first-order label-domain moment features are output in parallel after all the pixels in an image are raster-scanned.

The data selector is designed to select an FIFO output for an external PC with X and Y address signals and a 151.2 MHz clock from the input image or the three types of label-domain moment features of the 2048 labeled regions.

Fig. 4.5 shows the timing chart for the control signals of the multi-object feature extraction module. The time delay for calculating the 2048 label-domain zeroth- and

first-order moment features is 23 clocks (one clock = 6.6 ns) after all of the pixels in an input image are raster-scanned with a 151.2 MHz clock. In contrast, the time delay to output them to the external PC is one frame (= 0.5 ms): 1 clock for the binarization submodule, 9 clocks for the cell-based moment feature calculation submodule, and 22 clocks for the connected component labeling submodule. Table 4.1 summarizes the resource consumption of the user-specific FPGA on the IDP Express board when the multi-object feature extraction module is implemented.

4.2.3 Execution Time

For our blink-spot projection method, the moment feature extraction of multiple spots requires pixel-level computation with a complexity of $O(MN)$. This is accelerated by implementing the 4×4 cell-based multi-object feature extraction module in the hardware. Meanwhile, the (3) multi-spot tracking and 3D measurement requiring object-level computation with a computational complexity of $O(L)$ is implemented on the PC in the software.

Fig. 4.6 shows the total execution time (HW&SW) when $L_X \times L_Y$ blink spots (L_X = L_Y =4, 8, 12 and 16) were extracted from 512×512 pixel images on our HFR camera-projector system with the hardware-implemented multi-object feature extraction and the total execution time (SW) when all of the processes of our method were implemented by software on the PC of the camera-projector system. The parameters for ID consistency with time are $K_f = K_p$ =18. The execution time of the software-implemented process (2) was 2.8 ms or more in all of the cases. The execution time of the hardware-implemented process (2) was less than 2.0 ms with parallel processing accelerated by hardware logic. The execution time of the software-implemented process (3) increased with the number of blink spots; however, the total execution time was less than 2.0 ms even when 16×16 blink spots were extracted from a 512×512 pixel image because the number of blink spots was much less than the number of pixels.

Table 4.2 presents the total execution time and its breakdown when 16×16 blink spots (L_X=L_Y= 16) were extracted. The total execution time was 1.6 ms when 16×16

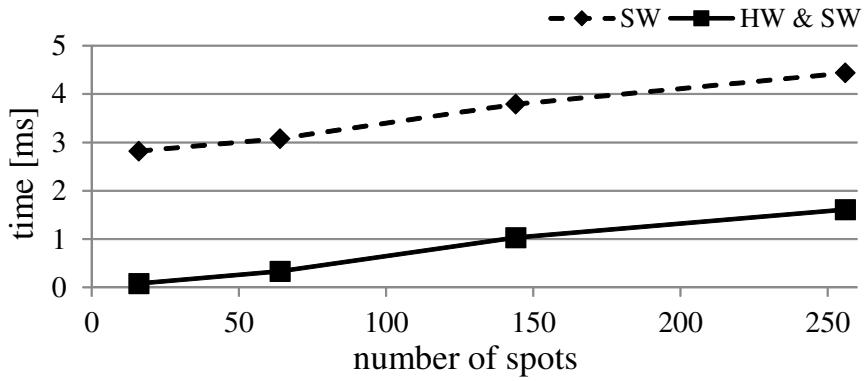


Figure 4.6: Relationship between execution times and number of spots.

Table 4.2: Execution time (16 × 16 spots).

	time[ms]	
	SW	HW&SW
(2) Moment feature extraction of multiple spots	2.812	0.034
2-a) Image acquisition	0.643	0.034
2-b) Binarization	0.602	
2-c) Cell-based moment feature calculation	1.299	
2-d) Connected component labeling	0.268	
2-e) Spot size and location calculation	0.001	
(3) Multi-spot tracking and 3D measurement	1.590	
3-a) Frame-to-frame correspondence	0.091	
3-b) Start-bit frame detection	0.001	
3-c) Blink-state determination	0.542	
3-d) Spot ID number decoding	0.875	
3-e) Triangulation	0.081	
Total execution time	4.439	1.625

blink spots were processed with hardware logic; their 3D measurement could be conducted in real time at 500 fps. We confirmed the real-time 3D measurement using a 16 × 16 blink-spot projection for 512 × 512 pixel images at 500 fps with stable synchronization between the high-speed vision and projector.

4.3 Experiments

4.3.1 Experimental Setting

We mounted the 3D module on the end of a six degrees-of-freedom (DOF) robot manipulator RV-4F (Mitsubishi Electric Co., Japan) and measured 3D scenes at different heights. Fig. 4.7 shows an overview of the 3D module mounted on the robot manipulator.

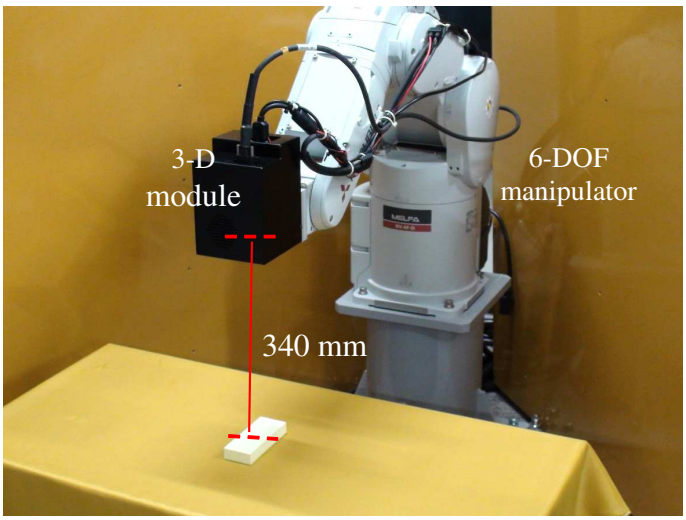
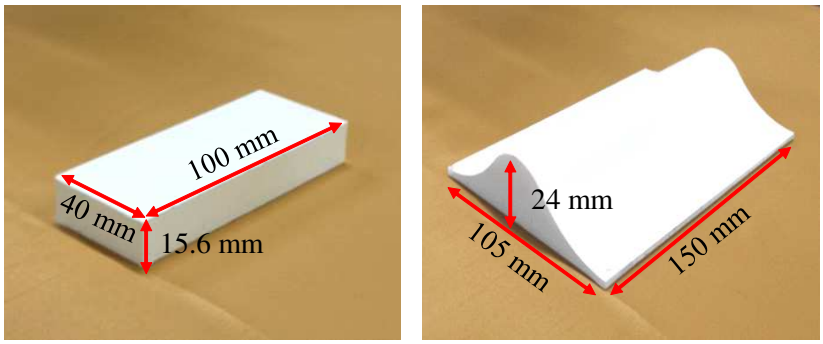


Figure 4.7: 3D module mounted on robot manipulator.



(a) cuboid (b) curved-surface object

Figure 4.8: Objects to be measured.

The experimental parameters used for the 16×16 blink-dot projection were as follows:

– threshold to determine active cells: $\theta_L = 1$.

– sizes of large and small spots: $w_L = 4$, $w_S = 2$.

– interval between two adjacent spots: $D_{X_p} = 7$ and $D_{Y_p} = 26$.

– parameter to determine start-bit frame: $\theta_S = 0.7$.

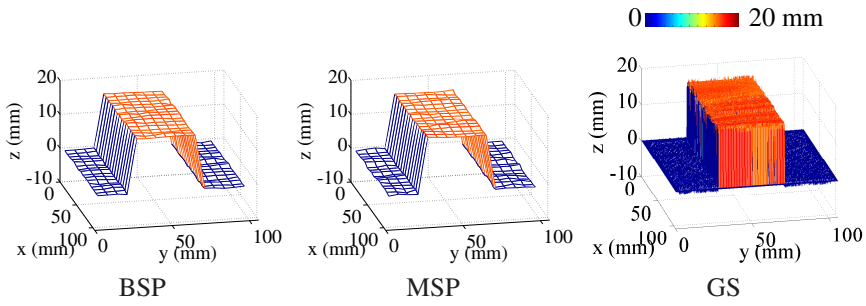
– parameters for frame-to-frame correspondence:$\theta_D = \theta_G = 15$.

– binarization threshold: $\theta_B = 50$.

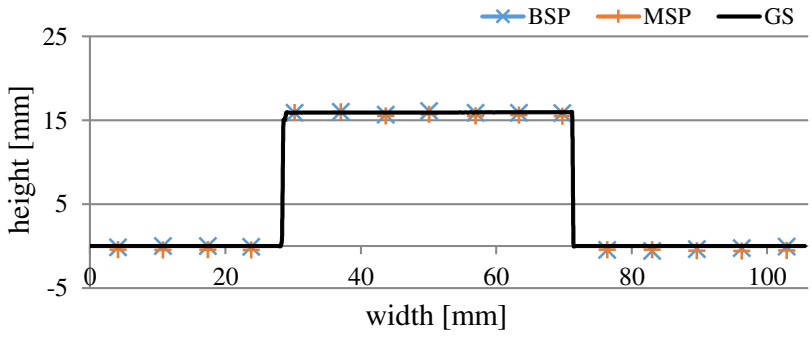
– parameter to determine blink state: $\theta_I = 12$.

– parameters for ID consistency in time: $K_f = K_p = 18$.

In the experiments, we moved the 3D module horizontally 340 mm above the level



(a) 3D shapes



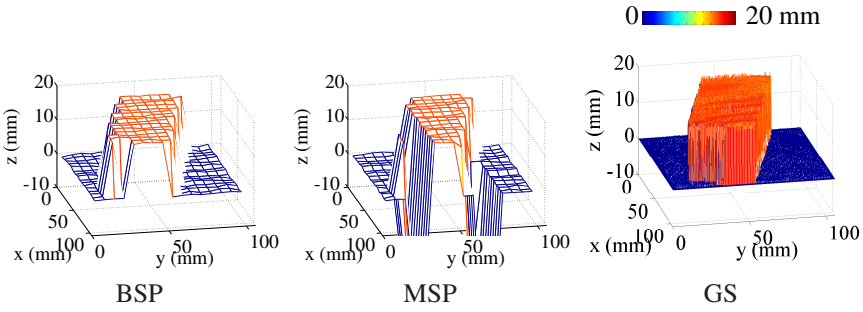
(b) 3D intersections

Figure 4.9: 3D profiles of cuboid measured when still 3D module could observe all spots .

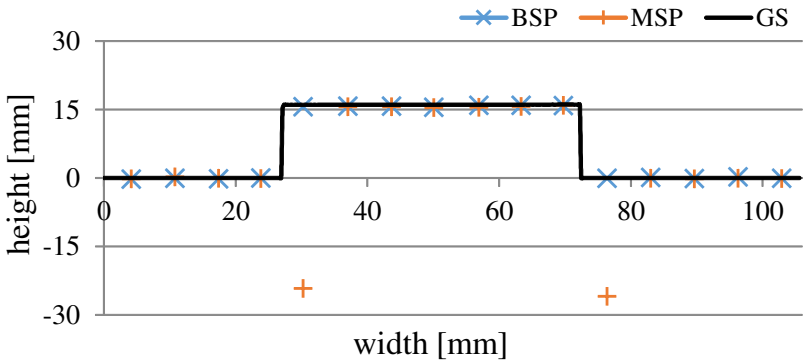
plane, where we placed several objects to be measured. The 3D shapes of 16×16 points measured with our blink-spot projection (BSP) method were compared with those of the 16×16 points measured with the multi-spot projection (MSP) method [44] and 512×512 points measured with the 10-bit Gray-code structured light (GS) method [70]. The 3D shapes measured with the BSP method were calculated in real time at 500 fps. The 3D shapes measured with the MSP and GS methods were calculated offline after the same HFR camera-projector system captured input images of 512×512 pixels at 500 fps. The 3D module was moved in the same manner as for the BSP method. The MSP parameters were the same as the BSP parameters, except that the spot size for the projection was fixed to $w_L = w_S = 2$ without blinking. For the GS method, the minimum stripe width of the Gray-code projection patterns was set to 1 pixel.

4.3.2 Scenes from 3D Module at Fixed Location

First, we verified the accuracy for the 3D shape of a 15.6 mm tall cuboid on the level plane when the 3D module was located 340 mm above the level surface without any



(a) 3D shapes



(b) 3D intersections

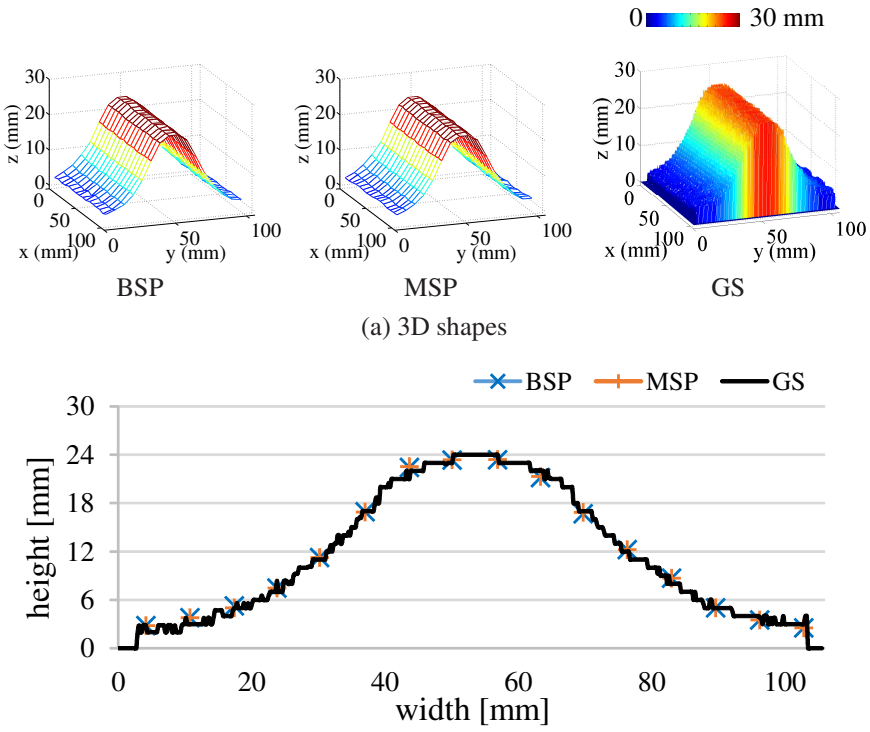
Figure 4.10: 3D profiles of cuboid measured when still 3D module could not observe several spots

.

motion. Fig. 4.8(a) shows the cuboid to be measured; the width, length, and height of the cuboid were 40.0, 100.0, and 15.6 mm, respectively.

Fig. 4.9 shows the (a) 3D shapes and (b) 3D profiles intersected on the horizontal centreline when the cuboid was located so that its edges were parallel to the x-, y- and z-axes and all of the spots were observable from the 3D module. The BSP, MSP, and GS methods measured the height of the surface of the cuboid to be 15.9, 15.9, and 16.0 mm, respectively. Compared with the actual height of the cuboid, the maximum error of the BSP method was within 0.5 mm, corresponding to the digitized error in the camera image plane. Those of the MSP and GS methods were within 0.5 and 1.2 mm, respectively.

Fig. 4.10 shows the (a) 3D shapes and (b) 3D intersection profiles when the cuboid was obliquely located on the level surface so that several spots were unobservable from the 3D module. The 3D shapes of the cuboid were accurately measured by the BSP and GS methods. However, the MSP method produced serious errors around the edges of the



(a) 3D shapes

(b) 3D intersections

Figure 4.11: 3D profiles of curved-surface object measured with still 3D module.

cuboid; the measured 3D shape showed local deviations of 20 mm or more from the actual height of the cuboid. This is because several spots were occluded or split into small ghost spots around the large gaps in depth, and we could not guarantee the assumption that all of the spots could be observed from the 3D module for correct spot identification with the MSP method. As a result, misidentified spots produced serious errors in the triangulation for 3D measurement.

Fig. 4.11 shows the (a) 3D shapes and (b) 3D intersection profiles when a curved-surface object was observed from the still 3D module. Fig. 4.8(b) shows the curved-surface object to be measured; the width, length, and height were 105.0, 150.0 and 24.0 mm, respectively. We confirmed that the BSP, MSP, and GS methods showed similar accuracies to the results displayed in Fig. 4.9 when the location and orientation of the object were changed on the level plane; the maximum difference in depth between the BSP and MSP methods was 0.2 mm, and that between the BSP and GS methods was 1.4 mm. This is because all of the spots were always observable according to the smooth curvature

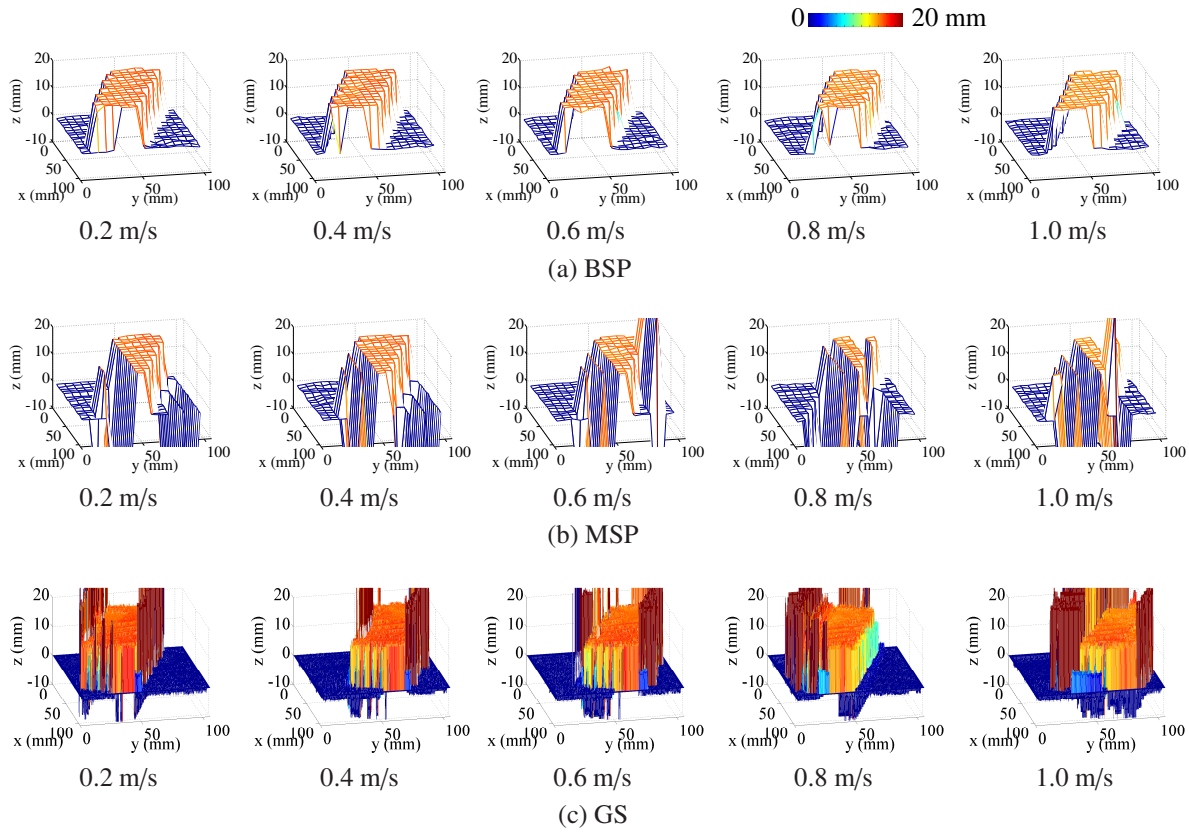


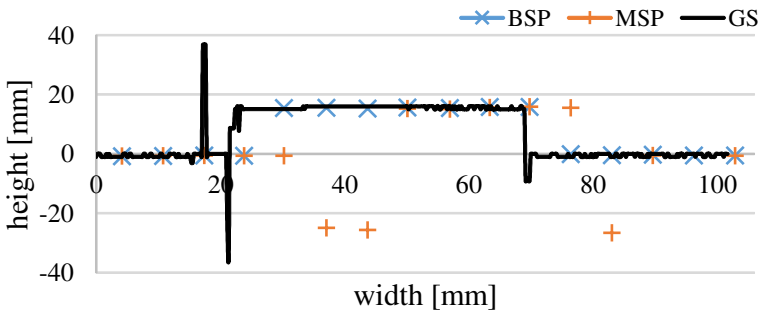
Figure 4.12: 3D shapes of cuboid measured from 3D module moving at constant speeds .

of the object.

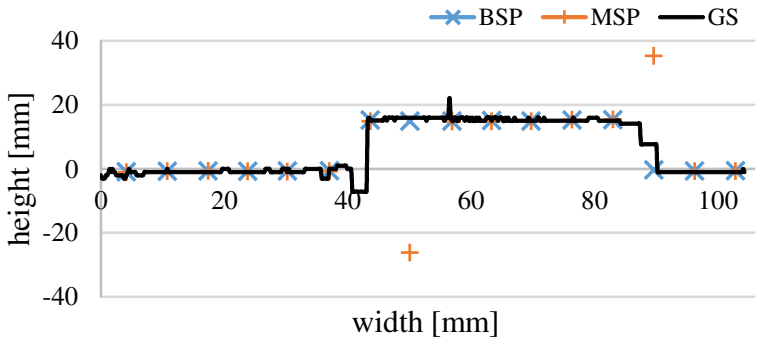
These results indicate that the BSP method showed similar accuracy to the MSP and GS methods when all of the spots were observable with a static 3D module; the BSP and MSP methods used much fewer measurement points than the GS method. When several spots could not be observed around large gaps in depth, misidentified spots often produced serious errors in static 3D measurement with the MSP method, whereas the BSP and GS methods were robust against missing spots because they could identify the spots with time-variant projection patterns.

4.3.3 Scenes from 3D Module Moving at Constant Speeds

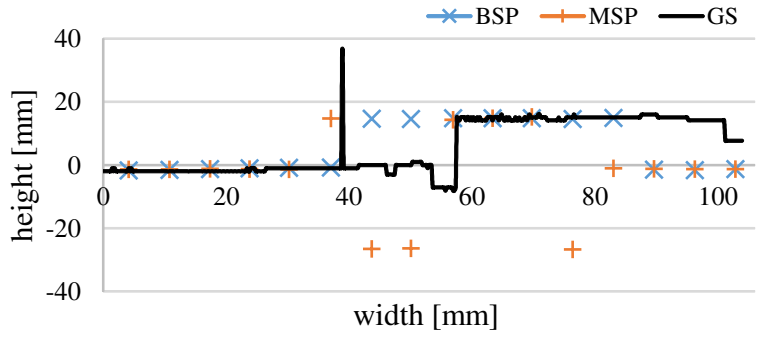
Next, the accuracy of the BSP method when the 3D module was moved horizontally at several constant velocities was compared with those of the MSP and GS methods. We measured the 3D shape of the same 15.6 mm-height cuboid used in the static scene



(a) 0.2 m/s



(b) 0.4 m/s



(c) 0.6 m/s

Figure 4.13: 3D intersections of cuboid measured from 3D module moving at constant speeds.

observation. Fig. 4.12 shows the 3D shapes measured by the BSP, MSP and GS methods when the 3D module passed above the object at speeds of 0.0, 0,2, 0.4, 0.6, 0.8, and 1.0 m/s, where several spots were unobservable from the 3D module. Fig. 4.13 shows the measured 3D profiles on the horizontal centreline of the 3D shapes when the 3D module was moved at 0.2, 0.6, and 1.0 m/s. The 3D shapes measured by the BSP method were similar to the actual height of the cuboid regardless of the moving speed of the 3D module, whereas the 3D shapes measured by the GS method showed larger errors around the edges

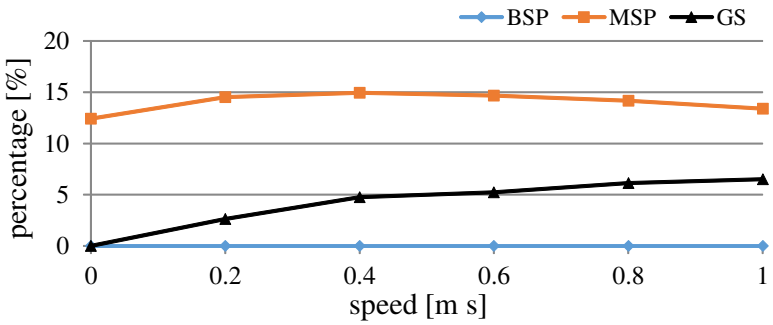


Figure 4.14: Relationship between moving speeds and inaccuracies of 3D measurement

.

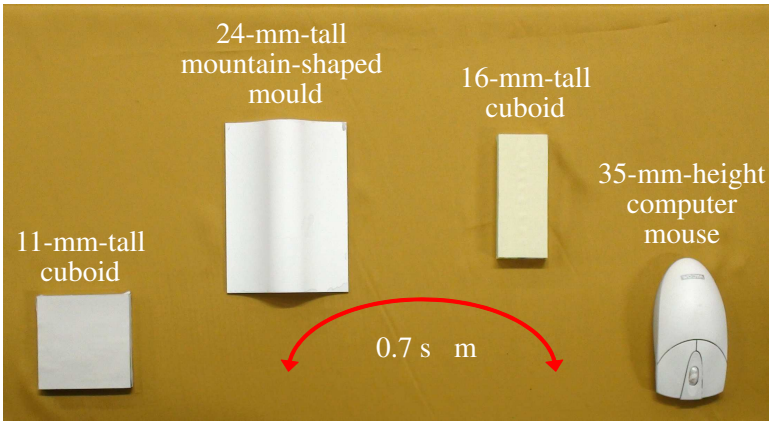
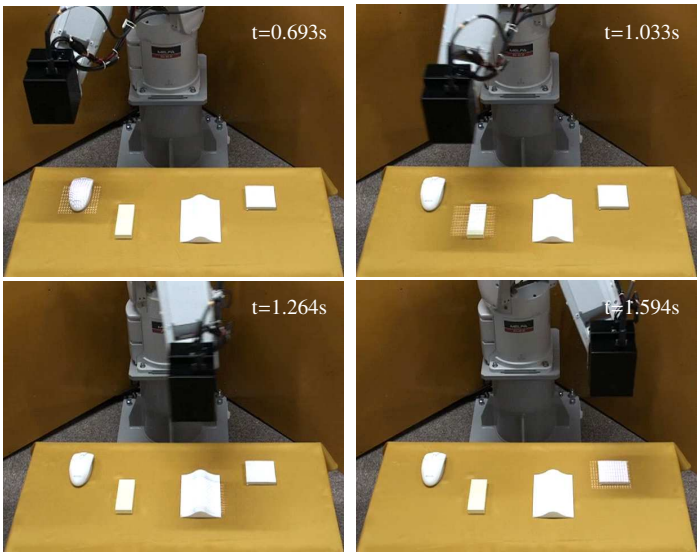


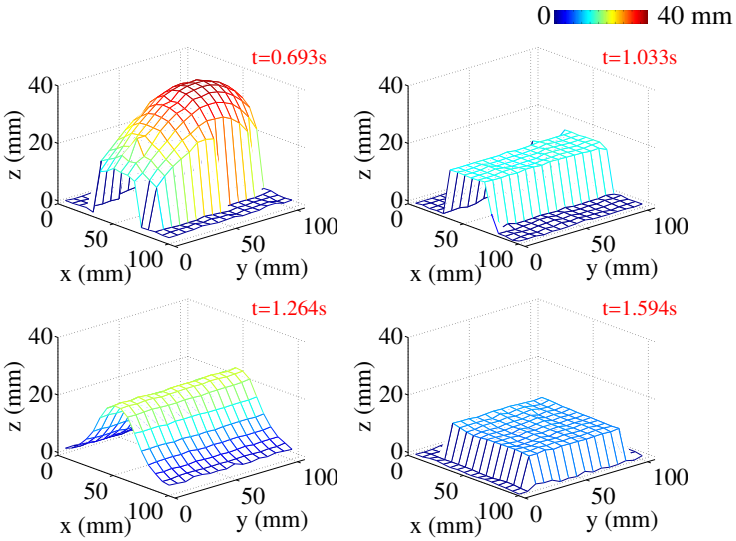
Figure 4.15: 3D objects located on arc orbit.

of the cuboid, where there were large gaps in depth because of the greater moving speed. The MSP method produced serious errors around the edges of cuboid at all speeds; the degree of these errors was independent of their moving speed and similar to the errors of the MSP method when the 3D module was still.

Fig. 4.14 shows the relationship between the moving speed of the 3D module and the percentage of inaccurate measurement points for the BSP method when the 3D module passed above the cuboid compared to those for the MSP and GS methods. The measurement points were regarded as inaccurate when their z coordinates deviated 1.0 mm or more from $z = 0$ or 15.6 mm, which corresponded to the heights of the level plane and the surface of the cuboid, respectively. The BSP method did not measure any points inaccurately at all speeds, whereas the inaccuracy percentage for the GS method increased with the moving speed: 0.0%, 2.6%, 4.7%, 5.2%, 6.1%, and 6.5% at 0.0, 0.2, 0.4, 0.6,



(a) experimental scenes



(b) 3D shapes

Figure 4.16: 3D measurement results when 3D module was moved on arc orbit.

0.8, and 1.0 m/s, respectively. For the MSP method, the inaccuracy percentage did not greatly vary as the moving speed increased; it was more than 10 % at all speeds. These results corresponded to the fact that the errors of the GS method were caused by synchronization errors from decoding space codes using multiple light patterns with different timings, whereas those of the MSP method were caused by misidentified spots in a single frame, as described in the previous subsection. The BSP method markedly reduced the 3D measurement errors for fast-moving scenes with large gaps in depth compared with

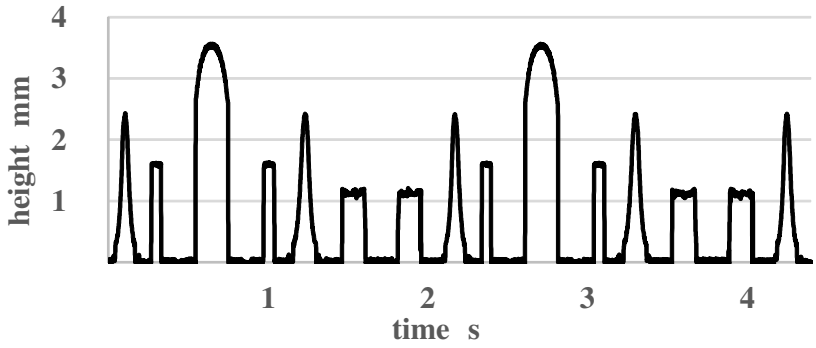


Figure 4.17: z coordinate of $(7, 7)$-th point.

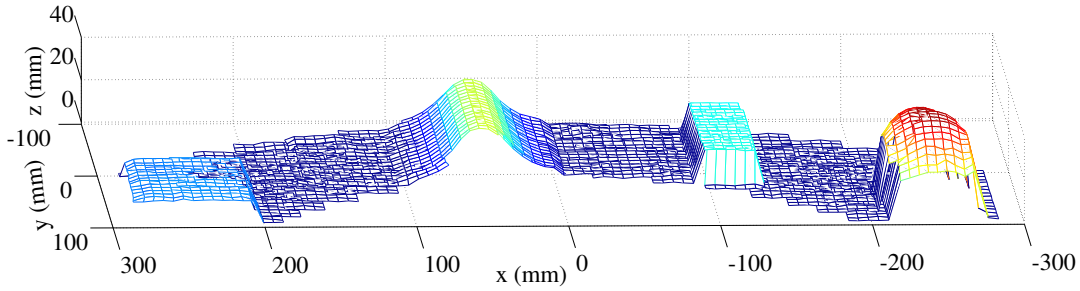


Figure 4.18: Synthesized 3D Scene.

the MSP and GS methods.

4.3.4 Scene from 3D Module Moving on Arc Orbit

Finally, we present the experimental results when the manipulator periodically moved the 3D module left-right-left on an arc orbit with a radius of 34 cm at a cycle time of 2.4 s. The orbit was on a plane 340 mm above the level plane. As shown in Fig. 4.15, 16 and 11 mm tall cuboids, a 24 mm tall mountain-shaped mould, and a 35 mm tall computer mouse were located on the level plane. Fig. 4.16 shows (a) the experimental scenes captured using a standard digital video camera and (b) the 16×16 3D shapes measured by the BSP method that were taken at $t = 0.693$, 1.033, 1.264, and 1.594 s. Fig. 4.17 shows the z-coordinate of the $(7, 7)$-th spot at $t = 0.0$–2.4 s, which corresponded to the cycle time of 2.4 s. The 3D shapes of the different cuboids, mould, and computer mouse were correctly measured with 16×16 measurement points even when the 3D module was quickly moved above the 3D scene during the measurement. In Fig. 4.17, the z-coordinate value changed

to approximately 35, 16, 24, and 11 mm over a cycle time of 2.4 s, which corresponded to the heights and locations of the cuboids and computer mouse.

Fig. 4.18 shows a synthesized 3D image where 40 selected 3D shapes of 16×16 measurement points over a cycle time of 2.4 s were plotted on the arc orbit of the 3D module. These results show that the 3D shapes were correctly obtained for fast-moving scenes without any synchronization or tracking error as they corresponded to the locations of the cuboids, mould, and computer mouse in Fig. 4.15.

4.3.5 Discussions

The proposed blink-spot projection method uses a series spot pattern with two sizes to divide the measured scene into different parts, and a neighbourhood-search algorithm that is accelerated by implementing a cell-based labeling algorithm in hardware logic is proposed to track corresponding spots in previous frames. These features enable our proposed method to measure fast moving objects accurately without synchronization errors. Compared with projection methods that encode purely in the spatial domain with an assumption of local smoothness, such as Microsoft Kinect [88], our method measures the scene with gap-depth accuracy without imposing restrictions on the surface of the scene. Sequential projection techniques that use time-varying light patterns without involving tracking strategies, such as Gray-code projection [70] and phase-shift projection [47], are reliable for static objects but their accuracies decrease sharply for moving objects. In contrast, the accuracy of the blink-spot projection method is stable even when the measured object moves at high speed. Existing space-time projection techniques, such as stripe boundary codes [57], utilize the advantages of both the space domain and the time domain. However, their execution times limit their effectiveness for measurement of high-speed moving objects, whereas the blink-spot projection method can operate at 500 fps. Further, color variation at high frequency constitutes a limitation for many scanning methods, such as the Hamming color code pattern [39]. Our proposed method is robust to this aspect because the decoding procedure for the illuminated scene is based on brightness.

Although we have demonstrated that our proposed system with blink-spot projec-

tion is capable of fast 3D measurement, it has several limitations that have to be discussed and illustrated. The introduction to the generation of projection patterns in Section 4.2.1 (a) shows that the resolution of a light pattern is $L_X \times L_Y$ and the maximum number of spots in the light pattern is 1024, restricted by the cell-based labelling algorithm, which means that the proposed system can not recover a density depth map with a one-time scan. As shown in Fig. 4.16, the resolution of the depth map can be improved when all the depth images have been registered together; however, the registration procedure with its high computational complexity increases its execution time. Another limitation is that our method may not recover scenes with high frequency discontinuous depth. The ID decoding procedure introduced in Section 4.2.1 is based on the implementation of a filter in the spatial domain with the assumption that at least one spot exists in the three adjacent spots with a reasonable ID. When the scene is too complex to follow this assumption, a considerable amount of noises occurs during the measurement. Further, there is no unified method to determine the parameters because they are related to the light pattern, ambient light condition and the reflectance of the measuring scene. Once the texture or light condition of the measured scene changes significantly, performance of parameter tuning is crucial for high accuracy measurement.

4.4 Concluding Remarks

In this paper, we propose a blink-spot projection method for the accurate and robust 3D shape measurement of fast-moving scenes with ID-based frame-to-frame spot correspondence. We implemented our method with 16×16 blink-spot projection on an HFR camera-projector system; the 3D positions of all the spots in a camera view can be simultaneously calculated at 500 fps by processing 512×512 pixel images in real time. The effectiveness of the method was verified through several experiments on moving scenes where the 3D module was mounted on a robot manipulator, and the results were compared with the 3D shapes measured using other methods. Based on these results, we plan to improve our HFR blink-spot-projection system for much denser and more robust 3D shape measurement. We plan to deploy our proposed 3D sensor to various application fields,

such as robot feedback control and fast moving 3D object tracking and recognition, in the future.

Chapter 5

Real-time Projection Mapping Using High-frame-rate Structured Light 3D Vision

5.1 Introduction

Advances in video technology have led to augmented reality (AR) [73] being proposed for realistically overlaying virtual information on real world materials. With the help of such technology, the information about the surrounding of real world is enhanced or becomes interactive, which offers a vivid and fluent interface for the simple understanding of the scene.

Tracking-based AR system [74] is a well-known techniques which is designed for real-time visualization of synthesized images via display devices such as head-mount and handheld displays; and images are not directly displayed on the real environment to make it indeed augmented. Due to such inconsistencies between the real environment and images on computer displays, presenting realistic augmented information to operators in real environments is difficult. Projection mapping [75, 76] is another well-known spatial augmented reality technology for projecting computer-generated light patterns from projectors onto the real environment. It involves complex-shaped or non-monotoned objects and can turn the real environment into a virtual display surface by generating projection light patterns that are matched with the real environment. However, these studies or technologies mainly focus on rendering static scenes, with an assumption that the information of the real environment is obtained previously, but not on enhancing the appearance of dynamic scenes in which the recognition of a time-varying scene and projection mapping

with automatic alignment restrict rendering performance. Recently, various projection mapping systems that involve image sensors or depth sensors in the rendering of dynamic scenes have been developed. The image-sensor-based projection mapping systems [77] track moving objects using image segmentation algorithms without taking into account the geometrical shape of the projection surface, while the real-time three-dimensional (3D) data in the depth-sensor-based projection mapping system [78] benefits automatic scene recognition and computer graphics generation. However, restricted by the working frame rate of the depth sensor, drawbacks such as the acquisition time of depth images and non-pixel-wise space resolution in 3D sensing and projection remain when observing a high-speed or high-frequency moving scene.

Our goal is to create a robust real-time projection mapping system that creates accurate, dense RGB light patterns to render dynamic scenes for easy observation or understanding without imposing restrictions on surface texture, scene complexity, and object motion. To this end, we report on the development of a high-frame-rate (HFR) projection-mapping system that can acquire and process 512×512 depth-images in real time at 500 fps and project computer-generated light patterns onto time-varying 3D scenes at 60 fps. The depth-image processing is accelerated by installing a GPU board to process the 8-bit Gray-code structured light method using infrared (IR) light patterns. The RGB light patterns are interactively generated according to the depth-images so that the patterns are projected onto the 3D scene and enhanced with pixel-wise correspondence. Experimental results for depth-based color mapping, augmented reality spirit level, and a tracking AR wristwatch are presented to verify the efficacy.

5.2 Real-time HFR Projection Mapping System

5.2.1 Real-time HFR Projection Mapping System

Our HFR projection mapping system comprises two projectors (DLP Light Commander 5500, Texas Instruments), an HFR vision platform IDP Express [94], a GPU board (Tesla 1060, NVIDIA), and a personal computer (PC). Fig. 5.1 shows its configuration. The DLP Light Commander 5500 is a development kit for HFR projection based on

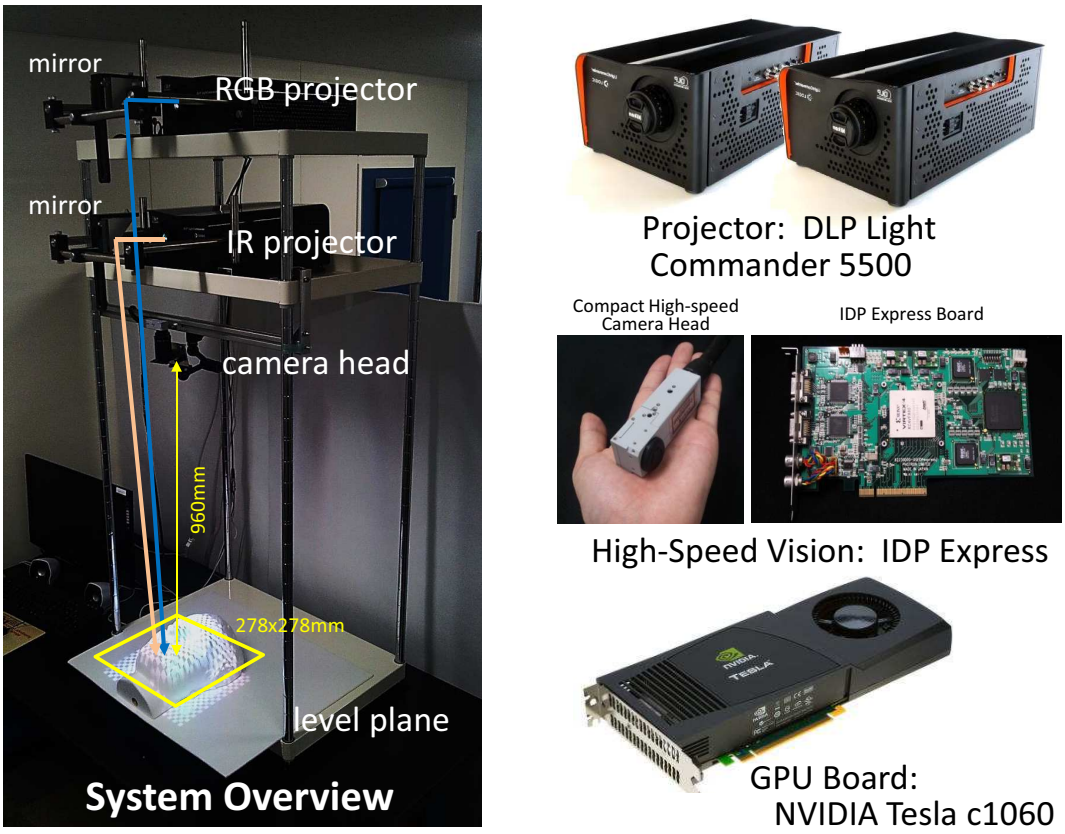


Figure 5.1: System overview.

DMD device technology, which is composed of a high-performance light engine consisting of red (R, 623 nm), green (G, 525 nm), blue (B, 460 nm), and infrared (IR, 850 nm) LEDs, a DLP 0.55 XGA Chipset, and its controller. The two projectors, located on the horizontal shelves, project light patterns onto a common workspace. One projector, hereinafter referred to as the "IR projector," was used for fast structured light 3D measurement to project 1024×768 IR binary light patterns at 1000 fps in synchrony with the HFR vision platform, while the other, hereinafter referred to as the "RGB projector," was used for projection mapping as a standard XGA projector connected to a PC video card, which is visible to the human eye. The IR projector was placed on the lower shelf, while the RGB projector was placed on the upper shelf. Two F-mount 50 mm-focal-length lenses (Ai AF NIKKOR 50 mm f/2.8D lenses, Nikon) were mounted on the projectors. Right-angle aluminum-coated mirrors were used to change the vertical direction of the projections from the projectors.

The IDP Express [94] was designed to implement real-time video processing and recording of 512×512 images at 2000 fps. It comprises a camera head to capture 8-bit gray 512×512 images at 2000 fps, and a dedicated FPGA board (IDP Express board). The IDP Express board has two camera inputs and trigger I/Os for external synchronization. Two 512×512 images and their processed results can be mapped onto the PC memory at 2000 fps via the PCI-e bus. A C-mount 17 mm-focal-length lens with an IR band-pass filter, whose center wavelength and full width-half max are 830 nm and 260 nm, respectively, was mounted on the camera head. RGB light patterns and other daily lightings were reduced in the images, and only IR light patterns captured at 1000 fps for the structured light 3D measurement.

The Tesla 1060 is a computer processor board accelerated by an NVIDIA Tesla T10 GPU. It has a processing performance of 933 Gflops/s using 240 processor cores operating at 1.296 GHz and a bandwidth of 102 GB/s with its inner global memory of 4 GB and fast shared memory of 16 kB. A PC with 16-lane PCI-e 2.0 buses and a processor chipset with DMA were adopted to transfer memory-mapped data between standard memory and the Tesla 1060 via the PCI-e bus at high speed. We used a PC with the following specifications: ASUS P6T7 WS SuperComputer motherboard, Intel Core (TM) i7 3.20 GHz CPU, 3 GB RAM, two 16-lane PCI-e 2.0 buses, and graphic video card (QuadroFX 380, NVIDIA). Structured light 3D measurement was accelerated using parallel-processing software on the Tesla 1060. We used a CUDA IDE provided by NVIDIA to code the algorithms with dedicated API functions for the IDP Express in Windows XP (32 bit), which enabled us to access memory mapped data. A DVI video output of the Quadro FX 380 was connected to the HDMI video input of the RGB projector, and the RGB light patterns generated for projection mapping were projected at dozens of frames per second.

For the xyz-coordinate system in the projection-mapping workspace, the xy-plane was set on a level plane at $z = 0$ mm. The origin was set to the point of intersection of the optical axis of the camera lens and the xy plane. The optical axes of the camera, IR projector, and RGB projector lenses were parallel to the z-axis. The optical center of the camera lens was set at a height of $z = 899$ mm. Via the right angle mirrors, the optical center of the IR projector lens was virtually set at $z = 1204$ mm, and that of the RGB

projector lens was virtually set at $z = 1504$ mm. In order not to disrupt the RGB projection with the right angle mirror for IR projection, the distance between the optical axes of the IR projector and the RGB projector lenses was set to 60 mm. The IR projector projects a 1024×768 IR light pattern in a 272×205 mm region on the level plane. The projected IR light pattern was captured in a 512×512 image using the IDP Express. The image area, a 278×278 mm square on the level plane, and depth information over a 272×205 mm region were observed. The RGB projector projects a 1024×768 RGB light pattern in a 343×257 mm region on the level plane, and covers the depth-measurable 272×205 mm region.

5.2.2 Algorithm

To project computer-generated patterns that respond to time-varying 3D scenes, we implemented (a) structured light 3D measurement and (b) depth-based projection mapping on our HFR camera-projector system. In the structured light 3D measurement based on Inokuchi's method [95], 1024×768 IR binary light patterns coded with an 8-bit Gray-code are projected at 1000 fps; 512×512 images illuminated by light patterns are captured at 1000 fps in synchrony with the IR projector. In the depth-based projection mapping, the depth-images and features captured at a high frame rate in process (a) were used to generate RGB light patterns to project onto the 3D scene, corresponding to the projection-mapping applications used in our study.

5.2.2.1 Structured Light 3D Measurement

The IR projector projects eight pairs of positive and negative light patterns with Gray-code for the consideration of the projection-pattern number and to ensure the required accuracy in a certain order; gray-level 512×512 images are captured corresponding to the projection images. Two images paired with one pair of light patterns, are differentiated owing to the robustness against ambiguities due to the non-uniform brightness at 2 ms intervals; The space-code image is obtained using binary images of the current and previous frames, transformed to the depth of $z = D(X, Y, 2k + 1)$ on the basis of the geometric properties between camera and IR projector. For the projection mapping, the

3D position, orientation and velocity are acquired as follows:

1) Target region extraction

By differentiating the depth-images with a reference depth-image $D_R(X, Y)$ for background reduction, the target region is extracted as follows:

$$Q(X,Y,2k+1) = \begin{cases} 1 & D(X,Y,2k+1) - D_R(X,Y) > \theta_R \\ 0 & \text{(otherwise)} \end{cases}, \tag{5.1}$$

where θ_R is a threshold to extract a target region, and $D_R(X, Y)$ is the 3D background scene provided prior.

2) 3D position, orientation and velocity acquisition

Using the following zeroth-, first-, and second-order moment features M_{pqr} of the 3D point group S_{2k+1} that satisfies $Q(X, Y, 2k+1) = 1$,

$$M_{pqr}(k) = \sum_{(x,y,z)\in S_{2k+1}} x^p y^q z^r \quad (p + q + r \leq 2), \tag{5.2}$$

the 3D position, orientation and velocity of the target object are calculated as follows

$$(\bar{x}(k), \bar{y}(k), \bar{z}(k)) = \left(\frac{M_{100}(k)}{M_{000}(k)}, \frac{M_{010}(k)}{M_{000}(k)}, \frac{M_{001}(k)}{M_{000}(k)} \right), \tag{5.3}$$

$$\begin{aligned} \phi_x(k) &= \frac{1}{2}\tan^{-1} \frac{2(M_{011}(k)M_{000}(k) - M_{010}(k)M_{001}(k))}{(M_{002}(k) - M_{020}(k))M_{000}(k) - M_{001}(k)^2 + M_{010}(k)^2}, \\ \phi_y(k) &= \frac{1}{2}\tan^{-1} \frac{2(M_{101}(k)M_{000}(k) - M_{001}(k)M_{100}(k))}{(M_{200}(k) - M_{002}(k))M_{000}(k) - M_{100}(k)^2 + M_{001}(k)^2}, \\ \phi_z(k) &= \frac{1}{2}\tan^{-1} \frac{2(M_{110(k)}M_{000}(k) - M_{100}(k)M_{010}(k))}{(M_{020}(k) - M_{200}(k))M_{000}(k) - M_{010}(k)^2 + M_{100}(k)^2}, \end{aligned} \tag{5.4}$$

$$
\begin{aligned}
v_x(k) &= \frac{1}{8}\sum_{m=0}^{7}\Big(\bar{x}(k-m) - \bar{x}(k-(m+1))\Big), \\
v_y(k) &= \frac{1}{8}\sum_{m=0}^{7}\Big(\bar{y}(k-m) - \bar{y}(k-(m+1))\Big), \\
v_z(k) &= \frac{1}{8}\sum_{m=0}^{7}\Big(\bar{z}(k-m) - \bar{z}(k-(m+1))\Big).
\end{aligned} \tag{5.5}
$$

where $(\bar{x}(k), \bar{y}(k), \bar{z}(k))$ is the averaged 3D position of the target object; $\phi_x(k)$, $\phi_y(k)$, and $\phi_z(k)$ are rotation angles around the x-, y-, and z-axes, respectively; $v_x(k)$, $v_y(k)$ and $v_z(k)$ are the average velocity in the x-, y-, and z-axes, respectively.

3) Depth-based Projection Mapping

According to the enhanced application tasks, the following depth-based projection-mapping algorithms are implemented on our camera-projector system.

5.2.2.2 Depth-based color mapping

a) Assignment of color properties

Corresponding to the 512×512 depth-image $D(X, Y, k)$, a color property $P(x, y, z)$ is assigned for 3D points (x, y, z) using the following color map function with respect to depth:

$$
P(x, y, z) = \begin{cases} Cm(z) & (x, y, z) \in S_k \\ \emptyset & (\text{otherwise}) \end{cases}. \tag{5.6}
$$

In our study, several color map functions $Cm(z)$ such as a cyclic jet color map and a reduced-color-depth map were implemented for sensitive and distinct depth visualization.

b) Projection of RGB light patterns

The color properties $P(x, y, z)$ were converted into a 1024×768 RGB light pattern $P(X'', Y'', k)$. (X'', Y'') is a pixel coordinate value in the RGB projection images. This conversion is conducted with a 3×4 projection matrix $T_{P'}$ for the RGB projector, which

indicates the relationship between the xyz- and the $X''Y''$-coordinate systems as follows:

$$H_{P'}\begin{pmatrix} X'' \\ Y'' \\ 1 \end{pmatrix} = T_{P'}\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}, \tag{5.7}$$

where $H_{P'}$ is a parameter, and the projection matrix $T_{P'}$ is obtained by prior calibration. Thus, 1024×768 RGB light patterns are projected for pixel-wise projection mapping from the RGB projector onto the measured 3D scene.

2) AR spirit level

a) Generation of spirit level patterns

A computer graphic (CG) pattern $G(x', y')$ is designed for the AR spirit level, in which two pointers are movable in a guide circle of radius R. The two pointers are located at $(a\phi_x(k), 0)$ and $(0, a\phi_y(k))$ on the vertical and horizontal axes of the circle in $G(x', y')$ so that their distances from the center of the circle increase sensitively with a large proportionality constant a, even when the rotation angles around the x- and y-axes are slightly small.

The CG pattern $G(x', y')$ is projected as color properties $P(x, y, z)$ of 3D points (x, y, z) on an approximated tangent plane of the target object. The approximated tangent plane involves the averaged 3D position $(\bar{x}(k), \bar{y}(k), \bar{z}(k))$, and its normal vector corresponds to the rotation matrix $R(\phi_x(k), \phi_y(k), \phi_z(k))$, expressed by the rotation angles $\phi_x(k)$, $\phi_y(k)$, and $\phi_z(k)$ around the x, y, and z-axes, respectively. The coordinate value (x, y, z) on the tangent plane is converted from (x', y') as follows:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = R(\phi_x(k), \phi_y(k), \phi_z(k))\begin{pmatrix} x' \\ y' \\ 0 \end{pmatrix} + \begin{pmatrix} \bar{x}(k) \\ \bar{y}(k) \\ \bar{z}(k) \end{pmatrix}. \tag{5.8}$$

b) Projection of RGB light patterns

The same process as that in 1-b), is conducted.

3) AR wristwatch

a) Generation of wristwatch patterns

A CG pattern $G(x'', y'')$ is designed for the AR wristwatch, which is a color wristwatch with a size of 185 × 185 pixels. when the target object moves and rotates under the illumination of IR projector, the CG pattern $G(x'', y'')$ is projected as $P(x, y, z)$ of 3D point (x, y, z) on an approximated tangent plane determined by the predicted centroid position and rotation matrix. The predicted centroid position is obtained according to the averaged 3D position $(\bar{x}(k), \bar{y}(k), \bar{z}(k))$, the moving speed $v_x(k), v_y(k), v_z(k)$, the rotation matrix $R(\phi_x(k), \phi_y(k), \phi_z(k))$ and the projection delay τ; the rotation matrix $R(\phi_x(k), \phi_y(k), \phi_z(k))$ is expressed by the rotation angles $\phi_x(k)$, $\phi_y(k)$, and $\phi_z(k)$ around the x, y, and z-axes, respectively. The coordinate value (x, y, z) on the tangent plane is obtained as follows:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = R(\phi_x(k), \phi_y(k), \phi_z(k)) \begin{pmatrix} x' \\ y' \\ 0 \end{pmatrix} + \begin{pmatrix} \bar{x}(k) \\ \bar{y}(k) \\ \bar{z}(k) \end{pmatrix} + \frac{1}{\tau} \begin{pmatrix} \bar{v_x}(k) \\ \bar{v_y}(k) \\ \bar{v_z}(k) \end{pmatrix}. \tag{5.9}$$

b) Projection of RGB light patterns

The same process as that in 1-b), is conducted.

5.2.3 Specifications

The procedures of structured light 3D measurement are accelerated by executing them in parallel with 512 blocks of 1×512 pixels on the GPU board. The 512×512 depth-image is outputted at 500 fps using pipelined parallel-processing for input images between 16 frames, as shown in Fig. 5.2. The total execution time for structured light 3D measurement, including the transfer time from the PC memory to the GPU board for a 512×512 input image is within 0.39 ms.

For depth-based projection mapping, 1024×768 RGB light patterns to be projected were also generated by executing processes in parallel on the GPU board. In the case of depth-based color mapping, color properties were assigned within 0.01 ms and RGB light patterns were generated within 0.01 ms. In the case of AR spirit level, CG patterns were

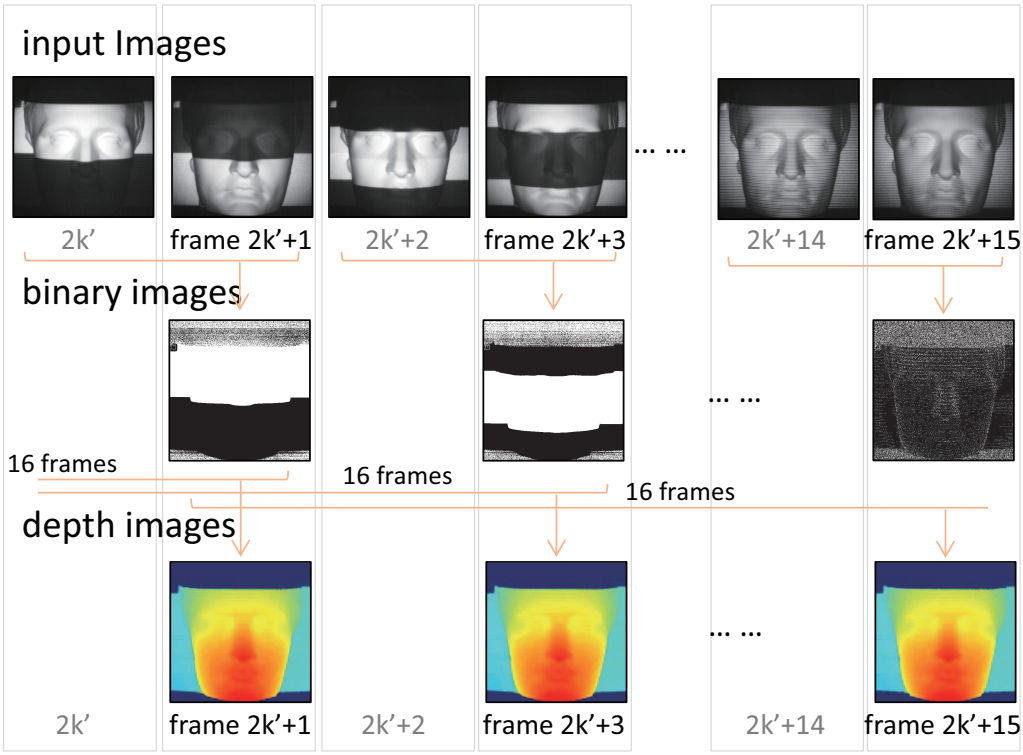
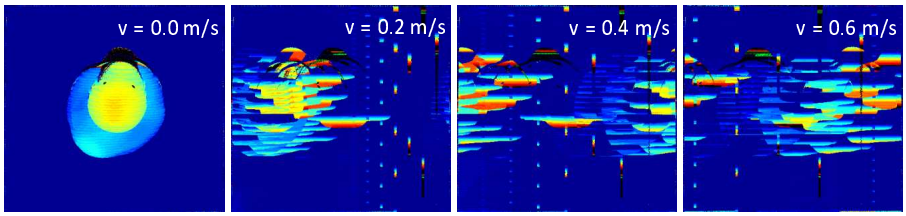


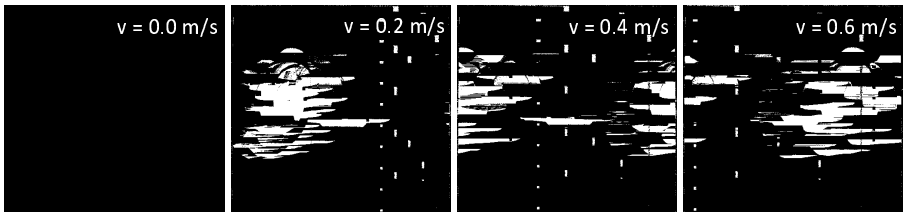
Figure 5.2: Pipelining-output of depth-image.

generated within 0.03 ms and RGB light patterns were generated within 0.01 ms. In both tasks, the transfer time from the GPU board to the PC memory for RGB light patterns was 1.46 ms. Including the depth-image acquisition in the structured light 3D measurement, the total execution times for (1) depth-based color mapping, (2) AR spirit level, and (3) AR wristwatch, were 1.88 ms, 1.89 ms, and 1.89 ms respectively. Thus, our HFR camera-projector system can generate projection patterns for projection mapping with low latency at the millisecond level, interactively with time-varying 3D scenes. Here, the projection rate of the RGB projector was limited at 60 fps when 1024×768 RGB light patterns were transferred from the PC via the HDMI video output. Thus in this study, the RGB light patterns were projected with a time delay of 18.0 ms at intervals of 16.7 ms, whereas the 512×512 depth-images were obtained in real time at 500 fps.

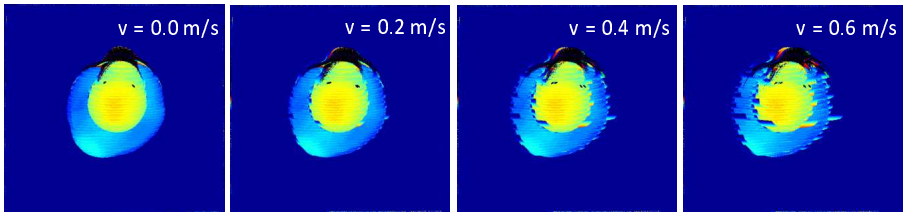
5.3 Experiments



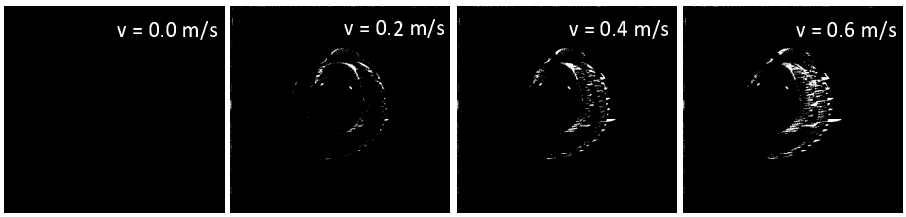
(a) Acquired depth image using 30 fps depth sensor



(b) Difference map between RGB light pattern and object (30 fps depth sensor)



(c) Acquired depth image using 500 fps depth sensor



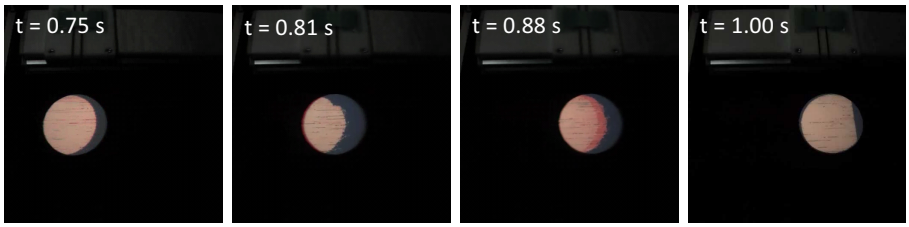
(d) Difference map between RGB light pattern and object (500 fps depth sensor)

Figure 5.3: Verification of HFR depth sensor.

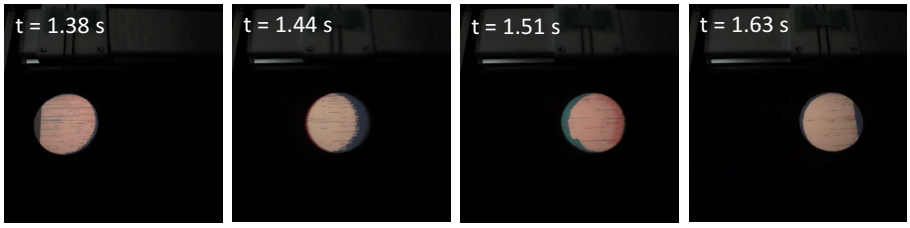
5.3.1 Robustness in dynamic projection mapping

In the structured light method, the Gray-code light patterns are projected sequentially to illuminate the measuring scene. The displacement d in captured images encountered in the 3D shape measurement of a moving object is proportional to its velocity v and inversely proportional to the frame rate f of depth vision , which causes inaccuracies in the measurements and leads to improper projection mapping.

$$d = v/f, \tag{5.10}$$



(a) Projection mapping without prediction

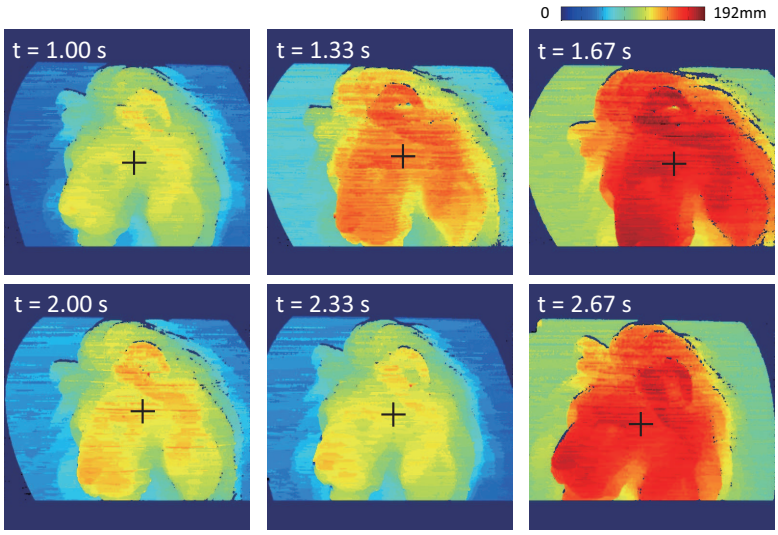


(b) Projection mapping with prediction

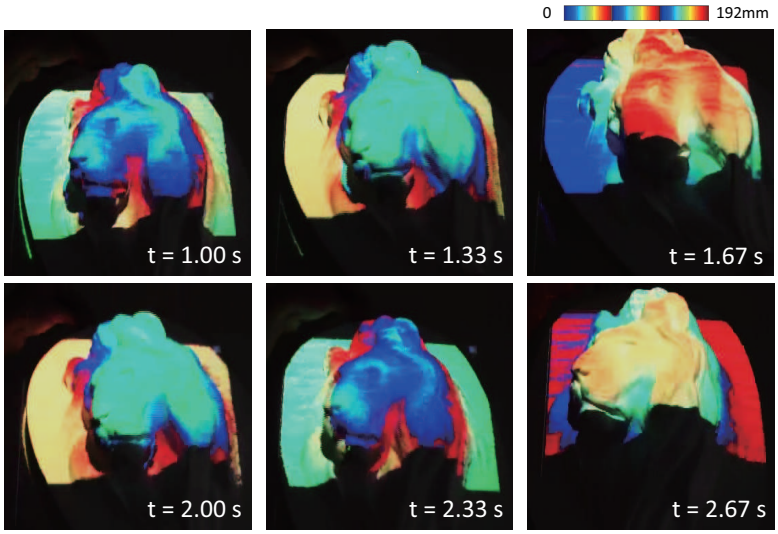
Figure 5.4: Verification of dynamic projection mapping.

To reduce such synchronization errors, we used 500 fps depth vision to shorten the time taken for the projection of multiple light patterns [96]. Its effectiveness was illustrated by a simulation experiment with an assumption of no projection delay. In this experiment, a toy duck that moved at various speeds, 0.0 m/s, 0.2 m/s, 0.4 m/s, and 0.6 m/s, was measured and mapped using a projection mapping system with a 30 fps depth sensor and our proposed system. Fig. 5.3 (a) and (c) show the extracted depth images obtained by a 30 fps depth sensor and 500 fps depth sensor, while Fig. 5.3 (b) and (d) present the difference maps between the RGB light pattern and the toy duck obtained by the two above-mentioned systems. Compared with depth images extracted by the 500 fps depth sensor, those extracted by the 30 fps depth sensor were not correct at $v = 0.2$ m/s, 0.4 m/s, and 0.6 m/s, when the toy duck was moving. However, the depth image extracted at $v = 0.0$ m/s, when the toy duck was static, was correct. For projection mapping, the rendering of our proposed system had a robust performance even when the object moved fast, while the rendering quality of the system using a low-frame-rate depth sensor was reduced sharply. This experiment illustrated that the impulsive noise and unmeasurable pixels around the edges of an object having discontinuous depth are reduced significantly with the help of an HFR depth sensor. Such features befit the projection mapping.

In our proposed system, the system delay composed of the time of depth-image



(a) depth-images



(b) color-mapped scenes

Figure 5.5: Experimental results for a moving lion relief.

extraction and the inherent projection delay of the projector reduces the performance of projection mapping. Concerning the extraction time of the depth-image, it is shortened to a level of 2 ms with the help of high-speed vision. Regarding the projection delay, the centroid position and rotation are involved to improve projection-mapping applications with a prediction algorithm. We present projection-mapping results of a disc moving over a disk plane. Fig. 5.4 (a) and (b) show the augmented experimental scenes captured by a standard video camera during the motion of the disc. Fig. 5.4 (b) is implemented including predication of motional state while the RGB images of (a) are projected onto

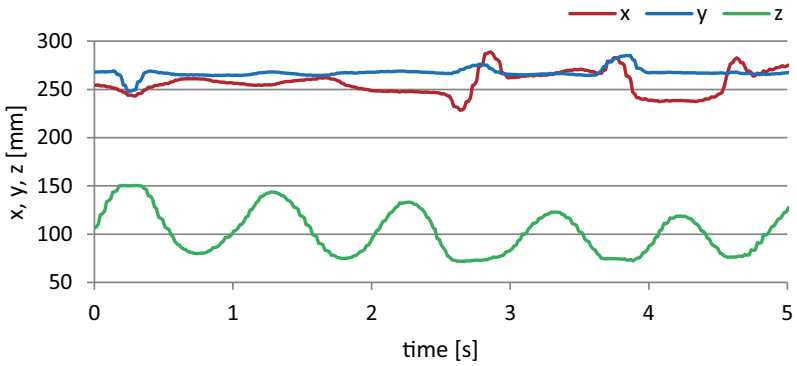


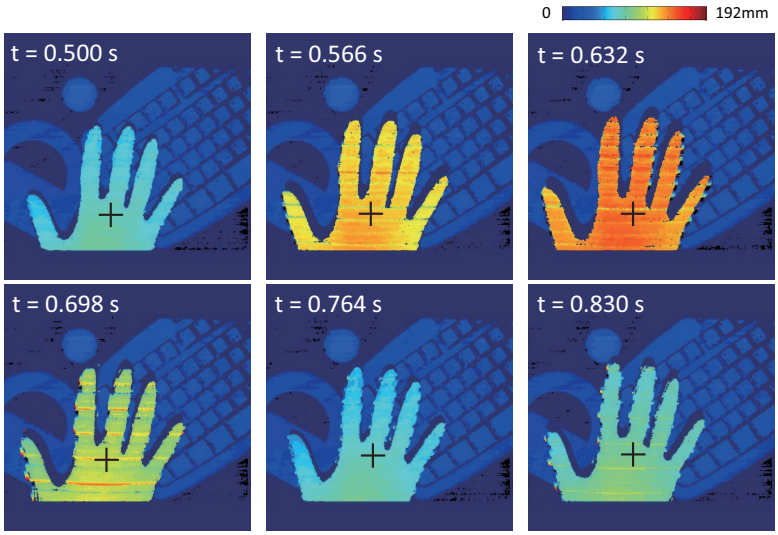
Figure 5.6: 3D position of the centroid position of a moving lion relief.

the measured scene directly. It can be seen that the improved projection mapping has better performance for dynamic scenes, since there is a smaller displacement between projected RGB image and object.

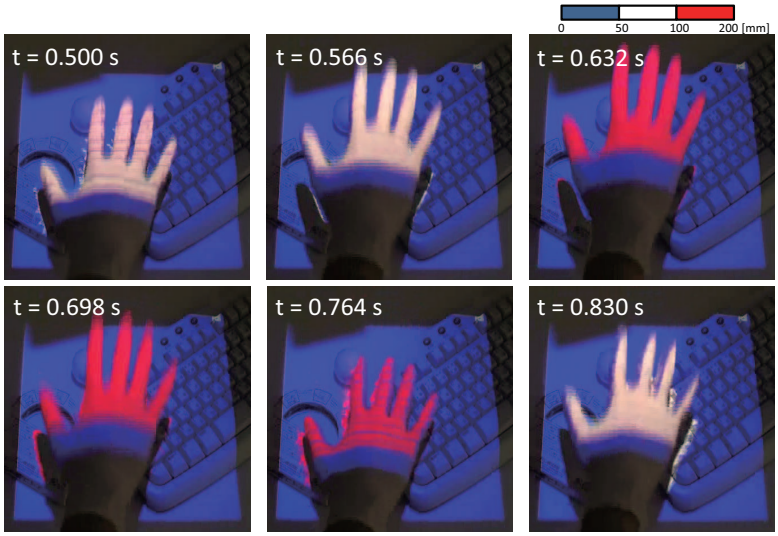
5.3.2 Depth-based Color Mapping

In this section, we look at experimental results obtained for projection mapping of (a) a moving plaster lion relief, and (b) a moving human hand over a desktop when the depth-based color mapping was implemented.

The plaster lion relief to be enhanced was moved above the level plane with periodic up-and-down motions once or less per second and slight rotations around the z axis by human hands. The height, width, and depth of the lion relief were 31 cm, 27 cm, and 10 cm, respectively. The reference depth-image was provided as the level plane, and the color map function was set to a cyclic jet color map. Fig. 5.5 shows (a) the color-mapped 512×512 depth-images, and (b) the experimental scenes captured using a standard digital video camera, taken at intervals of 0.33 s for t = 1.00–2.67 s. t = 0 was the start for the observation time. The x, y, and z coordinates of its centroid position for t = 0.0–5.0 s were measured as shown in Fig. 5.6. It can be seen that the 3D position of the relief was periodically changed in the z direction whereas the x and y coordinates were not changed as much, corresponding to its up-and-down movement. Corresponding to the measuring position and orientation, the 3D shape information of the relief was correctly measured in the depth-image in Fig. 5.5(a), even when the relief was moved up-and-down with slight rotation. In Fig. 5.5(b), the white-surface relief was enhanced with a cyclic jet color map,



(a) depth-images



(b) color-mapped scenes

Figure 5.7: Experimental results for a moving human hand.

which can directly visualize its detailed height information for the human eye, and the RGB light patterns were correctly projected for pixel-wise projection mapping on the moving lion relief. In the experiment, there remained slight displacements between the lion relief and the projected color map information when the relief was moved rapidly. These displacements were caused by the latency of the projector and video card used in the projection, which had a delay time of dozens of milliseconds.

We now consider the experimental results for a human hand periodically moved

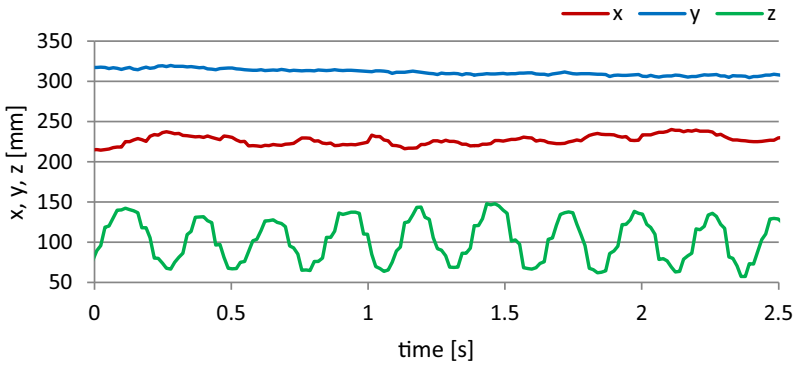


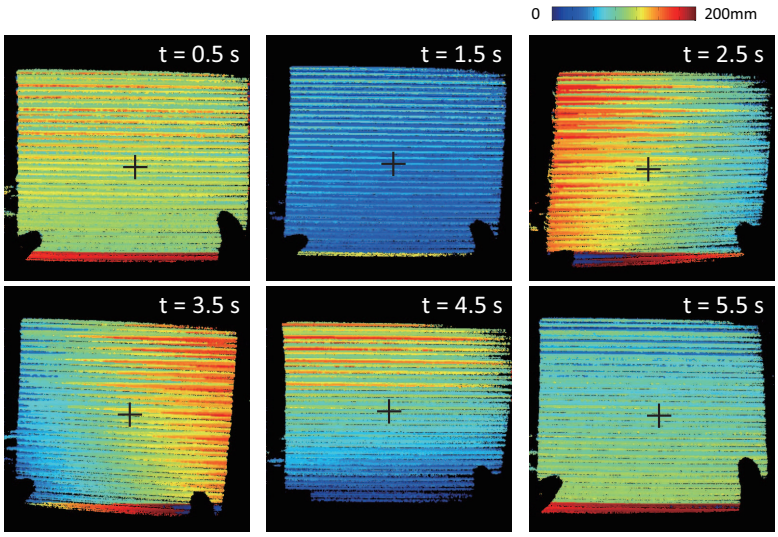
Figure 5.8: 3D position of the centroid position of a moving human hand.

over the desktop. The right hand was moved periodically at a frequency of approximately 3 Hz in the z direction from 55 to 145 mm above the level plane. On the level plane, a computer keyboard, books, and many 3D objects were placed as background objects. A precalculated reference depth-image was provided as the same background scene in the real-time experiment and the color map function was set to a three-color-depth color map; the target object was red-mapped when $z \geq z_{top} = 100$ mm, otherwise the target object was white-mapped; the background scene was always blue-mapped. Fig. 5.7 shows (a) the depth-images, and (b) the experimental scenes, which were taken at intervals of 0.066 s for $t = 0.50$–0.83 s. Fig. 5.8 shows the x, y, and z coordinates of the centroid position of the human hand for $t = 0.0$–2.5 s. The 3D position of the human hand was periodically changed three times per second in the z direction, corresponding to the periodic movement of the human hand. The 3D shape of the human hand and background objects were measured in Fig. 5.7(a). In Fig. 5.7(b), it can be seen that the human hand was highlighted with white or red colors depending on the 3D position of the human hand, whereas the background scene was always lighted in blue.

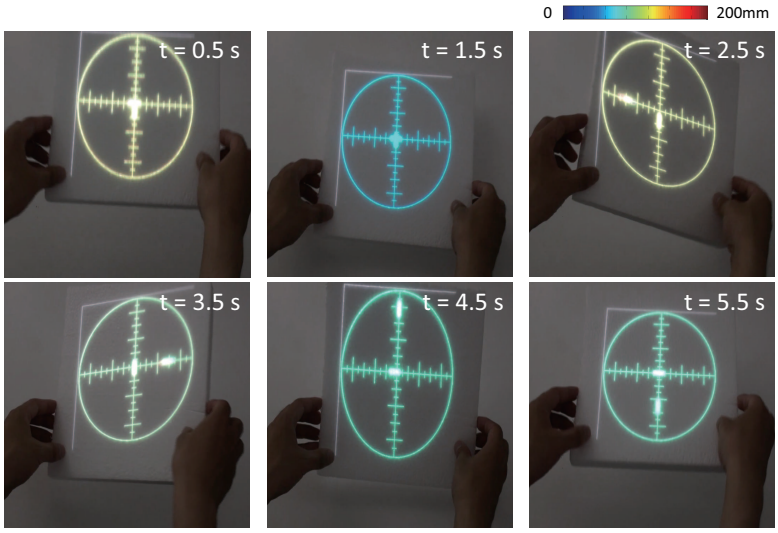
These results indicate that our system can execute real-time pixel-wise projection mapping on a moving 3D object for depth-enhanced visualization.

5.3.3 AR Spirit Level

Let us look at the experimental results for the AR spirit level. A 50×50 mm white plate was manually moved over the level plane; it was moved up and down, and then alternately tilted around the y axis and the x axis. The graphic pattern to be projected



(a) depth-images



(b) projection-mapped AR spirit level

Figure 5.9: Experimental results for AR spirit level.

involved a guide circle of radius 100 mm and two 20-mm-diameter pointers; the pointers moved 6 mm per degree for the rotation angles around the x and y axes, and the color of the graphic pattern was determined by the z coordinate of the target object. The reference depth-image was provided as the level plane.

Fig. 5.9 shows (a) the depth-images, and (b) the experimental scenes, taken at intervals of 1.0 s for t = 0.5–5.5 s. The rotation angles around the x- and y-axes and the z coordinate of the target object for t = 0.0–6.0 s were measured as shown in Fig. 5.10. The

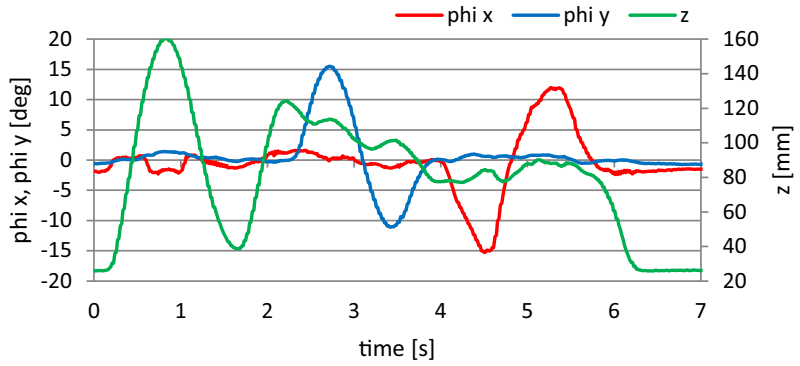


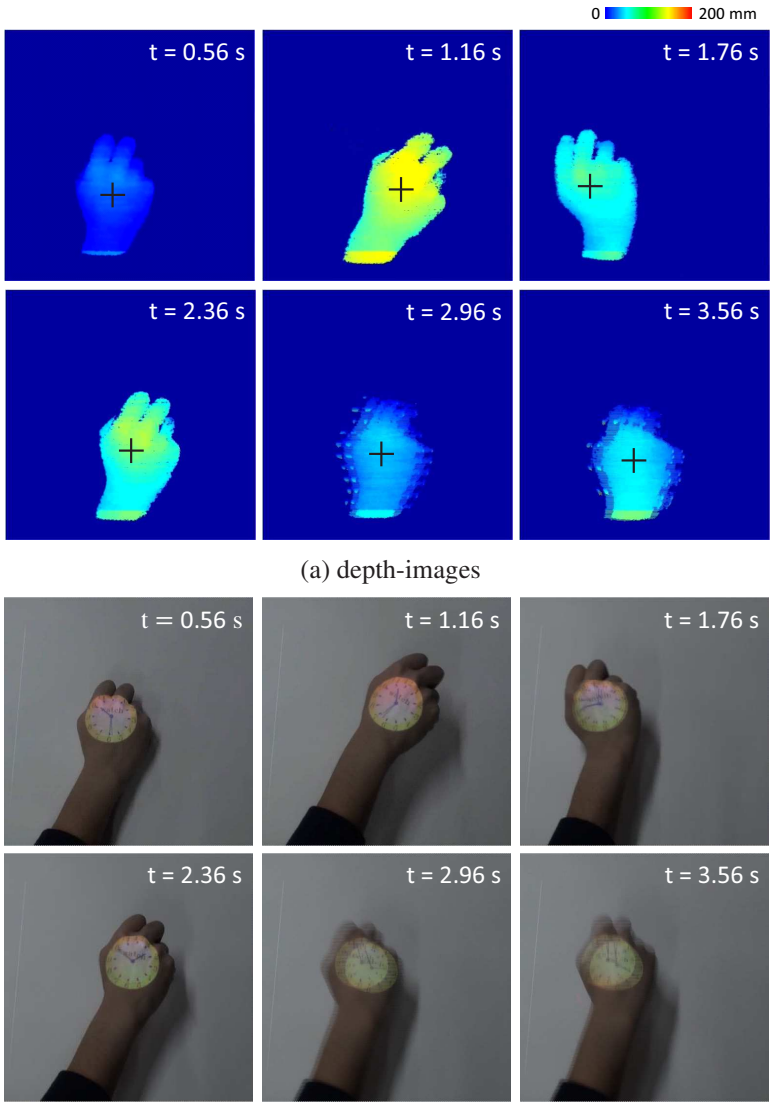
Figure 5.10: 3D orientation and z-coordinate of the centroid position of a white plate.

graphic pattern was projected at the center of the white paper and its pointer positions and color were correctly changed, corresponding to the rotation angles and the z coordinate of the white plate; the z coordinate was changed from 20 to 165 mm, and then the angles were alternately changed in the range from approximately -15 to 15 degree. Thus, it can be seen that the slight rotation angle at sub-degree level is enhanced for easy visualization on the white plane at AR spirit level, even when its rotation angle is too small for the human eye to inspect its slanted tendency without projection mapping.

5.3.4 AR wristwatch

This part presented the experimental results for AR wristwatch. In this experiment, the wrist was conducted reciprocating motion from one side to another side with arbitrary title around the $x-$ and $y-$axis. The projected CG wristwatch pattern has a size of 185×185 pixels with color property.

The depth-images and the experimental scenes are shown in Figure 5.11, which was taken at an interval of 0.60 for t =0.56–3.56 s; Figure 5.12(a) shows the 3D position (x, y, z) of the centroid position of human hand in this period, while Figure 5.12(b) presents rotation angles $(\phi(x), \phi(y))$ around the $x-$ and $y-$axis and the velocity (v_x, v_y) in this two directions. The graphic pattern was projected on the wrist corresponding to the 3D position, rotation angle and velocity; the z coordinate was changed from 45 to 165 mm, and then the angles and velocity changed in the range from around -20 to 20 degree and -2.2 to 2.2 m/s, respectively. It can be found that the 3D pose of a moving object can be tracked and enhanced in real-time for easy visualization by our system.



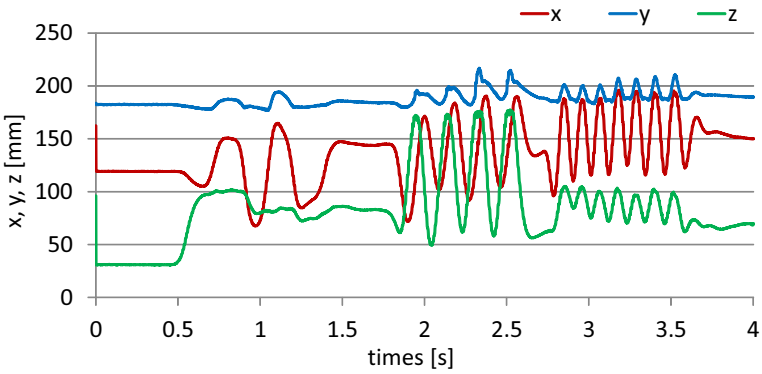
(a) depth-images

(b) color-mapped scenes

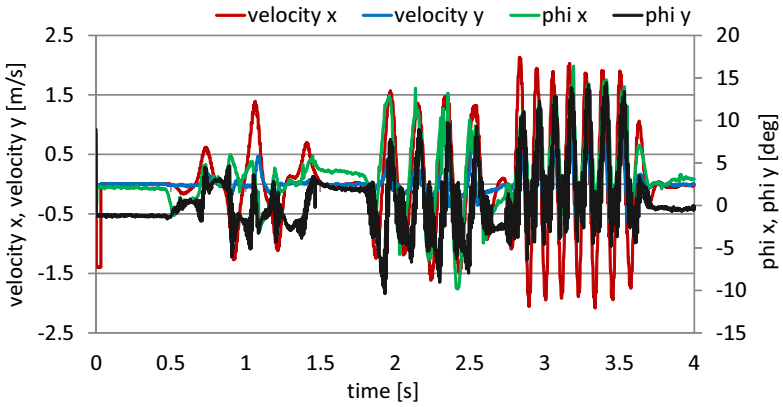
Figure 5.11: Experimental results for AR wristwatch.

5.4 Concluding Remarks

In this paper, we reported on the development of a real-time projection-mapping system which can acquire 512×512 depth-images in real time at 500 fps and project depth-based light patterns to be enhanced for time-varying 3D scenes. Experimental results from enhanced application tasks for dynamic 3D scenes verified the efficacy of our system. On the basis of the experimental results obtained, we plan to improve our camera-projector system for more responsive 3D projection mappings by accelerating the



(a) 3D position of the centroid position of human hand



(b) velocity and rotation angle of human hand

Figure 5.12: Moving state for AR wristwatch.

CG image generation for a projector with short time lag, and to extend the enhanced applications for various AR-based human-computer interactions. Meanwhile, we plan to implement the projection-mapping system using both horizontal and vertical projection patterns to exploit its advantages of robustness and extensibility. To improve the user experience, we plan to develop a graphical user interface that includes various parameters for device settings, such as brightness control of the RGB projector, to allow human users to interact with our proposed system in a user-friendly manner.

Chapter 6

A Fast 3D Shape Scanner Using Two HFR Camera-projector Systems

6.1 Introduction

The 3D shape extraction becoming increasingly important, with various applications in fields such as computer graphic, object recognition, and computer-human interface. For many of these applications, the entire shape acquisition of dynamic scenes in real-time is desirable. Restricted by the angle of view of a camera, the entire shape acquisition is realized on the basis of the cooperation of multiple 3D modules that locate at different perspectives, with an implementation of active or passive 3D shape measurement approach. The structured light illumination [97, 98, 99] is a well-known active 3D shape measurement method that is based on a camera-projector system. In such a system, the light patterns encoded with a certain strategy are projected onto the scene to be measured while a synchronized camera located at another perspective is used to capture deformation images; the 3D shape is obtained using the geometric properties between the camera and the projector. The structured light illumination is divided into sequential projection technique and one-shot projection technique according to the number of projection patterns. The passive measurement method [100] typically uses multiple cameras that work at normal ambient light, and the 3D reconstruction of the scene to be measured is realized by corresponding the similar image features in different image planes. Recently, many 3D scanning systems have been reported based on the above mentioned techniques; However, the frame rate of vision platform, the acceleration of the corresponding process, and the

restrictions on the color or texture of the surface of a measured scene present a challenge to improve their performance.

Our goal is to develop a robust, fast, and entire shape acquisition scanner that can generate accurate and dense 3D reconstruction for a dynamic scene including the color of its surface without posing restrictions on the texture and complexity of the scene. To this end, we proposed a 3D scanner that is based on the cooperation of two high-frame-rate (HFR) camera-projector systems. For the measurement of dynamic scenes [101], the execution of 3D data extraction has to be fast to allow more information to be obtained. To achieve high measurement accuracy and reduce time consumption, the proposed scanner is implemented with the Gray-code light patterns that are encoded in both horizontal and vertical, one of the sequential projection techniques. Compared with the one-shot projection that obtains the codeword of a pixel using local various or geometric constraints, the Gary-code projection is very reliable and robust to the surface characteristics since only binary values are used in the encoding strategy without concerning to the neighbourhood for a certain pixel. Meanwhile, in the Gray-code projection, the specific codeword of a pixel is obtained by corresponding the pixels at the same position in one pattern sequence that makes it feasible to accelerate by using a GPU board. However, the decoding procedure in one-shot projection is realized by using complicated image processing algorithms which give a challenge to speed up. Typically, the sequential projection is sensitive to the motion of a target object, because the synchronization errors caused by misreading the light patterns projected at different points as those at the same one occur when moving objects are under observation. To reduce this effect, we implement the structured light illumination on an HFR camera-projector system composing of a high-speed vision platform and a DLP high-speed projector to minish the displacement among one pattern sequence, whose effectiveness has been illustrated by Gao [102]. When two or more camera-projector systems are employed by one scanner, the projection interference caused by the light patterns are projected onto the measured scene from different projectors at the same time is a thorny problem. In our study, a frame-straddling high-speed vision platform, in which the time delay between the two camera heads can be controlled, is developed by implementing the hardware logics on the HFR vision used in

camera-projector system. Meanwhile, the two projectors are synchronized with the two cameras by using impulse signal. In the implementation, the illumination time of the projector is shorter than the exposure time of the camera to keep that the measured scene is illuminated only by a light pattern in one exposure period of the camera.

In Section 6.2 of this paper, we describes related works on 3D scanner that uses structured light illumination. In Section 6.3, the configuration of our proposed system and a outline of the implemented algorithm are presented. In Section 6.4, several experimental results of the measurement of moving scenes are presented to verify its performance.

6.2 Previous Works

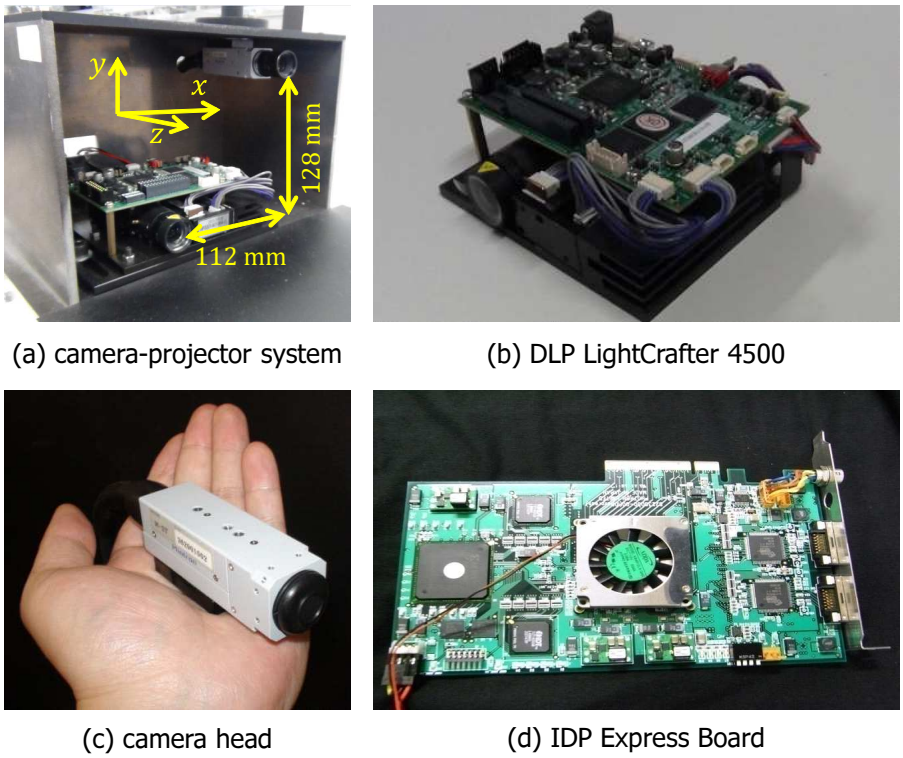
Recently, various structured light illumination-based scanning systems that focus on simultaneous 3D measurement and rapid display have been reported. By projecting light patterns using certain codification strategies [97, 98, 99], the structured light illumination realizes fast and robust correspondence between light patterns and captured images with a significantly reduced corresponding complexity. However, the improvement of measurement accuracy and execution speed still poses a major challenge to their performance.

The Gray-code projection proposed by Innokuchi [103] uses multiple stripes with binary information to form a sequence of light patterns. When the surface of the scene to be measured is illuminated under such light patterns, each point on the surface processes a unique codeword that differs from the other points; a large number of light patterns is necessary for the high accuracy and high resolution measurement. Gao et al. [102] reported a real-time 3D scanner that can output 3D video of 512×512 pixels at 500 fps on the basis of 8-bit Gray-code light patterns and a GPU-based HFR camera-projector system, which has been expanded to be a real-time projection mapping system [104]. Liu et al. [105] presented a high-speed 3D vision with automatic motion compensation that can reduce the synchronization errors that exists in Gray-code projection when a moving object is measured.

Phase-shift [106] is another sequential projection technique that encodes the codeword using a set of sinusoidal patterns. One of the outstanding features of this method is

that it reduces the number of light patterns without losing resolution and accuracy. However, depth exceeding the range equivalent to one fringe period can not be unwrapped uniquely due to the periodicity of the fringe signal. T. Weise et al. [107] developed a 3D scanning system combining stereo and phase-shit projection for a dynamic scene reconstruction, which can output accurate depth maps at 17 fps. Zhang et al. [109] reported a multilevel quality-guided phase unwrapping algorithm for real-time 3D shape measurement that can generate 640×480 depth maps at 30 fps. A real-time, high-precision 3D scanning system [110] with an off-the-shelf projector and a commodity camera is proposed which can eliminate the phase ambiguity by embedding a speckle-like signal in the sinusoidal fringe patterns.

The 3D measurement system that is on the basis of the one-shot projection technique encodes the codeword purely in the spatial domain; correspondence is established by searching similar features in the light pattern and captured images via a spatial window. For these systems, they are robust to the motion of a target object, since only one acquired image of the scene under such light pattern illumination is required to derive the full depth image. However, they become unstable when there are color or texture on the surface of the scene to be measured, while they are not very effective in real time because of the high computational complexity of correspondence. Microsoft Kinect is a commercial product that can acquire the 3D shapes of measuring scenes at 30 fps by using a spatially coded infra-red light pattern with an assumption of local smoothness. Kawasaki et al. [112] proposed a dense 3D reconstruction method using a single color code pattern for fast moving object; an entire shape acquisition system using multiple cameras and projectors [111] is realized based on such method. The 3D scanning system using Hamming color code [113] or De Bruijn sequence [114] are reported for fast and accurate measurement. Some other types light patterns that establish the correspondence using geometric constraints [115, 116] are also proposed.



(a) camera-projector system (b) DLP LightCrafter 4500

(c) camera head (d) IDP Express Board

Figure 6.1: Composing elements of the proposed system.

6.3 Fast 3D shape scanner

6.3.1 System Configuration

The proposed fast 3D shape scanner consists of a dual-camera high-speed vision platform (IDP Express board) with a frame-straddling function for two cameras inputs, a personal computer (PC) installed with a GPU board (NVIDIA Tesla 1060) for parallel processing, and two separate HFR camera-projector systems for 3D measurement that are placed at opposite sites so that they can capture different profiles of the measured scene. The composing elements of the proposed system and system overview are shown in Fig. 6.1 and Fig. 6.2, respectively.

The IDP Express is designed to implement various types of real-time video processing algorithms and recording of 512×512 images at high frame rate that can be mapped onto the PC memory simultaneously via the PCI-e bus. It comprises two camera inputs to connect camera heads, trigger I/Os for external synchronization, and a dedicated FPGA board (Xilinx XC3S5000) for the hardware implementation of algorithms. The detailed

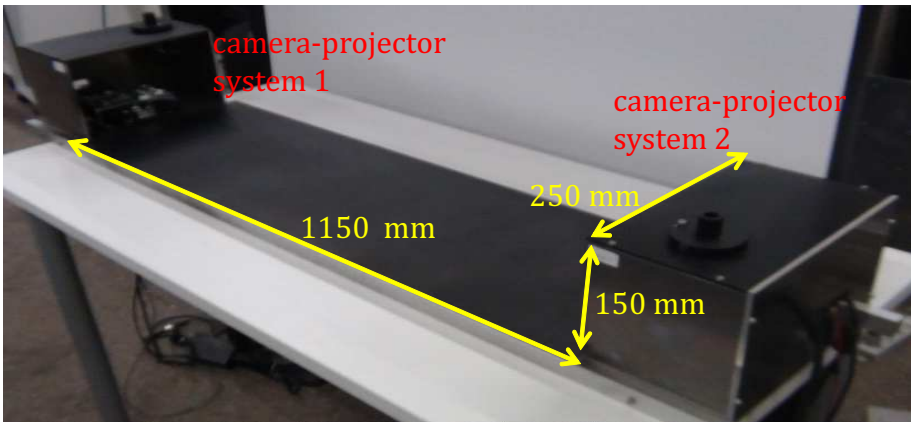


Figure 6.2: System overview.

explanation of image processing path and transfer in IDP Express board can be found in [117]. The overview of the board is shown in Fig. 6.1 (d). In our study, the IDP Express board was dedicated and improved for dual-camera frame-straddling by developing the hardware logics for time delay control between the two camera heads. The time delay between the two camera heads is simultaneously controlled via the PC from 0 to 1 ms in 9.9 ns steps in capturing 512×512 images at 500 fps. Thus, the IDP Express can work as a frame-straddling high-speed vision platform [118].

The Tesla C1060 is a computer processor board on the basis of the NVIDIA Tesla T10 GPU; it is capable of a processing performance of 933 Gflops/s using 240 processor cores operating at 1.296 GHz and a bandwidth of 102 GB/s for its internal 4 GB memory. A PC with a 16-lane PCI-e 2.0 bus and a processor chipset with a DMA function are adapted to transfer memory-mapped data between standard memory and the Tesla 1060 at high speed via the PCI-e bus. The PC has the following specifications: ASUSTeK P6T7 WS main board, Intel Core i7 3.20 GHz CPU, 3 GB of memory, two 16-lane PCI-e 2.0 buses, and Windows 7 Professional 32 bit OS. In this study, the implemented algorithm for 3D measurement and registration was accelerated by parallel processing on the Tesla 1060.

The camera-projector system composes of a high-speed DLP projector (Texas Instruments DLP LightCrafter 4500) and a color camera head, which is shown in Fig. 6.1. The DLP LightCrafter 4500 is a compact evaluation module enabling high-speed binary

light pattern projection using a DMD device, which can project hundreds of 854×480 binary patterns at 500 fps or more in synchronization with an external system. The camera head can capture 8-bit Bayer images of 512×512 pixels at 5000 fps or more, and the input images can be mapped in the PC memory at the same frame rate. As shown in Fig. 6.2, two HFR camera-projector systems are placed at the opposite sites on the module with a distance between them to be 1150 mm. To eliminate the projection interference, the time delay between the two cameras is controlled to be 1 ms, while the illumination time of the projectors and the exposure time of the cameras are set to be 0.5 ms and 0.7 ms. respectively.

6.3.2 Implemented Algorithm and Its Specification

To realize fast entire shape reconstruction, we implemented the structured light 3D measurement and registration using multiple threads processing: one threads for image capturing, two threads for 3D measurement, one thread for data registration, and one thread for display using OpenGL. The flowchart of the implemented algorithm is shown in Fig. 6.3.

(a) Projection of Gary-code light patterns

Twenty pairs of positive and negative light patterns with 10-bit Gray-code encoded in both horizontal and vertical are projected in the order of $\{g_x^0, g_x^1\}$, $\{g_y^0, g_y^1\}$, $\cdots$, and $\{g_y^{18}, g_y^{19}\}$ as follows:

$$\begin{aligned} g_x^{2i}(X', Y') &= \left\lfloor \frac{2^i \cdot Y'}{480} + \frac{1}{2} \right\rfloor \bmod 2 \\ g_x^{2i+1}(X', Y') &= \overline{g_x^{2i}(X', Y')} \end{aligned} \quad (i = 0, \cdots, 9), \tag{6.1}$$

$$\begin{aligned} g_y^{2i}(X', Y') &= \left\lfloor \frac{2^i \cdot X'}{854} + \frac{1}{2} \right\rfloor \bmod 2 \\ g_y^{2i+1}(X', Y') &= \overline{g_y^{2i}(X', Y')} \end{aligned} \quad (i = 0, \cdots, 9), \tag{6.2}$$

where $\lfloor x \rfloor$ is the minimum integer that is greater than or equal to x; i denotes the LSB and MSB order in the gray code. (X', Y') is a pixel coordinate value in the projection images.

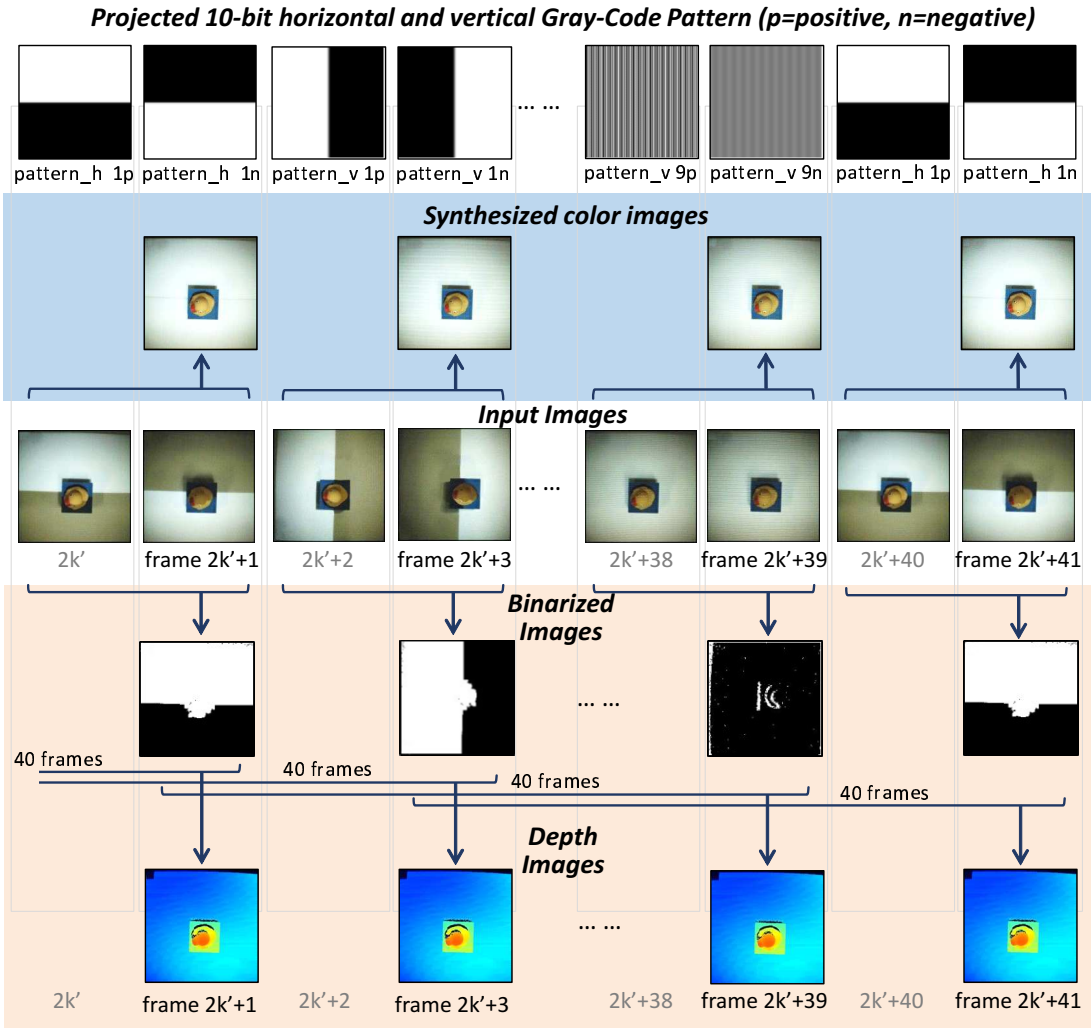


Figure 6.3: Flowchart of implemented algorithm.

(b) Image acquisition

Corresponding to the projection image of $g(X', Y')$, a Bayer image of 512×512 pixels is captured at time $t = k\tau$ as follows:

$$^{b}I(X, Y, k) = \mathrm{Proj}(g(X', Y')), \tag{6.3}$$

where k is the frame number of the captured images, and the frame interval is $\tau = 2$ ms for each camera. (X, Y) represents the coordinate value of a pixel in the captured images.

(c) Bayer-to-gray-level conversion

A gray-level image $I(X, Y, k)$ of 512×512 pixels is obtained by using a color image

${}^{RGB}I(X, Y, k)=[R, G, B]$, which is converted from an acquired Bayer image as follow:

$$I(X, Y, k) = \text{Graylevel}({}^{RGB}I(X, Y, k)). \tag{6.4}$$

$${}^{RGB}I(X, Y, k) = \text{Bayer}({}^{b}I(X, Y, k)). \tag{6.5}$$

(d) Binarization

A pair of images $I(X, Y, 2k')$ and $I(X, Y, 2k' + 1)$ are differentiated on the basis of robustness to ambiguities due to nonuniform brightness at 2 ms intervals. The binary image for space encoding is obtained with a threshold θ_b as follows:

$$G(X,Y,2k'+1) = \begin{cases} 1 & I(X,Y,2k') - I(X,Y,2k'+1) > \theta_b \\ 0 & I(X,Y,2k') - I(X,Y,2k'+1) < -\theta_b, \\ \phi & \text{(otherwise)} \end{cases} \tag{6.6}$$

where ϕ denotes the ambiguous binarization state.

(e) Gray-to-binary conversion

Corresponding to the light patterns that encoded in two directions, the binarization image $G(X, Y, 2k' + 1)$ is converted with a pure binary code at the current and previous frames in such two directions as follows

$$B_h(X, Y, k') = \left(\sum_{i=k' \bmod 10}^{9} G(X, Y, 4(k'-i)+1) \right) \bmod 2, \tag{6.7}$$

$$B_v(X, Y, k') = \left(\sum_{i=k' \bmod 10}^{9} G(X, Y, 4(k'-i)+3) \right) \bmod 2, \tag{6.8}$$

where we consider the unmeasurable state when there are two or more ambiguous binarizations among the ten frames; 0 is substituted for ϕ when there is one ambiguous binarization.

(f) Space-code images generation

Two space-code images $C_h(X, Y, k')$ and $C_v(X, Y, k')$ are obtained using the binary images $B_h(X, Y, k)$ and $B_v(X, Y, k)$ $(i = 0, \cdots, 9)$ at the current and previous frames as follows:

$$C_h(X, Y, k') = \sum_{i=0}^{9} 2^{k' \bmod 10} \cdot B_h(X, Y, k'), \tag{6.9}$$

$$C_v(X, Y, k') = \sum_{i=0}^{9} 2^{k' \bmod 10} \cdot B_v(X, Y, k'). \tag{6.10}$$

The space code images are filtered with a 3×3 median filter.

(g) Triangulation

A 3D map (x, y, z) is obtained by solving a simultaneous equation with space-code images $C_h(X, Y, k')$ and $C_v(X, Y, k')$.

$$H_C \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} = T_C \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}, H_P \begin{pmatrix} C_v(X, Y, k') \\ C_h(X, Y, k') \\ 1 \end{pmatrix} = T_P \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}, \tag{6.11}$$

The matrices T_C and T_P are obtained by prior calibration while H_C and H_P are parameters. The depth image $D(X, Y, k')$ is filtered by a 5×5 median filter. Here, we denote the 3D map obtained by the right camera-projector system as (x_r, y_r, z_r), while the other one is marked as (x_l, y_l, z_l).

(h) Recovery of color RGB image

The color information ${}^{RGB'}I(X, Y, k')$ of a measured scene is recovered by extracting the RGB value in ${}^{RGB}I(X, Y, 2k')$ or ${}^{RGB}I(X, Y, 2k' + 1)$ by comparing their gray-level images:

$${}^{RGB'}I(X,Y,k') = \begin{cases} {}^{RGB}I(X,Y,2k') & I(X,Y,2k') > I(X,Y,2k'+1) \\ {}^{RGB}I(X,Y,2k' + 1) & \text{(otherwise)} \end{cases}, \tag{6.12}$$

Table 6.1: Execution time of proposed algorithm.

	Time [ms]
Image acquisition time	0.03
Transfer to GPU	0.08
Bater-to-gray-level conversion	0.01
Binarization	0.01
Gray-to-binary conversion	0.02
Space code image generation	0.02
3×3 median filter	0.02
3D triangulation	0.01
Recovery of color RGB image	0.01
Registration	0.01
Transfer to PC memory	1.21
Total time	1.43

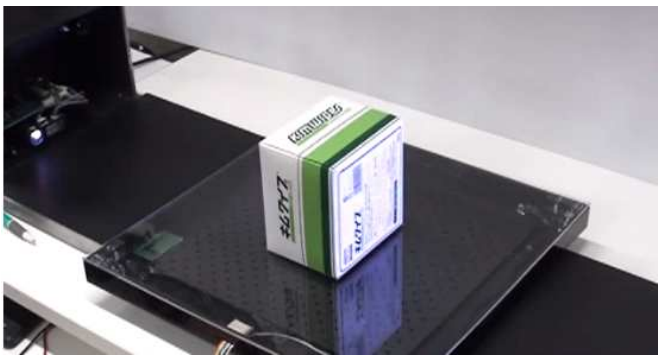
(i) Registration

The entire shape (x', y', z')= $\left((x_r^{'}, y_r^{'}, z_r^{'}) \cup (x_l^{'}, y_l^{'}, z_l^{'})\right)$ is obtained by transforming the 3D map extracted by the two camera-projector systems to the world coordinate system with (R_r, T_r) and (R_l, T_l) to be their rotation matrix and translation matrix as follows:

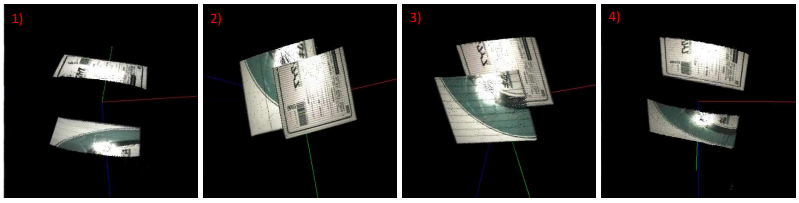
$$\begin{pmatrix} x_r^{'} \\ y_r^{'} \\ z_r^{'} \end{pmatrix} = R_r \begin{pmatrix} x_r \\ y_r \\ z_r \end{pmatrix} + T_r, \tag{6.13}$$

$$\begin{pmatrix} x_l^{'} \\ x_l^{'} \\ x_l^{'} \end{pmatrix} = R_l \begin{pmatrix} x_l \\ y_l \\ z_l \end{pmatrix} + T_l. \tag{6.14}$$

In the implementation of the above mentioned algorithm, process (1) is conducted offline to generation light patterns, while processes (b)-(i) are parallel processed with 512 blocks of 1 × 512 pixels on the GPU board. Table 6.1 presents the total execution time and its breakdown when images of 512×512 pixels are processed, which includes the time of image acquisition, the transfer time from PC memory to GPU board for a captured image, and the transfer time from GPU board to PC memory for registered 3D data. From



(a) experimental scenes



(b) OpenGL view of reconstruction result

Figure 6.4: 3D reconstruction for static box.

this table, we can find that the total time of execution is 1.43 ms so that the proposed system can work in real time at 250 fps. The registered 3D data with color information is displayed using OpenGl at 20 fps.

6.4 Experiments

To show the effectiveness of our proposed scanning system, two experiments, 3D reconstruction for a static box and 3D reconstruction for a moving human-hand, are conducted. In these experiments, the binarization threshold θ_b for the structured light 3D measurement was set to be 2; while the coordinate system of the optical center of the right camera-projection system is selected to be the world coordinate system.

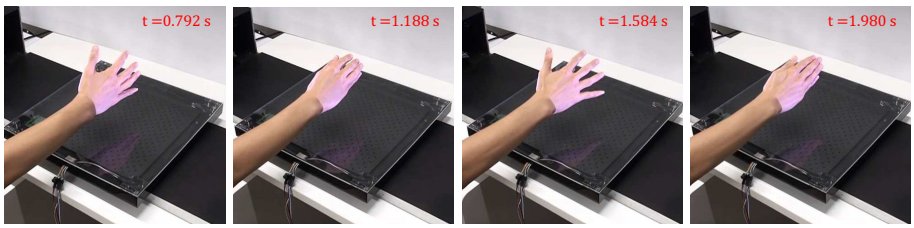
6.4.1 A Color-patterned Static Box

In this experiment, a color box of size 150×150×80 mm placed on the middle of the scanner is under the illumination of Gray-code pattern; the scene to be measured is shown in Fig. 6.4 (a). Fig. 6.4 (b) shows the reconstructed results for the box, which were presented at four different viewpoints. It can be found that the two camera-projector

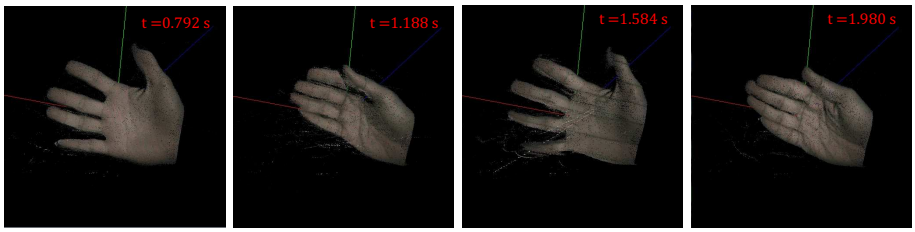
systems used in our proposed scanner can extract dense and accurate 3D shape of the measured scene with color information. Meanwhile, we can observe that the general shape of the box is obtained by registering the two individual 3D data together.

6.4.2 A Color-patterned Moving Human-hand

In this experiment, we show the effectiveness of the proposed system for a moving object. A human-hand periodically opened and closed with arbitrary rotation at a frequency of approximately 1 times per second on the middle of the scanner. Fig. 6.5 (a) presents the motion states of the human-hand, which were captured at 30 fps by using a standard video camera at intervals of 0.396 s for t=0.792-1.980 s. Fig. 6.5 (b) shows the reconstructed results obtained by our proposed scanner at the same time. Observing from these figures, we can find that the entire 3D shape of a moving human-hand can be reconstructed correctly even it moves fast. Through this experiments we confirm that the proposed scanner system can realized entire 3D shape reconstruction in real-time at 250 fps by registering the 3D data obtained by two individual camera-projector systems together. Meanwhile, It also can be that the synchronization errors occurs during the procedure of reconstruction.



(a) experimental scenes



(b) OpenGL view of reconstruction result

Figure 6.5: 3D reconstruction for a moving human-hand.

6.5 Concluding Remarks

In this paper, we developed a GPU-based fast 3D shape reconstruction scanner system that can output 3D video at 250 fps by using two individual HFR camera-projector systems. Several reconstruction experiments for both static scene and dynamic scene are performed to verify its effectiveness. On the basis of the experimental results obtained, we plan to improve our scanner system using more camera-projectors to realize a reconstruction of dead-zone free. Furthermore, an automatic motion-compensation algorithm will be employed to improve its performance for moving object scanning.

Chapter 7

Conclusion

With the development of computer vision, the computer-vision-based 3D measurement is becoming increasely important in many application fields, such as robot feedback control, industrial inspection, and entertainment. Many research groups have developed real-time 3D measurement systems based the time-varying projection method, spatial projection method, or space-time projection method. However, there are limitations in the existing method: time-varying light projection method without involving tracking strategies are reliable for static objects but their accuracies decrease sharply for moving objects because of the synchronization errors; the spatial projection methods that encode purely in the spatial domain measure the target scenes with an assumption of local smoothness; space-time projection techniques utilize the advantages of both the space domain and the time domain with a heavy computation time. On the other hand, limited by the video signal formats (e.g. NTSC 30 fps and PAL 25 fps) that are designed on the characteristics of human, most of the existing 3D measurement systems work at a level of tens frame. To overcome the restrictions of conventional systems, I developed two methods: a sparse blink-spot projection method without synchronization errors and a density motion-compensated stripe projection method with reduced synchronization errors.

Firstly, I proposed a blink-spot projection method that can reduce the synchronization errors of the sequential structured light illumination, which are caused by multiple light patterns projected with different timings when fast-moving objects are observed. In this method, a series of spot array patterns, whose spot sizes change at different timings corresponding to their identification number, is projected onto scenes to be measured by

a high-speed projector. Based on simultaneous and robust frame-to-frame tracking of the projected spots using their ID numbers, the 3D shape of the measuring scene can be obtained without misalignments, even when there are fast movements in the camera view. I implemented our method with an HFR projector-camera system that can process 512×512 pixel images in real-time at 500 fps to track and recognize 16×16 spots in the images.

In the second part, I introduced an HFR structured light vision that can simultaneously obtain depth images of 512×512 pixels at 500 fps by implementing a motion-compensated coded structured light method on an HFR camera-projector platform; the 3D computation is accelerated using the parallel processing on a GPU board. This method can remarkably reduce the synchronization errors in the structured-light-based measurement.

In the third part, a projection-mapping system is developed that can project RGB light patterns that are enhanced for 3D scenes using a graphics processing unit (GPU) based HFR vision system synchronized with HFR projectors. This system can acquire 512×512 depth-images in real time at 500 fps. The depth-images processing is accelerated by installing a GPU board for parallel processing of Gray-code structured light illumination using infrared light patterns projected from an IR projector. Using the computed depth-image, suitable RGB light patterns to be projected are generated in real time for enhanced application tasks. They are projected from an RGB projector as augmented information onto a 3D scene with pixel-wise correspondence even when the 3D scene is time-varied.

At last, A fast 3D shape scanner is developed that can output 3D video at 250 fps using two HFR camera-projector systems with an implementation of 10-bit Gray code light pattern encoded in both horizontal and vertical. The 3D data, which is extracted by the two camera-projector systems, are registered together to obtain an entire shape. To avoid the interference of projection patterns, the high-speed vision platform used in this system is dedicated and improved for dual-camera frame-straddling by developing a hardware logics for the time delay control between the two cameras.

In the future, I want to combine the blink-spot projection method and motion-compensated stripe projection method to extend its applicabilities to various fields con-

sidering the high-speed and real-time processing property, such as robot feedback control, industrial inspection and entertainment.

Bibliography

[1] T. M. Bernard, B. Y. Zavidovique, and F. J. Devos, "A programmable artificial retina," *IEEE J. Solid-State Circuits,* vol. 28, no. 7, pp. 789–797, 2013.

[2] J. E. Eklund, C. Svensson, and A. Astrom, "VLSI implementation of a focal plane image processor - A realization of the near-sensor image processing concept," *IEEE Trans. VLSI Systems,* vol. 4, no. 3, pp. 322–335, 1996.

[3] T. Komuro, I. Ishii, and M. Ishikawa, "Vision chip architecture using generalpurpose processing elements for lms vision system," *Proceedings of IEEE International Workshop on Computer Architecture for Machine Perception,* pp. 276–279, 1997.

[4] M. Ishikawa, K. Ogawa, T. Komuro, and I. Ishii, "A cmos vision chip with simd processing element array for lms image processing," *Proceedings of IEEE International Solid-State Circuits Conference,* pp. 206–207, 1999.

[5] T. Komuro, S. Kagami, and M. Ishikawa, "A dynamically reconfigurable SIMD processor for a vision chip," *IEEE J. Solid-State Circ.,* vol. 39, no. 1, pp. 265–268, 2004.

[6] I. Ishii, K. Yamamoto, and M. Kubozono, "Higher order autocorrelation vision chip," *IEEE Trans. Electron Dev.,* vol. 53, no. 8, pp. 1797–1804, 2006.

[7] S. Hirai, M. Zakoji, A. Masubuchi, and T. Tsuboi, "Realtime FPGA-based vision system," *J. Robot. Mechat.*, 17(4) (2005) 401–409.

[8] Y. Watanabe, T. Komuro, and M. Ishikawa, "955-fps real-time shape measurement of a moving/deforming object using high-speed vision for numerous-point analysis," *Proc. IEEE Int. Conf. Robot. Autom.*, pp. 3192–3197, 2007.

[9] I. Ishii, R. Sukenobe, T. Taniguchi, and K. Yamamoto, "Development of high-speed and real-time vision platform, H^3 Vision," *Proc. IEEE/RSJ Int. Conf. Intelli. Rob. Sys.*, pp. 3671–3678, 2009.

[10] I. Ishii, T. Tatebe, Q. Gu, Y. Moriue, T. Takaki and K. Tajima, "2000 fps real-time vision system with high-frame-rate video recording," *IEEE Int. Conf. Rob. Autom.*, pp. 1536–1541, 2010.

[11] K. Yamamoto, T. Tsuji, S. Ikeda, and I. Ishii, "A Real-time Finger-tapping Interface Using High-speed Vision System," *Proc. IEEE Int. Conf. on Systems, Man, and Cybernetics*, pp. 296–303, 2006.

[12] I. Ishii, I.Ohara, T. Tatebe, T. Takaki, "1000-fps Target Tracking Using Vibration-based Image Features," *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 1837–1842, 2011.

[13] Y. Nakabo, M. Ishikawa, H. Toyoda, and S. Mizuno, "1 ms column parallel vision system and its application of high speed target tracking," *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 650–655, 2000.

[14] Y. Nakabo, I. Ishii, and M. Ishikawa, "3D tracking using two high-speed vision systems," *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 360–365, 2002.

[15] A. Namiki, Y. Imai, M. Ishikawa, and M. Kaneko, "Development of a high-speed multifingered hand system and its application to catching," *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2666–2671, 2003.

[16] D. Shiokata, A. Namiki, and M. Ishikawa, "Robot Dribbling Using a High-Speed Multifingered Hand and a High-Speed Vision System," *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3945–3950, 2005.

[17] S. Mizusawa, A. Namiki, and M. Ishikawa, "Tool Manipulation by a Multifingered Hand Using a High-speed Vision," *Proceedings of the 8th SICE System Integration Division Annual Conference*, pp. 55–66, 2007.

[18] L. Chen, H. Yang, T. Takaki, and I. Ishii, "Real-Time Optical Flow Estimation Using Multiple Frame-Straddling Intervals," *Journal of Robotics and Mechatronics*, Vol.24, No.4, pp. 686–698, 2012.

[19] Q. Gu, A. Noman, T. Aoyama, T. Takaki, and I. Ishii, "A Fast Color Tracking System with Automatic Exposure Control," *Proceeding of the IEEE International Conference on Information and Automation*, pp. 1302–1307, 2013.

[20] Y. Nie, I. Ishii, K. Yamamoto, K. Orito, and H. Matsuda, "Real-time scratching behavior quantification system for laboratory mice using high-speed vision," *Journal of Real-Time Image Processing*, Vol. 4, No. 2, pp. 181–190, 2009.

[21] Q. Gu, T. Takaki and I. Ishii, 2000-fps multi-object extraction based on cell-based labeling, *Proc. IEEE Int. Conf. Image Process.*, pp. 3761–3764, 2010.

[22] F. Bernardini, and H. Rushmeier, " The 3D Model Acquisition Pipeline," *COMPUTER GRAPHICS forum*, Vol. 21, No. 2, pp. 149–172, 1999.

[23] E. Baltsavias, " Airborne laser scanning: basic relations and formulas," *ISPRS Journal of Photogrammetry & Remote Sensing*, Vol. 54, pp. 199–214, 1999.

[24] S. Seitz, " An overview of passive vision techniques," *SIGGRAPH 2000 Course on 3D Photography, Course Notes*, 1999.

[25] M. Muquit, T, Shibahara and T. Aoki, " a high-accurate passive 3D measurement system using phase-based image matching," *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*, Vol. E89-A, No. 3, pp. 586–697, 2006.

[26] O. Faugeras, " Three dimensional computer vision - A geometric viewpoint," *MIT Press*, 1993.

[27] D. Kobayashi, T. Takubo, and A. Ueno, "Model-Based Footstep Planning Method for Biped Walking on 3D Field," *Journal of Robotics and Mechatronics,* vol. 27, no. 2, pp. 156–166, 2015.

[28] S. Akizuki, and M. Hashimoto, "Stable Position and Pose Estimation of Industrial Parts Using Evaluation of Observability of 3D Vector Pairs," *Journal of Robotics and Mechatronics,* vol. 27, no. 2, pp. 174–181, 2015.

[29] J. Pages, J. Salvi, R. Garcia, and C. Matabosch, "Overview of coded light projection techniques for automatic 3D profiling," in *IEEE International Conference on Robotics and Automation,* 2003, pp. 133–138.

[30] S. Zhang, "Recent progresses on real-time 3D shape measurement using digital fringe projection techniques," *Optics and lasers in engineering,* vol. 48, no. 2, pp. 149–158, 2010.

[31] G. Geng, "Structured-light 3D surface imaging: a tutorial," *Advances in Optics and Photonics,* vol. 3, no. 2, pp. 128–160, 2011.

[32] M. Morita, T. Saito, S. Kurihara, and K. Kohiyama, "A Design and Implementation of 3D Scanning System Using a Slit Lighting," *Imaging & Visual Computing The Journal of the Institute of Image Electronics Engineers of Japan,* vol. 33, no. 4B, pp. 555–564, 2004.

[33] Microsoft, Xbox 360 Kinect, http://www.xbox.com/en-US/kinect.

[34] F. Forster, "A high-resolution and high accuracy real-time 3D sensor based on structured light," *Proceedings of the 3th International Symposium on 3D Data Processing, Visualization, and Transmission,* 2006, pp. 208–215.

[35] Q. Gu, T. Takaki, and I. Ishii, "Fast FPGA-based multi-object feature extraction," *IEEE Trans. Circuits and Systems for Video Technology,* vol. 23, no. 1, pp. 30–45, 2013.

[36] Q. Gu, T. Takaki, and I. Ishii, "A fast multi-object extraction algorithm based on cell-based connected components labeling," *IEICE Trans. Information and Systems,* vol. E95-D, no. 2, pp. 636–645, 2012.

[37] X. Su and Q. Zhang, "Dynamic 3-D shape measurement method: A review," *Optics and Lasers in Engineering,* vol. 48, no. 2, pp. 191–204, 2010.

[38] J. Davis, R. Ramamoothi, and S. Rusinkiewicz, "Spacetime stereo: A unifying framework for depth from triangulation," *IEEE Trans. On Pattern Analysis and Machine Intelligence (PAMI),* vol. 27, no. 2, pp. 296–302, 2005.

[39] S. Yamazaki, A. Nukada, M. Mochimaru, "Hamming Color Code forDense and Robust One-shot 3D Scanning," *The British Machine Vision Conference (BMVC),* 2011, pp. 96.1-96.9.

[40] A. Osman Ulusoy, F. Calakli, and G. Taubin, "One-shot scanning using De Bruijn spaced grids," *IEEE 12th International Conference on Computer Vision Workshops(ICCV Workshops),* 2009, pp. 1786–1792.

[41] H. Kawasaki, R. Furukawa, R. Sagawa, Y. Yagi, "Dynamic scene shape reconstruction using a single structured light pattern," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* 2008, pp. 1–8.

[42] R. Sagawa, H. Kawasaki, S. Kiyota and R. Furukawa, " Dense one-shot 3d reconstruction by detecting continuous regions with parallel line projection," *IEEE International Conference on Computer Vision (ICCV),* 2011, pp. 1911–1918.

[43] K. Nakazawa and C. Suzuki, "Development of 3-D robot vision sensor with fiber grating: Fusion of 2-D intensity image and discrete range image," in *International Conference on Industrial Electronics, Control and Instrumentation,* 1991, pp. 2368–2372.

[44] K. Umeda, "A compact range image sensor suitable for robots," in *IEEE International Conference on Robotics and Automation (ICRA),* 2004, pp. 3167–3172.

[45] Y. Watanabe, T. Komura, and M. Ishikawa, "955-fps real-time shape measurement of a moving/deforming object using high-speed vision for numerous-point analysis," in *IEEE International Conference on Robotics and Automation (ICRA),* 2007, pp. 3192–3197.

[46] S. Inokuchi, K. Sato, and F. Matsuda, "Range imaging system for 3-D object recognition," in *Proceedings of the International Conference on Pattern Recognition,* 1984, pp. 806–808.

[47] S. S. Gorthi, and P. Rastogi, "Fringe Projection Techniques: Whither we are?," *Optics and Lasers in Engiering,* vol. 48, no. 2, pp. 133–140, 2010.

[48] M. Kumagai, "Development of a 3D Vision Range Sensor Using Equiphase Light Section Method," *Journal of Robotics and Mechatronics,* vol. 17, no. 2, pp. 110–115, 2005.

[49] H. Gao, T. Takaki, and I. Ishii, "GPU-based real-time structure light 3D scanner at 500 fps," in *Proc. SPIE 8437 (SPIE Photonics Europe / Real-Time Image and Video Processing),* 2012, pp. 8437-18.

[50] J. Chen, T. Yamamoto, T. Aoyama, T. Takaki, and I. Ishii, "Simultaneous Projection Mapping Using High-frame-rate Depth Vision," in *IEEE International Conference on Robotics and Automation (ICRA),* 2014, pp. 4506–4511.

[51] Y. Liu, H. Gao, Q. Gu, T. Aoyama, T. Takaki, and I. Ishii, "High-Frame-Rate Structured Light 3-D Vision for Fast Moving Objects," *Journal of Robotics and Mechatronics,* vol. 26, no. 3, pp. 311–320, 2014.

[52] Y. Zhang, Z. Xiong, and F. Wu, "Unambiguous 3D measurement from speckle-embedded fringe," *Applied Optics,* vol. 52, no. 32, pp. 7797–7805, 2013.

[53] S. Zhang, D. Van Der Weide, and J. Oliver, "Superfast phase-shifting method for 3-D shape measurement," *Optics express,* vol. 18, no. 9, pp. 9684–9689, 2010.

[54] Y. Wang, K. Liu, Q. Hao, D. L. Lau, and L. G. Hassebrook, "Period coded phase shifting strategy for real-time 3-D structured light illumination," *IEEE Trans. Image Process,* vol. 20, no. 11, pp. 3001–3013, 2011.

[55] S. Zhang, X. Li, and S. Yau, "Multilevel quality-guided phase unwrapping algorithm for real-time three-dimensional shape reconstruction," *Applied Optics,* vol. 46, no. 1, pp. 50–57, 2007.

[56] L. Zhang, B. Curless, and S. Seitz, "Spacetime stereo: Shape recovery for dynamic scenes," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* 2003, pp. 367–374.

[57] O. Hall-Holt, S. Rusinkiewicz, "Stripe boundary codes for real-time structured-light range scanning of moving objects," *The Eighth IEEE International Conference on Computer Vision (ICCV),* 2001, pp. 359–366.

[58] L. Zhang, B. Curless, and S. Seitz, "Rapid Shape Acquisition Using Color Structured Light and Multi-pass Dynamic Programming," in *Proceedings of the 1st International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT),* 2002, pp. 24–36.

[59] T. Weise, B. Leibe, and L. Van Gool, "Fast 3d scanning with automatic motion compensation," *IEEE Conference on Computer Vision and Pattern Recognition,* 2007, pp. 1–8.

[60] I. Ishii, T. Tatebe, Q. Gu, Y. Moriue, T. Takaki, and K. Tajima, "2000 fps real-time vision system with high-frame-rate video recording," in *IEEE International Conference on Robotics and Automation (ICRA),* 2010, pp. 1536–1541.

[61] J. Pages, J. Salvi, R. Garcia and C. Matabosch, Overview of Coded Light Projection Techniques for Automatic 3D profiling, Prof. IEEE Int. Conf. Robot. Automat., 133–138 (2003).

[62] S. Zhang, Recent Progress on Real-time 3D Shape Measurement Using Digital Fringe Projection Techniques, Opt. Lasers Eng., 48(2), 149–158 (2010).

[63] J. Geng, Structured-light 3D Surface Imaging: A Tutorial, Adv. Opt. Photon., 3(2), 128–160 (2011)

[64] C. Guan, L. Hassebrook, and D. Lau, Composite Structured Light Pattern for Three-dimensional Video, Opt. Express, 11(5), 406–417 (2003)

[65] K. Sakashita, R. Sagawa, R. Furukawa, H. Kawasaki, and Y. Yagi, A system for capturing textured 3D shapes based on one-shot grid pattern with multi-band camera and infrared projector, Int. Conf. on 3D Imaging, Modeling, Processing, Visualization and Transmission, 49-56 (2011).

[66] Microsoft, xbox 360 kinect, http://www.xbox.com/en-us/kinect (2006)

[67] H. Gao, T. Takaki, and I. Ishii, GPU-based real-time structured light 3-D scanner at 500 fps, Proc. SPIE 8437, (SPIE Photonics Europe / Real-Time Image and Video Processing), 84370J (2012)

[68] J. Chen, T. Yamamoto, T. Aoyama, T. Takaki, and I. Ishii, Simultaneous Projection Mapping Using High-frame-rate Depth Vision, Proc. IEEE Int. Conf. on Robotics and Automation (2014).

[69] J. Chen, Q. Gu, Hao Gao, T. Aoyama, T. Takaki, and I. Ishii, Fast 3-D Shape Measurement Using Blink-Dot Projection, Proc. IEEE/RSJ Int. Conf. on Intelli. Robot. and Syst., 2683–2688 (2013).

[70] S. Inokuchi, K. sato, and F. Matsuda, Range imaging system for 3-D object recognition, Proc. Int. Conf. Pat. Recog. 806–808 (1984).

[71] D. Moreno and G. Taubin, Simple, Accurate and Robust Projector-Camera Calibration. 2012 Second Int. Conf. on 3D Imaging, Modeling, Processing, Visualization and Transmission, 464–471 (2012).

[72] I. Ishii, T. Tatebe, Q. Gu, Y. Moriue, T. Takaki, and K. Tajima, 2000 fps Real-time Vision System with High-frame-rate Video Recording, Proc. IEEE Int. Conf.. Robot. Automat., 1536–1541 (2010).

[73] R.T. Azuma, A survey of augmented reality, *Teleoperators and Virtual Environments* Vol. 6, No. 4, pp. 355–385, 1997.

[74] X. Zhang, S. Fronz, and N. Navab, Visual marker detection and decoding in AR systems: a comparative study, *Proceedings of the 1st International Symposium on Mixed and Augmented Reality*, pp. 97–106, 2002.

[75] O. Bimber and R. Raskar, *Spatial augmented reality, merging real and virtual worlds*, AK Peters, 2005.

[76] M. Mine, D. Rose, B. Yang, J. van Baar, and A. Grundhofer, Projection-based augmented reality in Disney Theme Parks, *IEEE Computer*, Vol. 45, No. 7, pp. 32–40, 2012.

[77] T. Sueishi, H. Oku, and . Ishikawa, Robust high-speed tracking against illumination changes for dynamic projection mapping, *IEEE Virtual Reality Conference*, pp. 97–104, 2015.

[78] Y. Yasumuro, M. Imura, Y. Manabe, O. Oshiro, and K. Chihara, Projection-based augmented reality with automated shape scanning, *International Society for Optics and Photonics*, pp. 555–562, 2005.

[79] M. Flagg and J.M. Rehg, Projector-guided painting, *Proceedings of the 19th annual ACM Symposium on User Interface Software and Technology*, pp. 235–244, 2006.

[80] B. Jones, H. Benko, E. Ofek, and A. Wilson, IllumiRoom: peripheral projected illusions for interactive experiences, *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 869-878, 2013.

[81] R. Raskar, and K. Low, Interacting with spatially augmented reality, */Proceedings of the 1st international Conference on Computer Graphics, Virtual Reality and Visualisation*, pp. 101-108, 2001.

[82] H. Yoo, and H. Kim, On study of the projection mapping in media arts, *Advanced Science and Technology Letters*, Vol. 54, pp. 73-76 , 2014.

[83] N. Sakata, T. Konishi, ans S. Nishida, Mobile interfaces using body worn projector and camera, *Proceedings of the 3rd International Conference on Virtual and Mixed Reality: Held as Part of HCI International 2009*, pp. 106–113, 2009.

[84] O. Bimber, B. Frohlich, D. Schmalstieg, and L.M. Encarnacao, The virtual showcase, *IEEE Computer Graphics and Applications*, Vol. 21, No. 6, pp. 48–55, 2001.

[85] S. Chon, H. Lee, and J. Yoon, 3D architectural projection, Light Wall, *Leonardo*, Vol. 44, No. 2, pp. 172–173, 2011.

[86] J. Lee, Y. Kim, M. Heo, et al, Real-time projection-based augmented reality system for dynamic objects in the performing arts, *Symmetry*, Vol. 7, No. 1, pp. 182–192, 2015..

[87] A. Wilson, H. Benko, S. Izadi, and O. Hilliges, Steerable augmented reality with the beamatron, *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*, pp. 413–422, 2012.

[88] Microsoft, Xbox 360 Kinect, http://www.xbox.com/en-US/kinect

[89] R. Krempien, H. Hoppe, L. Kahrs, et al, Projector-based augmented reality for intuitive intraoperative guidance in image-guided 3d interstitial brachytherapy, *International Journal of Radiation Oncology Biology Physics*, Vol. 70, No. 3, pp. 944–952, 2008.

[90] S. Sakamaki, and N. Hashimoto, Time-delay compensation for dynamic projection mapping *SIGGRAPH Asia 2013 Posters. ACM*, pp. 39, 2013.

[91] H. Nagai, Y. Mukaigawa, ans Y. Ohta, Screen object tracking for projector-based mixed reality, *IEICE Technical Report*, Vol. 104, No. 391, pp. 1–6, 2004.

[92] K. Okumura, H. Oku, and M. Ishikawa, Lumipen, projection-based mixed reality for dynamic objects, *IEEE International Conference on Multimedia and Expo*, pp. 699–704, 2012.

[93] K. Okumura, H. Oku, and M. Ishikawa, High-Speed gaze controller for millisecond-order pan/tilt camera, *IEEE International Conference on Robotics and Automation*, pp. 699–704, 2012.

[94] I. Ishii, T. Tatebe, Q. Gu, Y. Moriue, T. Takaki, and K. Tajima, 2000 fps real-time vision system with high-frame-rate video recording, *IEEE International Conference on Robotics and Automation*, pp. 1536–1541, 2010.

[95] S. Inokuchi, K. Sato, and F. Matsuda, Range Imaging System for 3-D Object Recognition, *Proc. Int. Conf. Patt. Recog.*, pp. 806–808, 1984.

[96] Y. Liu, H. Gao, Q. Gu, T. Aoyama, T. Takaki, and I. Ishii, High-frame-rate structured light 3-D vision for fast moving objects, *Journal of Robotics and Mechatronics*, Vol.26 No.3, pp. 311–320, 2014.

[97] S. Zhang, Recent progresses on real-time 3D shape measurement using digital fringe projection techniques, *Optics and lasers in engineering*, Vol.48 No.2, pp. 149–158, 2010.

[98] G. Geng, Structured-light 3D surface imaging: a tutorial, *Advances in Optics and Photonics*, Vol.3 No.2, pp. 128–160, 2011.

[99] J. Pages and J. Salvi and R. Garcia and C. Matabosch, Overview of coded light projection techniques for automatic 3D profiling, *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 133–138, 2003.

[100] S. M. Seitz and B. Curless and J. Diebel and D. Scharstein and R. Szeliski, A comparsion and evaluation of multi-view stereo reconsturction algorithms, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 519–528, 2006.

[101] X. Su and Q. Zhang, Dynamic 3-D shape measurement method: A review, *Optics and Lasers in Engineering*, Vol.48 No.2, pp. 191–204, 2010.

[102] H. Gao and T. Takaki and I. Ishii, GPU-based real-time structure light 3D scanner at 500 fps, *Proc. SPIE 8437 (SPIE Photonics Europe / Real-Time Image and Video Processing)*, pp. 8437–8442, 2012.

[103] S. Inokuchi and K. Sato and F. Matsuda, Range imaging system for 3-D object recognition, *Proceedings of the International Conference on Pattern Recognition (ICPR)*, pp. 806–808, 1984.

[104] J. Chen and T. Yamamoto and T. Aoyama and T. Takaki and I. Ishii, Simultaneous Projection Mapping Using High-frame-rate Depth Vision *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4506–4511, 2014.

[105] Y. Liu and H. Gao and Q. Gu and T. Aoyama and T. Takaki and I. Ishii, High-Frame-Rate Structured Light 3-D Vision for Fast Moving Objects, *Journal of Robotics and Mechatronics*, Vol.26 No.3, pp. 311–320, 2014.

[106] S. S. Gorthi and P. Rastogi, Fringe Projection Techniques: Whither we are?, *Optics and Lasers in Engiering*, Vol.48 No.2, pp. 133–140, 2010.

[107] T. Weise and B. Leibe and L. Van Gool, Fast 3D scanning with automatic motion compensation, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–8, 2007.

[108] S. Zhang and X. Li and S. Yau, Multilevel quality-guided phase unwrapping algorithm for real-time three-dimensional shape reconstruction, *Applied Optics*, Vol.46 No.1, pp. 50–57, 2007.

[109] S. Zhang and X. Li and S. Yau, Multilevel quality-guided phase unwrapping algorithm for real-time three-dimensional shape reconstruction, *Applied Optics*, Vol.46 No.1, pp. 50–57, 2007.

[110] S. Zhang and X. Li and S. Yau, Unambiguous 3D measurement from speckle-embedded fringe, *Applied Optics*, Vol.52 No.32, pp. 7797–7805, 2013.

[111] R. Furukawa and R. Sagawa and H. Kawasaki and K. Sakashita and Y. Yagi and N. Asada, One-shot entire shape acquisition method using multiple projectors and cameras, *Proceeding of the Pacific-Rim Symposium Image and Video Technology (PSIVT)*, pp. 107–114, 2010.

[112] H. Kawasaki and R. Furukawa and R. Sagawa and Y. Yagi, Dynamic scene shape reconstruction using a single structured light pattern, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–8, 2008.

[113] S. Yamazaki and A. Nukada and M. Mochimaru, Hamming Color Code for-Dense and Robust One-shot 3D Scanning, *The British Machine Vision Conference (BMVC))*, pp. 96.1–96.9, 2011.

[114] S. Yamazaki and A. Nukada and M. Mochimaru, A high-resolution and high accuracy real-time 3D sensor based on structured light, *Proceedings of the 3th International Symposium on 3D Data Processing, Visualization, and Transmission*, pp. 208–215, 2006.

[115] K. Umeda, A compact range image sensor suitable for robots, *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3167–3172, 2004.

[116] R. Sagawa and H. Kawasaki and S. Kiyota and R. Furukawa, Dense one-shot 3D reconstruction by detecting continuous regions with parallel line projection, *IEEE International Conference on Computer Vision (ICCV)*, pp. 1911–1918, 2011.

[117] I. Ishii and T. Tatebe and Q. Gu and Y. Moriue and T. Takaki and K. Tajima, 2000 fps real-time vision system with high-frame-rate video recording, *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1536–1541, 2010.

[118] Q. Gu and T. Aoyama and T. Takaki and I. Ishii, Simultaneous Vision-Based Shape and Motion Analysis of Cells Fast-Flowing in a Microchannel, *IEEE Transactions on Automation Science and Engineering*, Vol.12 No.1, pp. 204–215, 2013.

Acknowledgment

I wish to thank my advisor, Professor Idaku Ishii, for his advice, suggestions, encouragement and patience. He has taught me in the various research fields of high-speed vision, vision based 3-D shape measurement, shape inspection. His serious attitude toward research has been an excellent example to me.

I would like to acknowledge, associate Professor Takeshi Takaki, Designated Associate Professor Qingyi Gu, and assistant Professor Tadayoshi Aoyama for theirs help in the selection of experimental materials and experimental setup, and theirs invaluable suggestions, discussions and comments on this study.

I would also like to Ms. Yukari Kaneyuki, Ms. Rumi Horiguchi, and Ms. Kazuko Yoshimura secretaries in Robotics Laboratory, who help the author in many ways, such as purchasing of equipments and delivering various procedures.

I would also like to express my sincere thanks to the bachelor, master and doctoral students in Robotics Laboratory for their help in life and my research.

Finally, I express my deepest gratitude to my family, who have done everything possible for me even in hard times. Without their support, I could never have reached this point.

June, 2015

Jun Chen