

Embedding Invisible Codes into Normal Video Projection: Principle, Evaluation, and Applications

Jingwen Dai, *Member, IEEE*, and Chi-Kit Ronald Chung, *Senior Member, IEEE*

Abstract—We describe a system of embedding codes into projection display for structured light-based sensing, with the purpose of letting the projector serve as both a display device and a 3-D sensor. The challenge is to make the codes imperceptible to human eyes so as not to disrupt the content of the original projection. There is the temporal resolution limit of human vision that one can exploit, by having a higher than necessary frame rate in the projection and stealing some frames for code projection. Yet, there is still the conflict between imperceptibility of the embedded codes and the robustness of code retrieval that has to be addressed. We introduce noise-tolerant schemes to both coding and decoding stages. At the coding end, specifically designed primitive shapes and large Hamming distance are employed to enhance tolerance toward noise. At the decoding end, pretrained primitive shape detectors are used to detect and identify the embedded codes, a task that is difficult to achieve by segmentation that is used in general structured light methods, because the weakly embedded information is generally interfered by substantial noise. Extensive experiments show that the proposed system is effective, even with the prerequisite of incurring minimum disturbance to the original projection.

Index Terms—Embedded pattern design, imperceptible structured light sensing, primitive shape detection and classification, sensitivity analysis.

I. INTRODUCTION

THE improving performance, declining price, and diminishing size of digital video projectors make it possible to use them prevalently. Being able to generate an arbitrarily large display is a feature of projectors that makes them exceedingly attractive, especially in applications that demand portability. On the other hand, the adoption of structured light illumination has been proven to be an effective and accurate means for 3-D information perception [1]. Recently, the availability of pico projectors with average dimensions of $4 \times 2 \times 1$ inches has widely extended the application domain of structured light systems. There are already pocket DCs, DVs and cellular phones (as shown in Fig. 1) in the consumable market that have both projector and camera built-in, making it possible

Manuscript received November 1, 2012; revised January 20, 2013 and March 26, 2013; accepted April 9, 2013. Date of publication June 20, 2013; date of current version December 3, 2013. This paper was recommended by Associate Editor P. Yin.

J. Dai is with the Department of Computer Science, University of North Carolina at Chapel Hill, Chapel Hill, NC 27599-3175 USA (e-mail: dai@cs.unc.edu).

C.-K. R. Chung is with Vocational Training Council of Hong Kong, Wan Chai, Hong Kong (e-mail: rchung@vtc.edu.hk).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCSVT.2013.2268993



Fig. 1. Mobile devices with built-in pico projector.

to implement structured light system in hand-held consumer electronic products.

In other words, a projector accompanied by a camera has the potential of achieving both display and sensing, i.e., for both input and output in human-computer interface, making it a possible device to replace traditional LCD panel, keyboard, and touch-sensitive screen altogether in computing, with only diminished size and weight. A projector has the potential of making a breakthrough of dramatically downsizing portable computing without sacrificing display size.

For these reasons, a projector-camera (ProCam) system has been actively researched in the last few years. Many research groups apply projectors in unconventional ways to develop new and innovative information displays that go beyond simple screen presentations [2].

Some researchers designed structured light systems in the non-visible spectrum [3]. In that way the media for regular projection and structured light-based sensing (SLS) can be made separately. However, additional hardware could be reduced and device size diminished if structured light and regular projection can be achieved through the same projector. This leads to the concept of Imperceptible Structured Light (ISL). ISL modulates the projected display either spatially or temporally to embed code patterns for SLS. In principle, due to limitations of human visual perception, the embedded code patterns can be made undetectable to the user, but cameras synchronized to the modulation are able to reconstruct the embedded codes for SLS.

The embedding of code patterns into regular projection can be used for a variety of applications including projector calibration, camera tracking, and 3-D scanning.

There is however a challenge in embedding codes into regular projection. While the codes should be made as undetectable as possible to the user, they have to be decodable to the camera for the purpose of SLS. On top of the dilemma, there is the inevitable fact that the displayed signals are generally corrupted by substantial noise that arises from the nonlinearity of the projector, the sensing defects of the camera, and the variation of the ambient illumination. The objective of this paper is to deal with the dilemma and the accompanying issues.

This article describes a novel method of embedding imperceptible structured codes into arbitrarily intended projection. Through precise projector-camera synchronization, structured codes consisting of three primitive shapes are embedded into the projection, in a way that is imperceptible to viewers but extractable from the "difference image" between successive images captured by a camera. To make the decoding process more robust against noise, we do not extract the codes by region segmentation in the image domain. Instead we employ specially trained classifiers to detect and identify the codes. To enhance the error tolerance further, specially designed primitive shapes and large Hamming distance are adopted in the spatial coding. Even with some bits of the codewords missed or wrongly coded, the correct correspondence could still be derived correctly.

A preliminary version of this paper was published in [4], in which we introduced the robust coding and decoding approach for embedding imperceptible structured code into arbitrary video projection. In [5], we integrated the method into robotic system and demonstrated the effectiveness in robotic applications. In training-based methods, generalization of the training results is often an issue, and it is especially so when the work scenario could have substantial variation between the training stage and the operation stage. In this paper, we provide a thorough experimental sensitivity evaluation together with the associated statistical analysis, they show that the scheme of embedding imperceptible codes into normal video projection can be maintained effective despite possible variations on sensing distance, projection-surface orientation, projection-surface shape, projection-surface texture and hardware configuration.

The remainder of this paper is structured as follows. In Section II, related works on imperceptible structured light sensing are briefly reviewed. The principle of embedding imperceptible codes along with robust coding and a noise-tolerant decoding mechanism are described in Section III. In Section IV, system setup and experimental results are shown. Sensitivity evaluation and potential applications are described in Sections V and VI. Conclusion and possible future work are offered in Section VII.

II. RELATED WORK

A proof of concept for embedding invisible structured light patterns into digital light processing (DLP) projections first appeared in the "Office of the Future" project [6]. In this paper, binary codes are embedded by projecting temporally alternating code images and their complements. Provided that the frequency of projection reaches the flicker fusion

threshold ($\geq 75\text{Hz}$), the pattern and the inverse pattern are visually integrated over time in human perception, and the illumination has the appearance of a flat field (white light) to humans. However, the demonstration required significant modification effort on the projection hardware and firmware, including removal of the color wheel and reprogramming of the controller. The resulting images were also in greyscale only. The implementation of such a setting was impossible without mastering and full access to the projection hardware.

Cotting *et al.* [7] introduced a coding scheme that synchronizes a camera to a specific time slot of a DLP micro-mirror flipping sequence in which imperceptible binary patterns are embedded. However, not all mirror states are available for all possible intensities, and the additional hardware, DVI repeater with tapped vertical sync signal, is not an off-the-shelf instrument.

However, with the development of digital projection technology, some so-called 3-D compatible DLP projectors with refresh rate of 120Hz or higher emerged recently. This makes it possible to implement imperceptible structured light without any hardware modification or extra assisting hardware. Many researcher began to study how to determine the embedded intensity properly to guarantee code imperceptibility.

In [8], subjective evaluation results and statistical analysis on the visual perceptibility of embedded codes in different ways were reported. The factors affecting code visibility are also outlined. Park *et al.* [9] presented a technology for adaptively adjusting the intensity of the embedded code with the goal of minimizing its visibility. It was regionally adapted depending on the spatial variation of neighboring pixels and their color distribution in the YIQ color space. The final code intensity was then weighted by the estimated local spatial variation. Since two manually defined parameters adjusted the overall strength of the integrated code, the system was not able to automatically calculate an optimized intensity. Grundhofer *et al.* [10] proposed a method considering the capabilities and limitations of human visual perception for embedding codes. It estimated the just noticeable differences (JND) based on the human contrast sensitivity function and adapted the code intensity on the fly through regional properties of the projected image and code, such as luminance and spatial frequencies. The shortcoming of this method was that some parameters need be premeasured using some optical devices (e.g. photometer), which were not accessible to nonprofessional users.

To the best of our knowledge, until now, few works focus on the decoding method in imperceptible code embedding configuration, especially when huge external noise could exist.

III. METHOD

A. Principle of Embedding Imperceptible Codes

The fundamental principle behind imperceptible structured code embedding is the temporal integration achieved by projecting each image twice at high frequency: a first image containing actual code information (e.g., by adding or subtracting a certain amount (Δ) to or from the pixels of the original image, depending upon the code) and a second image that compensates for the distortion in the first image. The vital

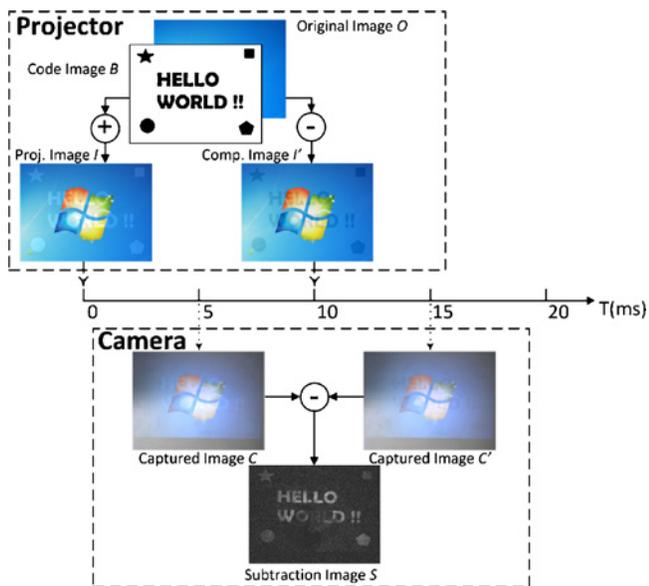


Fig. 2. Projector-camera synchronization and basic principle of embedding and extracting imperceptible codes.

aspects of ISL sensing are code embedding and projector-camera synchronization.

Since general projection is in color, it is possible to embed color code through three different channels. However, to enhance code robustness toward noise, we use binary code and embed it into all three color channels simultaneously. Let B , O , I , and I' be the binary code image, the original image, the projected image, and the complementary image, respectively. Then the projected image and complementary image could be formulated as

$$I_i(x, y) = O_i(x, y) + P(x, y) \quad (1)$$

$$I'_i(x, y) = O_i(x, y) - P(x, y) \quad (2)$$

$$P(x, y) = \begin{cases} \Delta, & \text{when } B(x, y) = 1 \\ 0, & \text{when } B(x, y) = 0 \end{cases} \quad (3)$$

where $i = \{R, G, B\}$ indicates red, green and blue channels, Δ is the embedded intensity.

To avoid intensity saturation at lower and higher intensity levels when adding or subtracting Δ , the original image needs to have the intensity range in each color channel compressed to between Δ to $255 - \Delta$. Since the embedded intensity required in the coding is small enough, the visual degradation due to contrast reduction is negligible.

The degree of imperceptibility thus depends upon the embedded intensity. A larger intensity enables the code to be more tolerant toward noise and more readable in the image of the projection, whilst a smaller intensity makes the embedded codes more invisible. In our design, code imperceptibility has higher priority, and thus embedded intensity is set to a very small value.

In order to achieve imperceptible structured light projection, the frequency of projection must exceed the flicker fusion threshold, which is 75Hz for most of the people. Here we take one projection-capture cycle as an example to elaborate

the strategy of projector-camera synchronization, which is illustrated in Fig. 2. Firstly, we ensure that the projector projects an image every 10ms , i.e., at 100Hz . As shown in Fig. 2, along the time axis, the projected image I and the complementary image I' are projected at the time instants 0ms , 10ms , respectively. With a refresh rate of the camera at about 100 frames per second, the camera captures the image C and C' at 5ms and 15ms , shortly after the projector projects the projected image and complementary image to the scene. At 20ms a new projection-capture cycle will resume. With the aforementioned projection-capture strategy, the system could capture 50 image pairs per second.

The embedded codes could be internally and simply extracted from the "subtraction image"¹ between consecutively captured images as

$$S(x, y) = \max_i [C_i(x, y) - C'_i(x, y)], \quad i = \{R, G, B\}. \quad (4)$$

Ideally, the subtraction image should be a binary image that has maximum value of 2Δ and minimum value of 0. However, the subtraction image in reality is generally disturbed by large external noises. Since the embedded intensity is always small, the subtraction image has low signal-to-noise ratio. It is generally nontrivial to retrieve the embedded codes. In the rest of this section, we describe how robust coding and noise-tolerant decoding approaches can help tackle the issue.

B. Design of Embedded Pattern

The strategy of encoding in general structured light methods could be classified into two categories [1]: time multiplexing and spatial multiplexing. The former can achieve denser data samples with higher accuracy, but at the expense of requiring multiple illuminations and image captures over time, which is not suitable for imperceptible code embedding [8] and dynamic scenes. In contrast, the latter labels each pattern position by the appearance profile (color, shape or their combination) of the neighboring positions. The appearance profile can be about various gray levels, colors, or geometric primitives, and the coding methods include De-Bruijn sequences [11]–[13], M-arrays [14]–[17], and non-formal coding [18]–[21]. The spacial coding scheme has the advantage that 3-D determination could be achieved with a single pattern.

Considering the constraints of imperceptible code embedding, we employ the spatial multiplex scheme to design our pattern. Due to the choice of using binary code for robust code embedding, the symbols cannot be coded with different colors. We thus use an alphabet set comprising three different geometrical primitives: cross, sandglass, and rhombus, as shown in Fig. 3. There are three advantages of this configuration. First, all the shapes own a natural center point, which simplifies the shape identification process in the decoding stage. Then, there are sufficient variations between different shapes; even with large disturbance from noise on the shapes, the decoding method could discriminate them. Moreover, the directional information carried by the cross

¹All the subtraction images in this article are scaled to $[0, 255]$ for illustration purpose.



Fig. 3. Primitive shapes. (a) Cross, (b) sandglass, and (c) rhombus.

shape could rectify the observation window in the step of neighborhood detection without the need of enforcing any other constraint.

In the decoding stage, the centroid of each detected primitive would be considered as the feature point position, and the 9-bit codeword associated to each feature point is composed of the elements in the 3×3 window centered on it. In traditional structured light methods, the uniqueness of the codeword is usually assured by M -arrays (perfect maps), which are random arrays of dimensions $r \times v$ in which a submatrix of dimensions $n \times m$ appears only once in the whole pattern [14]. The M -arrays give a total of $rv = 2^{nm} - 1$ unique submatrices in the pattern and a window property of $n \times m$. However, the Hamming distance between the codewords is 1, which is generally too small for our code embedding scenario in which the codeword retrieval errors could be large due to noise. To increase this value, we employ the method proposed in [22] to impose an additional constraint so that each window must remain unique even if the upper corners elements are missing. Respecting the above mentioned constraints, we generate the coding matrix through a brute-force approach: First, a submatrix of 3×3 is chosen randomly and is placed in the north-west vertex of the coding matrix that is being built. Then consecutive random columns of 1×3 are added to the right of this initial submatrix, fulfilling the constraints. Afterward, rows of 3×1 are added beneath the initial submatrix in a similar way. Then, both horizontal and vertical processes are repeated by incrementing the starting coordinates by one, until the whole coding matrix is filled. Whenever the process reaches a state where no possible elements can be placed, the array is cleared and the algorithm starts again with another initial submatrix. In the end, a matrix of dimensions 27×29 is generated, in which 95.97% of the codewords have a Hamming distance higher than 3 and the average Hamming distance is $\bar{H} = 6.0084$, so that even some bits in the codeword are missed or incorrectly coded, the codeword is still distinguishable. On the basis of this matrix, the binary code image composed of the primitive shapes appears like the one illustrated in Fig. 4, in which the size of each primitive shape is a collection of 11×11 pixels while the interval between each shape is 11 pixels. The total number of feature points is 783.

C. Primitive Shape Identification and Decoding

In the decoding stage, the existence of intense noises (from projector projection, camera sensing, ambient illumination and object surface reflection influence) makes it impossible to segment the primitive shape by the integrated use of region segmentation and edge or contour detection as often employed in ordinary structured light methods. Here, we regard the primitive shapes as objects to identify and detect rather than segment.

Compared with other object identification or recognition methods, the machine learning approach proposed by

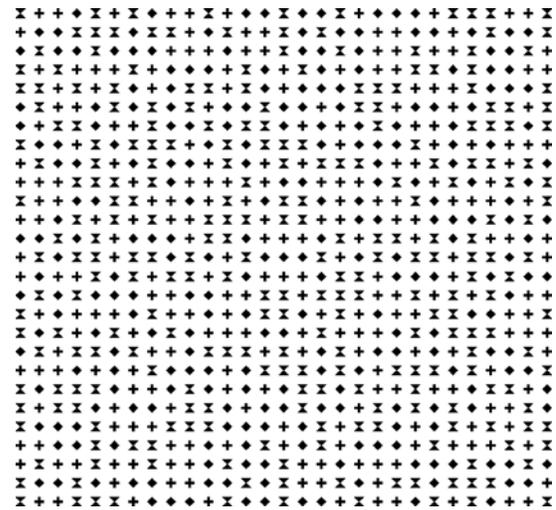


Fig. 4. Embedded binary code image.

P. Viola [23] has been shown to be capable of processing images rapidly with high detection rates for visual object detection. The approach is adopted here for training detector to identify the three primitive shapes. Below we use cross shape as an example to describe the procedure of detector training.

The performance of training-based detector has a great deal to do with the availability of training samples. Unlike generic objects like human face, body or vehicle, which have a large number of samples in a great many of public databases, we have to collect the specific training samples ourselves in the required configuration. 500 color images with different contents were collected from Google Image [24], and 40 cross shapes were embedded in those images at different positions to generate 500 pairs of projected images and complementary images.

A white planar projection screen was placed in front of the projector-camera system with the distance of $800mm$, the orientation of the screen was adjusted to make the projection area appear as a rectangle, i.e. the projection screen was parallel to the projection plane of the projector. By projecting the images, 500 subtraction images could be derived from image capture exercises. The subimages containing cross shapes were then segmented by manual labeling, which were considered as positive training samples. The background images with holes filled by random noise were divided into small patches to generate negative training samples. The training sample preparation process is shown in Fig. 5.

To obtain the optimal performance, the positive samples were resized to 20×20 , the extended haar-like features and Gentle Adaboost algorithm were employed, following the suggestion in [25]. Eventually, from over 7000 positive samples and 3000 negative samples, a 16-stage cascade classifier for cross detection was trained. Following the same procedure, the detectors for sandglass and rhombus shapes could be derived as well.

D. Codeword Retrieval

By using the pretrained primitive shape detectors, the centroid of each primitive, i.e., the position of each feature point,

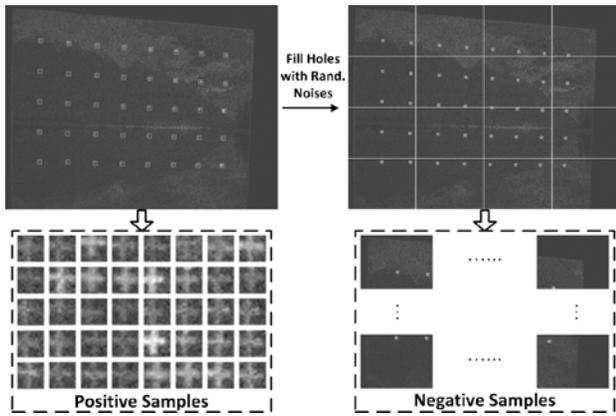


Fig. 5. Training sample preparation.

can be determined. Once a feature point is extracted from the image, its codeword can be produced from the associated 3×3 intensity window centered on the feature point. Let c_i be the code of point P_i , P_0 can be encoded in a fixed order say the order of c_0 - c_1 - c_2 - c_3 - c_4 - c_5 - c_6 - c_7 - c_8 as shown in Fig. 6(a). Hence, the codeword of P_0 is calculated as

$$CW(P_0) = \sum_{i=0}^8 10^{(8-i)} \times c_i. \quad (5)$$

It is time-consuming and inefficient for searching the primitive shapes in the whole image, so in pattern design stage, we choose the cross as one of the three primitive shapes, since the directional information embraced will speed up the primitive shapes searching. In decoding stage, if one cross shape is detected correctly by the pretrained detector, it implies that the appearance of this cross shape is preserved well. For this reason, we consider that this embedded cross shape is projected in one of the piece-wise smooth regions on the surface and angle between the average surface normal of this region and the optical axis of the projector (and the camera, since the projector-camera system is an approximate parallel-axes configuration) is an acute angle less than 60° . So in this region the distortion of the embedded patterns in subtraction image is not great enough to change the spatial relationship between neighborhood shapes. Hence, the directional information embraced in the cross shape could be adopted to rectify the search window around it to find the other two shapes. As illustrated Fig. 6 (a), the cross shapes are detected first, then two directions are fitted through the intensity distributions in the detected rectangle, and in the end, rhombus and sandglass shapes are detected in the nearby area along the two directions. The corresponding point on the projector image plane is known a priori.

Once the correspondence problem is solved, the depth z_c of the associated scene point can be determined via traditional triangulation algorithm [26] as

$$z_c = \frac{(\mathbf{R}\tilde{\mathbf{m}}_c \cdot \tilde{\mathbf{m}}_p)(\tilde{\mathbf{m}}_p \cdot \mathbf{T}) - \|\tilde{\mathbf{m}}_p\|^2(\mathbf{R}\tilde{\mathbf{m}}_c \cdot \mathbf{T})}{\|\mathbf{R}\tilde{\mathbf{m}}_c\|^2\|\tilde{\mathbf{m}}_p\|^2 - (\mathbf{R}\tilde{\mathbf{m}}_c \cdot \tilde{\mathbf{m}}_p)^2} \quad (6)$$

where $\tilde{\mathbf{m}}_c$ and $\tilde{\mathbf{m}}_p$ refer to the two corresponding points on the camera's and projector's image planes, respectively, in

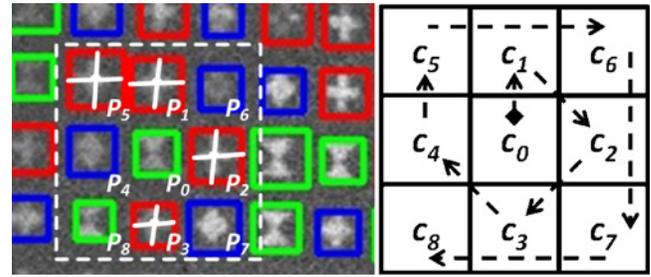


Fig. 6. Codeword retrieval: (a) one example and (b) coding order.

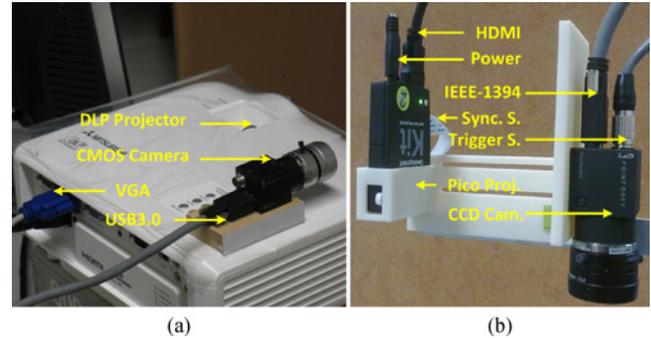


Fig. 7. Hardware configuration of two projector-camera systems. (a) PROCAMS-A and (b) PROCAMS-B.

homogeneous coordinates, whilst \mathbf{R} and \mathbf{T} are the rotation matrix and translation vector between the camera and projector coordinate system, respectively. The above is the 3-D sensing step we use in the system.

IV. EXPERIMENTS

A. Overview of Experiment Setup

To assess the feasibility of the proposed method for embedding imperceptible codes in regular projection, we conducted experiments on embedded code imperceptibility evaluation, primitive shape detector accuracy evaluation and primitive shape detector sensitivity evaluation.

In order to evaluate the performance of our method in different platforms, we set up two projector-camera systems using different equipment. The first one (*PROCAMS-A*) consisted of a consumer-level DLP projector (Mitsubishi EX240U projector) of 1024×768 resolution and 120Hz refresh rate, and a CMOS camera (Point Grey Flea 3 FL3-U3-13S2C with Myutron FV1520 $f15\text{mm}$ lens) of 1328×1048 resolution and 120fps , while the second one (*PROCAMS-B*) consisted of a Pico DLP projector with a native resolution of 640×480 and an interface for firmware configuration (TI DLP Pico Projector Development Kit 2 [27]), plus a CCD camera of 648×488 resolution at 120fps (Point Grey FL3-FW-03S1C camera with Myutron FV0622 $f6\text{mm}$ lens).

For *PROCAMS-A*, we first fixed the camera and projector rigidly, and the projector and camera were connected to a desktop computer through VGA and USB3.0 interfaces, respectively. Since there was no synchronization signal output in the consumer-level projector, the synchronization between projectors and cameras was implemented through software

delay. The hardware configuration is shown in Fig. 7(a). For *PROCAMS-B*, the projector and camera were mounted on a special designed framework rigidly, and were connected to a laptop computer through HDMI and IEEE-1394 interfaces, respectively, and the hardware trigger signal of the camera was connected to the sync. output of the projector for synchronization between them, which are illustrated in Fig. 7(b).

Moreover, the projector-camera systems were calibrated using an LCD monitor as the calibration object; the calibration method, detailed in [28], could derive the intrinsic and extrinsic parameters of the two instruments. Once the experimental system was set up and calibrated, we could conduct further experiments.

B. Embedded Code Imperceptibility Evaluation

Embedded code imperceptibility and user satisfaction are of the first priority in the system design. The imperceptibility depends on the embedded intensity. We conducted a subjective evaluation using *PROCAMS-A* based on a questionnaire. Ten persons were invited to participate in this experiment, of which six were male and four were female, and seven wearing glasses. Another 500 images were collected from Google Image [24] randomly, the content of the images included natural scene, portrait, architecture, animals and so on. Our proposed pattern was embedded into all the collected images with different intensities. The viewers were seated in front of a white planar screen at a distance of about 1m, and asked to comment on the images projected to the screen. The questions asked were simplified from the questionnaire in [8], focusing on the feeling of flickering, the recognition of image deterioration, and the overall satisfaction for projection quality. The score for each question was divided into ten levels.

The average scores of the subjective evaluation are illustrated in Fig. 8. When the embedded intensity is small, i.e., $\Delta = 5, 10$, the viewer could rarely notice the embedded codes and were satisfied with the projection quality. With the increase of the embedded intensity, the viewers' sense of flickering and image degradation became stronger. When $\Delta = 25$, almost every viewer was not satisfied with the projection quality.

In practice, because it was difficult to retrieve weakly embedded codes with the standard commercial cameras, we choose $\Delta = 10$ in our configuration, striking a compromise between user satisfaction and code imperceptibility.

C. Primitive Shape Detection Accuracy Evaluation

After embedded code imperceptibility evaluation, the experiments for primitive shape detection accuracy were carried out. Considering the training data for primitive shape detector training was collected by *PROCAMS-A*, we first evaluated the primitive shape detection accuracy on *PROCAMS-A*.

To assess accuracy, the experimental data with ground-truth were required. Three different primitives and the spatially coded pattern image were embedded into 500 images used for imperceptibility evaluation, respectively, with intensity $\Delta = 10$. Then the projected and complementary images were projected successively to a projection surface, while

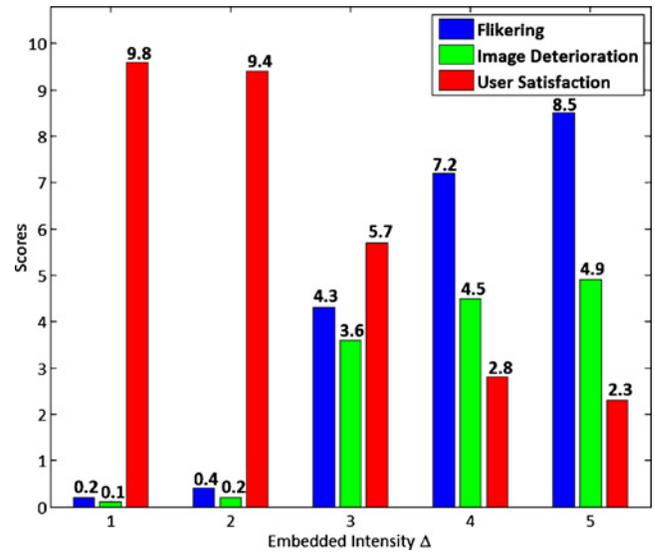


Fig. 8. Subjective evaluation results for code imperceptibility.

the camera conducted synchronized capture. The projection surface was the same as the one used for training data collection. Then the subtraction images embracing embedded codes information were derived for accuracy evaluation. The ground-truth was obtained by manual labeling in the image data captured under binary pattern illumination.

Experimental results in some subtraction images are presented in Fig. 9. The four subfigures display the cross [Fig. 9(a)], sandglass [Fig. 9(b)], rhombus [Fig. 9(c)] shapes, and the spatially coded pattern [Fig. 9(d)], respectively. For qualitative evaluation, the detected features are indicated by rectangles, and in the bottom-right subfigure, the cross, sandglass and rhombus shapes are separately marked by red, green, and blue rectangles. As described in [6], when the system is calibrated, the precision of 3-D reconstruction is determined by the correct correspondences establishment and the accuracy of corresponding points. Moreover, the correspondence is established by the detection and identification of the embedded primitive shapes. In the end, the performance of the system depends on the accuracy of primitive detectors, which are evaluated by hit rate (H), missing rate (M), false rate (F) and position error (E_d), which are formulated as

$$H = \frac{N_h}{N_t} \quad (7)$$

$$M = \frac{N_m}{N_t} \quad (8)$$

$$F = \frac{N_f}{N_t} \quad (9)$$

$$E_d = \sqrt{\epsilon_X^2 + \epsilon_Y^2} \quad (10)$$

$$\epsilon_X = \frac{1}{N_h} \sum_{i=1}^N |X_d - X_g|_i \quad (11)$$

$$\epsilon_Y = \frac{1}{N_h} \sum_{i=1}^N |Y_d - Y_g|_i \quad (12)$$

where N_t is the total embedded primitive shape number, N_h , N_m , and N_f are the number of correct detections, missed

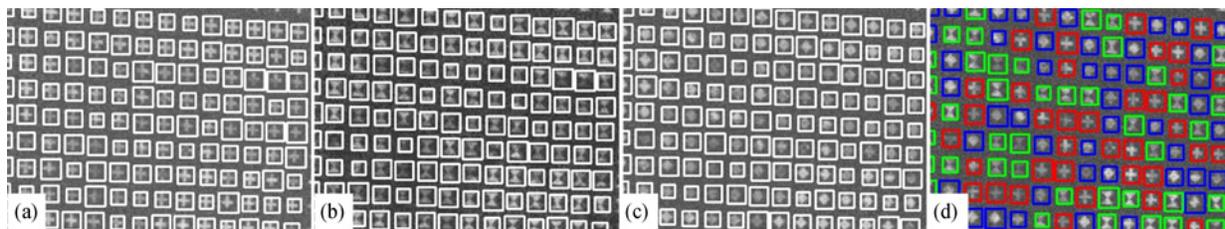


Fig. 9. Some qualitative experiment results on accuracy evaluation. (a) Cross shape detection. (b) Sandglass shape detection. (c) Rhombus shape detection. (d) Spatially coded pattern detection. The cross, sandglass, and rhombus shapes are separately marked by red, green, and blue rectangles.

TABLE I
BENCHMARK FOR SENSITIVITY EVALUATION

	H(%)	M(%)	F(%)	$E_d(\text{pixel})$	Corr. Acc.(%)
Cross	94.53	3.95	1.52	1.632	—
Rhombus	95.21	3.59	1.20	1.833	—
Sandglass	95.50	3.63	0.87	1.542	—
Whole Pattern	92.11	11.06	5.28	2.013	95.74

detections and false detections, respectively. ϵ_x and ϵ_y are the average feature point detection errors along the x -axis and y -axis, (X_d, Y_d) and (X_g, Y_g) are the detected coordinate and ground-truth, respectively.

The more detailed quantitative testing results are listed in Table I. Through the proposed method, 95.74% of the embedded feature points could their correspondences found correctly. By analyzing the missed and false detection cases, we find that the mistakes were mainly caused by large noise that occludes the embedded codes, implying that external noise has the greatest influence on the decoding process.

D. Robustness Evaluation

Due to the low signal-to-noise ratio property in imperceptible structured light sensing scenario, the embedded primitive shapes in the subtraction image are prone to be disturbed by the noises. Although our pretrained shape detectors have good anti-noise feature, some of primitive shapes are still missed or detected falsely, as shown in Table I. In order to obtain robust decoding, the larger Hamming distance is necessary in encoding stage, which ensures that correspondences can be still established correctly even if some bits in the codeword are unknown or wrong.

In this subsection, we compare the performances of different coding schemes with different Hamming distances on *PROCAMS-A*. Firstly, we followed the approach described in Section III-B to generate five coded pattern images with fixed Hamming distance $H_d = 1, \dots, 5$. The new generated pattern images have the same dimension as our proposed pattern image. These coded pattern images were embedded into 500 images used for imperceptibility evaluation, respectively, with intensity $\Delta = 10$. Next, just like the experiment of primitive shape detection evaluation, the projected and complementary images were projected successively to a projection surface, while the camera conducted synchronized capture. The projection surface was the same as that in primitive shape detection evaluation. Eventually, the subtraction images embracing different embedded codes were derived for robustness evaluation.

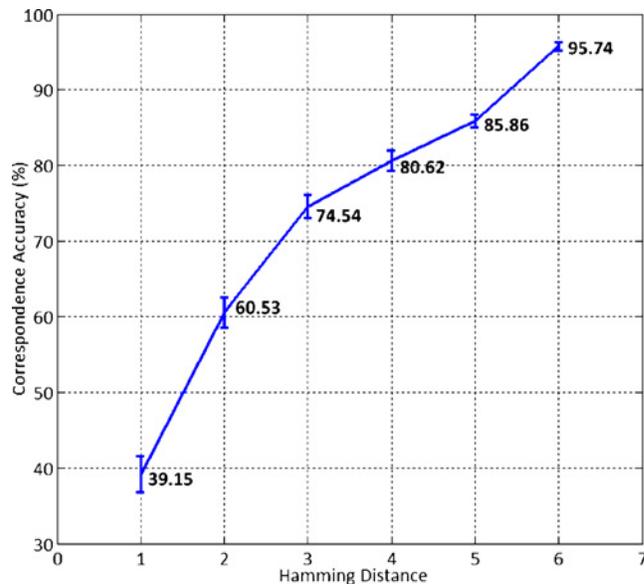


Fig. 10. Correspondence accuracy in different Hamming distance coding.

The ground-truth was obtained by manual labeling in the image data captured under regular binary pattern illumination.

Experimental results are illustrated in Fig. 10. When Hamming distance $H_d = 1$, which is adopted in traditional M-array coding approaches, in ISL only 39.15% of feature points could find their correspondences correctly. With the increase of Hamming distance, the correspondence accuracy is improved remarkably and in our proposed pattern, when the average Hamming distance $\bar{H}_d = 6$, the correspondences accuracy reaches 95.74%. Moreover, from the error bar, it is evident that the larger the Hamming distance is, the more robust the performance will be to the variations of projection surface and projected image.

E. Efficiency Evaluation

For some application scenarios, such as sensing moving or deformable objects, real-time performance is of great importance. Hence we implemented the proposed system in C++ using the Intel OpenCV Library [29] to evaluate its processing time. Through multithread programming, the projection-capture process and calculation process were executed in two different threads, respectively, each of which was able to run in real time in a desktop computer with Intel Core i5-760 2.8GHz CPU. Table II shows the average processing times for primitive shape detection, correspondence searching, and triangulation

TABLE II
AVERAGE PROCESSING TIME FOR THE 3-D SENSING PROCEDURE

Subroutine	Time (ms/frame)			
	High Resolution (1024 × 768)		Low Resolution (640 × 480)	
	w/ D.S.	w/o D.S.	w/ D.S.	w/o D.S.
Primitive Shape Detection	24.21	68.37	13.95	25.34
Correspondence Searching	4.53	4.46	4.47	4.49
Triangulation	3.12	3.12	3.16	3.13
Total	31.86	75.95	21.58	32.96

in the given system. Primitive shape detection is the most time-consuming process. Due to the searching characteristic of AdaBoost-based detector, the processing times varied according to image resolution. However, they all satisfied the requirement of real-time application. We conducted experiments that use cross shape embracing directional information in searching (D.S.) and which do not, respectively, the results shown in Table II indicate that the directional information could speed up primitive shape detection notably, especially in the high resolution case.

V. SENSITIVITY EVALUATION

It is obvious that the performance of our method depends on the performance of pretrained primitive shape detectors, which is determined by the training process to a great extent. Generally, for the training-based methods, generalization of the training results is an issue, especially, when the scenarios between training stage and operation stage are quite different.

In the framework of our method, due to the different sensor-object localization, different projection surfaces, different surrounding environment and different hardware platforms, the generalization of the pretrained detector is of great importance, since it is both impractical and impossible to retrain the detector for different scenarios. It is necessary to certify the validity of our method in different application scenarios.

In this section, we will evaluate the the sensitivity of primitive detectors under different circumstances, including variations on working distance, projection surface orientation, projection surface shape, projection surface texture and hardware configuration. Since the settings of accuracy evaluation in Section IV-C are the same as training sample collection stage, the results are considered as the benchmark for sensitivity evaluation.

A. Working Distance

The working distance is the average distance from the projector-camera system to the object surface. When the intrinsic parameters of the projector and camera (focal length and resolution) are fixed, the size of the primitive shapes in subtraction image data is determined by the working distance directly. In the configuration of training stage, the working distance is set as 800mm, the size of primitive shapes in image data is about 20 pixels. In the operation stage, the working distance is changed to 500mm, 1200mm, and 1600mm, the focal length of procams is slightly adjusted to get sharp projection and clear capture. Some subtraction images with

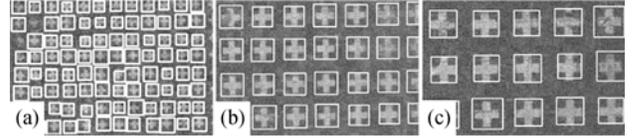


Fig. 11. Cross shape detection in different working distances. (a) 500mm, (b) 1200mm, and (c) 1600mm.

TABLE III
PRIMITIVE SHAPE DETECTION ACCURACY IN
DIFFERENT WORKING DISTANCES

Distance	Pri.	H(%)	M(%)	F(%)	E_d (pixel)
500mm	C	86.21	11.63	2.16	1.814
	R	85.83	12.57	1.60	1.836
	S	87.49	11.64	0.87	1.712
1200mm	C	94.44	4.32	1.24	1.728
	R	94.86	4.23	0.91	1.904
	S	94.49	4.62	0.89	1.572
1600mm	C	94.52	4.11	1.37	1.731
	R	95.06	3.92	1.02	1.910
	S	95.39	3.68	0.93	1.591

detection results are shown in Fig. 11; the size of the primitive shapes are around 15, 35, and 45 pixels, respectively.

The detailed quantitative results are listed in Table III. It is clear that when the working distance decreased to 500mm, the hit rates dropped, because it is difficult for primitive shape detectors to find small size shapes in image data. For the enlarged shapes in larger working distance, the performance of detectors are almost the same as those of the benchmark.

B. Projection Surface Orientation

Besides the size of the primitive shapes in image data, the distortions will also influence the performance of the pretrained detectors. The distortions mainly come from the variations on the orientation of the projection surface with respect to the sensing system and the variations on the shape of the projection surface. First, the detector accuracy will be evaluated under different projection surface orientations.

In the training data collection stage, the images were projected to a planar surface that is almost parallel to the image plane of the camera. Now in the operation stage, the orientation of the surface is adjusted to 10°, 20°, 30°, 40°, 50° in the yaw direction, as shown in Fig. 12. In each subimage, the upper part is the captured image to show the extent of distortion, while the lower part is the magnified subtraction image of the subregion indicated by the rectangle in captured image. The detection results are also shown in

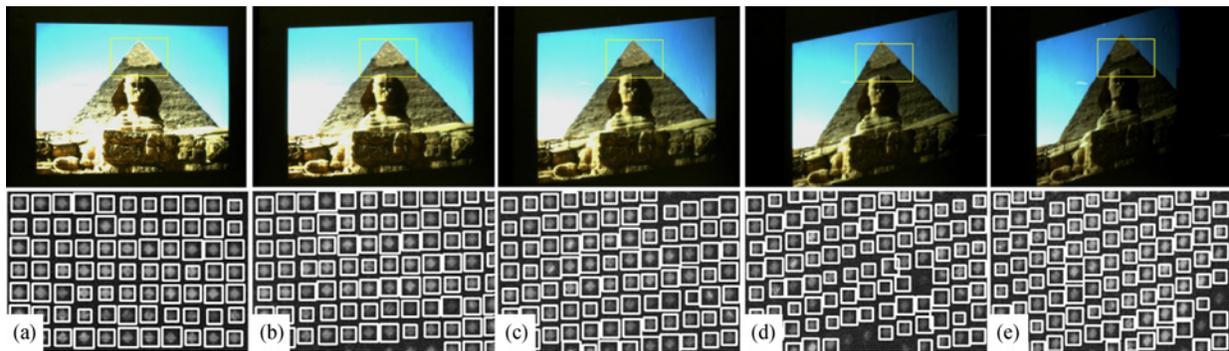


Fig. 12. Rhombus shape detection in projection surface under different orientations. (a) 10°. (b) 20°. (c) 30°. (d) 40°. (e) 50°.

TABLE IV
PRIMITIVE SHAPE DETECTION ACCURACY UNDER
DIFFERENT SURFACE ORIENTATIONS

Orientation	Pri.	H(%)	M(%)	F(%)	E_d (pixel)
10°	C	94.51	3.96	1.53	1.635
	R	95.08	3.60	1.22	1.845
	S	95.46	3.74	0.80	1.544
20°	C	94.50	3.96	1.54	1.634
	R	95.08	3.64	1.08	1.848
	S	95.43	3.77	0.80	1.564
30°	C	93.47	4.50	2.03	1.938
	R	92.15	6.37	1.48	2.141
	S	92.43	6.78	0.79	2.011
40°	C	90.19	7.70	2.11	2.414
	R	89.42	9.50	1.08	2.809
	S	91.23	7.87	0.90	2.374
50°	C	85.91	12.03	2.06	2.728
	R	85.48	12.81	1.71	2.904
	S	86.87	12.27	0.86	2.572

the subtraction images. More detailed quantitative results are listed in Table IV.

In the testing results, when the rotation degree θ is small, i.e., $\theta = 10^\circ, 20^\circ$, the performance is almost the same as those of the benchmark. With the increase of the rotation degree, the hit rates decrease slightly. When $\theta = 50^\circ$, more than 85% primitive shapes are still detected correctly, which satisfies the application requirements.

C. Projection Surface Shape

The alteration of projection surface shape will also result in the distortion of primitive shapes in the image data. In the training stage, the negative and positive sample were collected from the images projected to a planar surface. In this test, the projection surface are three different non-planar surfaces (convex paper, concave paper and plaster statue). Some test images and the statistical results are shown in Fig. 13 and Table V, respectively. In all three surfaces, although the hit rates have small decrease, it is still sufficient to derive correct correspondences for triangulation. In the plaster statue case, the missing detections are mainly found in the regions where the surface has sudden change.

D. Projection Surface Texture

The texture on the projection surface will affect the quality of captured images. In the benchmark training stage, the

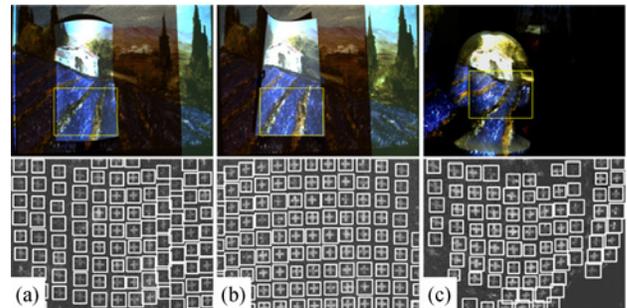


Fig. 13. Cross shape detection from different projection surfaces. (a) Convex paper. (b) Concave paper. (c) Plaster statue.

TABLE V
PRIMITIVE SHAPE DETECTION ACCURACY FROM PROJECTION
SURFACES OF DIFFERENT SHAPES

Surface	Pri.	H(%)	M(%)	F(%)	E_d (pixel)
Convex Paper	C	93.53	4.86	1.61	1.756
	R	93.25	5.29	1.46	2.043
	S	94.14	4.85	1.01	2.122
Concave Paper	C	93.64	4.84	1.52	1.762
	R	93.82	4.70	1.48	2.108
	S	93.76	5.41	0.83	2.135
Plaster Statue	C	84.81	13.33	1.86	2.028
	R	85.73	13.06	1.21	1.904
	S	86.09	13.03	0.88	2.075

projection surface is texture-free and in white color. In the operation stage (this test), the images are projected to a planar surface in green color, a cork board and a poster with text and images, as illustrated in Fig. 14. The quantitative results are listed in Table VI. The results indicate that the texture variation on the projection surface has little influence on the performance of the primitive shape detectors, since in our method the decoding process was conducted in the subtraction image, which reduces the texture's influence substantially.

E. Projector-Camera System

If the pretrained detectors are used in another application with different hardware configuration, the performance of the detectors would be affected, since the differences in the resolution of the projector and camera (high versus low), the camera sensor (CCD versus CMOS), and the optical parameters (different lenses) will change the appearance of the

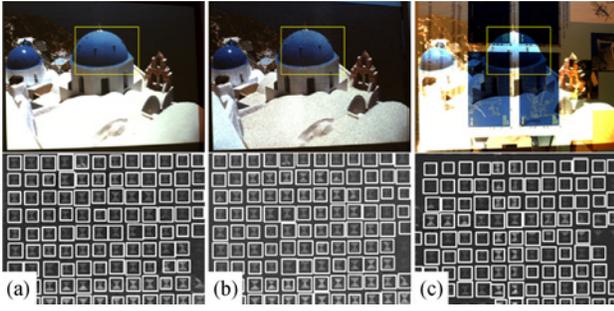


Fig. 14. Sandglass shape detection in different projection surface textures. (a) Green paper. (b) Cork board. (c) Poster.

TABLE VI

PRIMITIVE SHAPE DETECTION ACCURACY IN DIFFERENT PROJECTION SURFACE TEXTURE

Texture	Pri.	H(%)	M(%)	F(%)	E_d (pixel)
Green Paper	C	94.41	4.17	1.42	1.634
	R	95.19	3.66	1.15	1.836
	S	95.49	3.63	0.88	1.558
Cork Board	C	93.41	5.07	1.52	1.641
	R	94.25	4.43	1.32	1.850
	S	94.92	4.16	0.92	1.623
Poster	C	91.74	6.63	1.63	2.024
	R	90.28	8.25	1.47	1.996
	S	92.19	6.76	1.05	1.762

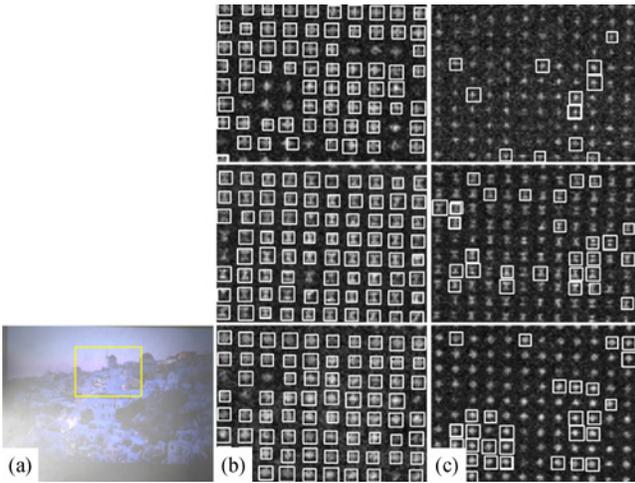


Fig. 15. Primitive shape detection in PROCAMS-B with different embedding approaches. (a) Captured image. (b) Cropped pattern. (c) Resized pattern.

primitive shape in the image data. In this test, the primitive detectors trained by the data collected from *PROCAMS-A* were applied to *PROCAMS-B* in the operation stage.

Due to the low projector resolution in *PROCAMS-B*, the dimension of the original pattern image was too large for embedding. We employed two methods to solve the issue. The first one was to select a subregion of the original pattern image as a new pattern image and the second one was to resize the original pattern image to coincide with the projector resolution. Some detection results in the subtraction images derived from the two different embedding methods are illustrated in Fig. 15(b) and (c). The quantitative results are also shown in Table VII.

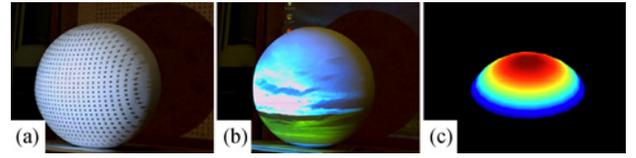


Fig. 16. Comparison experiment on sensing sphere object. (a) GSL illumination. (b) ISL illumination. (c) Reconstruction result of ISL.

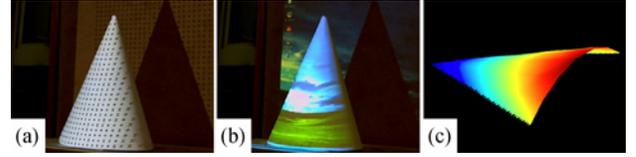


Fig. 17. Comparison experiment on sensing cone object. (a) GSL illumination. (b) ISL illumination. (c) Reconstruction result of ISL.



Fig. 18. Comparison experiment on sensing cylinder object. (a) GSL illumination. (b) ISL illumination. (c) Reconstruction result of ISL.

Compared with the benchmark, it is obvious that the performance in *PROCAMS-B* degraded intensively, especially in the resized pattern case. By analyzing the missed and false detection cases, we found that the mistakes were mainly caused by large noise from the low luminance of the pico projector and the extremely small primitive shapes in the image data.

VI. APPLICATIONS

The proposed method enables a common projector to serve the dual role of a display device as well as a 3-D sensor, which can be extended or integrated to many applications. In this section, we will show three cases to demonstrate the feasibility of our method.

A. 3-D Reconstruction with Regular Video Projection

3-D reconstruction is the most straightforward application for SLS. To show the effectiveness of our method in 3-D reconstruction task, we compared our method with the general structured light method that uses visible patterns.

As shown in Figs. 16(a), 17(a), 18(a), and Figs. 16(b), 17(b), 18(b), three objects (sphere, cone, and cylinder) with known dimensions were illuminated by visible binary pattern image (the same as Fig. 4) and code embedded normal projection, respectively.

In the general structured light scenario, feature points were extracted by segmentation and shape identification using the method proposed in [22], whilst in our code embedded regular projection scenario, feature points were detected and classified through the pretrained primitive shape detectors. The depth value of each feature point was calculated through triangulation using the intrinsic and extrinsic parameters of projector

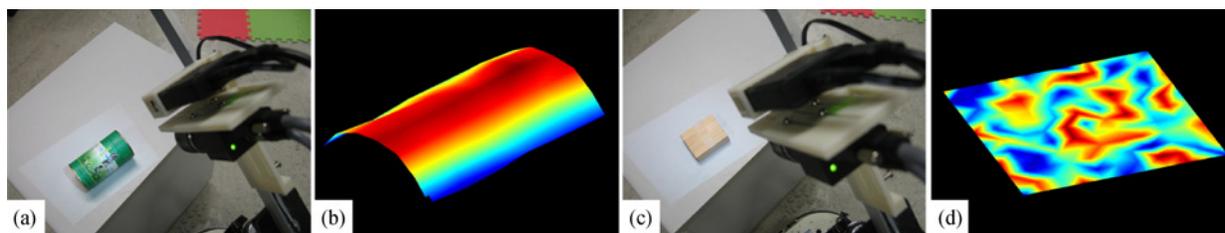


Fig. 19. Some 3-D sensing results on mobile robot. (a) ISL on green tea can. (b) Recovered surface of green tea can. (c) ISL on toy bricks. (d) Recovered surface of toy bricks.

TABLE VII

PRIMITIVE SHAPE DETECTION ACCURACY IN PROCAMS-B WITH DIFFERENT EMBEDDING APPROACHES

	Pri.	H(%)	M(%)	F(%)	$E_d(\text{pixel})$
Cropped Pat.	C	80.23	14.43	5.34	3.028
	R	79.93	14.17	5.92	2.981
	S	81.09	13.28	5.63	2.812
Resized Pat.	C	30.52	59.23	10.25	2.628
	R	30.63	58.03	11.34	2.913
	S	30.80	57.93	11.27	2.874

TABLE VIII

COMPARISON OF 3-D RECONSTRUCTION ACCURACIES

Object	General SL [22]		Our Method	
	$E_\mu(\text{mm})$	$E_\sigma(\text{mm})$	$E_\mu(\text{mm})$	$E_\sigma(\text{mm})$
Sphere	1.502	0.576	1.410	0.587
Cylinder	2.054	0.824	1.939	0.762
Cone	1.383	0.557	1.391	0.564

and camera. Then on the basis of point clouds calculated through our method, surfaces were rendered as illustrated in Figs. 16(c), 17(c), and 18(c). Since the dimensions of the objects were known, we could conduct quantitative accuracy assessment. The residual mean error E_μ and standard deviation E_σ of the calculated 3-D points with respect to ground-truth are listed in Table VIII. It is evident that our method has almost the same performance as that of the general structured light method in 3-D reconstruction. By the reason that textures on the cylindrical object obstruct code retrieval, the reconstruction error on the cylindrical object is greater than those of the other two objects. It is worth pointing out that in our method the decoding process was conducted in the subtraction image, which would reduce the texture influence.

B. Sensing Surrounding Environment on Mobile Robot Platform

For the purpose of illustrating the proposed method's potential applications in robotic system working in varied environment, we mounted a projector and a camera rigidly on a specially designed frame, and fixed the frame to a tripod affixed to a mobile robot manufactured by ARRICK Robotics [30], as shown in Fig. 20(a).

For a mobile robot, one of the essential capabilities is to sense the surrounding environment for navigation, obstacle avoidance, object recognition and some other purposes. We assist the visual sensing through a normal grey illumination with

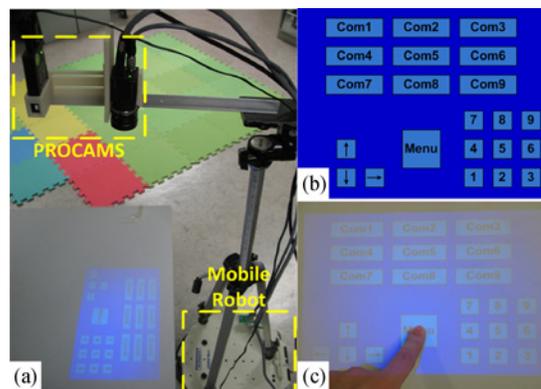


Fig. 20. Touch-sensitive user interface on projection surface. (a) Integration with mobile robot system and application scene. (b) User interface image. (c) User's click action on projected interface.

invisible codes embedded. By retrieving the embedded codes, correspondences between projection plane and image plane could be established accurately and efficiently. In Figs. 19(a) and (c), a green tea can and toy bricks were located in the illumination area of the projector, and 3-D depth information of certain points on the objects was acquired through simple triangulation in real-time. The surfaces of the objects were rendered in 3-D as shown in Figs. 19(b) and (d). Although the ground truth of the objects was not available, qualitative examination showed that the reconstructed surfaces were of reasonable quality.

C. Natural Human-Computer Interaction

Besides sensing capabilities, the mobile robot should also provide an effective channel for the interaction between users, such as an interface for system configuration or a display panel to show prompt information. Traditionally, an LCD monitor plus mouse-and-keyboard or an LCD touch-screen attached to the robot is used, which would inevitably increase the weight and size of the mobile robot, plus energy consumption. Our method enables a common projector to serve the dual role of a display device as well as a 3-D sensor with the assistance of camera, providing a platform for more natural user interface schemes. As shown in Fig. 20(a), a system configuration interface [Fig. 20(b)] was projected onto a desk surface, and a user was operating on the projected desk surface with bare-hand [Fig. 20(c)]. From an image alone, say of a finger on top of a table surface, one cannot tell whether the finger is actually touching the table surface or not. The case of a finger hanging

in air, and the case of a finger touching the table surface, could both produce the same image to the camera. By incorporating the structured light invisible embedded into the projection, 3-D acquisition can be made possible, and contact identification and finger movement recognition could be readily tackled.² It is possible to convert any textureless light color plane (table-surfaces, whiteboards or walls) to a touching sensitive screen, providing more natural and flexible interface for bare-hand human-robot interaction.

VII. CONCLUSION

We described a novel system of embedding imperceptible structured codes into normal projection that strikes the balance between imperceptibility and detectability of the codes. Through precise projector-camera synchronization, structured codes consisting of three primitive shapes were embedded into regular projection, in a way that is imperceptible to the user but extractable by a camera (through the difference image between successive images). The disturbances caused by external noise made it difficult to retrieve the codes by the region segmentation approaches adopted in general structured light-based systems. Instead of segmenting the codes, specially trained classifiers were employed to detect and identify them. To increase the robustness of code extraction, large Hamming distance was adopted in spatial coding. Even if some bits were missed or wrongly decoded, the correct correspondence between the projection panel and the image plane could still be arrived at correctly for structured light sensing. Extensive evaluations showed that the method is a promising one.

In the current system, the image capture interval is 10ms. In sensing object that moves fast, the substantial displacement between successive images will result in blur or destruction of the embedded codes in the difference image. Some compensation methods need be in place to deal with the issue and when the method is applied in the projector-camera system with low resolution, due to the low quality of the subtraction image, the performance deteriorates dramatically. Some image enhancement methods should be studied as the preprocessing step for primitive shape detection. In addition, the embedded code could be denser for more precise 3-D sensing. New coding scheme capable of generating denser patterns should be used. The proposed method enabled a common projector to serve the dual role of a display device as well as a 3-D sensor. That provides a platform for more natural user interface schemes. Our future work will lie on these directions.

REFERENCES

- [1] J. Salvi, S. Fernandez, T. Pribanic, and X. Llado, "A state of the art in structured light patterns for surface profilometry," *Pattern Recogn.*, vol. 43, no. 8, pp. 2666–2680, 2010.
- [2] O. Bimber, D. Iwai, G. Wetzstein, and A. Grundhöfer, "The visual computing of projector-camera systems," in *Proc. ACM SIGGRAPH Classes*, 2008, pp. 1–25.
- [3] D. Fofi, T. Sliwa, and Y. Voisin, "A comparative survey on invisible structured light," in *Proc. Mach. Vision Appl. Ind. Inspection XII*, 2004, pp. 90–98.
- [4] J. Dai and R. Chung, "On making projector both a display device and a 3-D sensor," in *Proc. 8th Int. Symp. Visual Comput.*, 2012, pp. 654–664.
- [5] J. Dai and R. Chung, "Embedding imperceptible codes into video projection and applications in robotics," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2012, pp. 4399–4404.
- [6] R. Raskar, G. Welch, M. Cutts, A. Lake, L. Stesin, and H. Fuchs, "The office of the future: A unified approach to image-based modeling and spatially immersive displays," in *Proc. SIGGRAPH*, 1998, pp. 179–188.
- [7] D. Cotting, M. Naef, M. Gross, and H. Fuchs, "Embedding imperceptible patterns into projected images for simultaneous acquisition and display," in *Proc. 3rd IEEE/ACM Int. Symp. Mixed Augmented Reality*, Nov. 2004, pp. 100–109.
- [8] H. Park, B.-K. Seo, and J.-I. Park, "Subjective evaluation on visual perceptibility of embedding complementary patterns for nonintrusive projection-based augmented reality," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 5, pp. 687–696, May 2010.
- [9] H. Park, M. hyun Lee, B. kuk Seo, Y. Jin, and J. il Park, "Content adaptive embedding of complementary patterns for nonintrusive direct-projected augmented reality," in *Proc. HCI Int.*, 2007, pp. 132–141.
- [10] A. Grundhöfer, M. Seeger, F. Hantsch, and O. Bimber, "Dynamic adaptation of projected imperceptible codes," in *Proc. 6th IEEE/ACM Int. Symp. Mixed Augmented Reality*, Nov. 2007, pp. 1–10.
- [11] K. L. Boyer and A. C. Kak, "Color-encoded structured light for rapid active ranging," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 9, no. 1, pp. 14–28, Jan. 1987.
- [12] J. Salvi, J. Batlle, and E. Mouaddib, "A robust-coded pattern projection for dynamic 3-D scene measurement," *Pattern Recogn. Lett.*, vol. 19, no. 11, pp. 1055–1065, 1998.
- [13] J. Pages, J. Salvi, and J. Forest, "A new optimised de Bruijn coding strategy for structured light patterns," in *Proc. 17th Int. Conf. Pattern Recogn.*, vol. 4, 2004, pp. 284–287.
- [14] T. Etzion, "Constructions for perfect maps and pseudorandom arrays," *IEEE Trans. Inf. Theory*, vol. 34, no. 5, pp. 1308–1316, Sep. 1988.
- [15] H. Morita, K. Yajima, and S. Sakata, "Reconstruction of surfaces of 3-D objects by m-array pattern projection method," in *Proc. 2nd Int. Conf. Comput. Vision*, 1988, pp. 468–473.
- [16] R. Morano, C. Ozturk, R. Conn, S. Dubin, S. Zietz, and J. Nissano, "Structured light using pseudorandom codes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 3, pp. 322–327, Mar. 1998.
- [17] J. Pages, C. Collewet, F. Chaumette, and J. Salvi, "An approach to visual servoing based on coded light," in *Proc. IEEE Int. Conf. Robotics Autom.*, May 2006, pp. 4118–4123.
- [18] M. Maruyama and S. Abe, "Range sensing by projecting multiple slits with random cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 6, pp. 647–651, Jun. 1993.
- [19] P. Fechteler and P. Eisert, "Adaptive colour classification for structured light systems," *IET Comput. Vision*, vol. 3, no. 2, pp. 49–59, 2009.
- [20] T. Koninckx and L. Van Gool, "Real-time range acquisition by adaptive structured light," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 3, pp. 432–445, Mar. 2006.
- [21] H. Kawasaki, R. Furukawa, R. Sagawa, and Y. Yagi, "Dynamic scene shape reconstruction using a single structured light pattern," in *Proc. IEEE Conf. Comput. Vision Pattern Recogn.*, Jun. 2008, pp. 1–8.
- [22] C. Albitar, P. Graebler, and C. Doignon, "Robust structured light coding for 3-D reconstruction," in *Proc. IEEE 11th Int. Conf. Comput. Vision*, Oct. 2007, pp. 1–6.
- [23] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vision Pattern Recogn.*, vol. 1, 2001, pp. 511–518.
- [24] Google. (2013, Apr.). *Google Image* [Online]. Available: <http://images.google.com/>
- [25] R. Lienhart, E. Kuranov, and V. Pisarevsky, "Empirical analysis of detection cascades of boosted classifiers for rapid object detection," in *Proc. DAGM 25th Pattern Recogn. Symp.*, 2003, pp. 297–304.
- [26] R. I. Hartley and P. Sturm, "Triangulation," *Comput. Vision Image Understand.*, vol. 68, no. 2, pp. 146–157, 1997.
- [27] TI. (2013, Apr.). *Pico Development Kit* [Online]. Available: <http://www.ti.com/tool/dlp1picokit/>
- [28] Z. Song and R. Chung, "Use of LCD panel for calibrating structured-light-based range sensing system," *IEEE Trans. Instrum. Meas.*, vol. 57, no. 11, pp. 2623–2630, Nov. 2008.
- [29] OpenCV. (2013, Apr.) [Online]. Available: <http://www.opencv.org/>
- [30] ARRICK Robotics. (2013, Apr.) [Online]. Available: <http://www.arrickrobotics.com/>, accessed: 10/04/2013.

²We have implemented fingertip touching detection method under invisible codes embedded illumination, but this is to be detailed in another article due to the-space constraint.



Jingwen Dai (M'12) received the B.E. degree in automation from Southeast University, Nanjing, China, in 2005, the M.E. degree in automation from Shanghai Jiao Tong University, Shanghai, China, in 2009, and the Ph.D. degree in mechanical and automation engineering from the Chinese University of Hong Kong, Hong Kong, in 2012.

He is a Post-Doctoral Research Associate with the Department of Computer Science, University of North Carolina at Chapel Hill, Chapel Hill, NC, USA. His research interests include computer vision and human-computer interaction.



Chi-Kit Ronald Chung (SM'99) received the B.S.E.E. degree from the University of Hong Kong, Hong Kong, and the Ph.D. degree in computer engineering from the University of Southern California, Los Angeles, USA.

He is the Deputy Executive Director of the Vocational Training Council (VTC) of Hong Kong, Hong Kong. He is also an Adjunct Professor with Chinese University of Hong Kong (CUHK), Hong Kong. Prior to joining VTC, he was Professor and Department Chairman of Mechanical and Automation Engineering with CUHK. His research interests include computer vision and robotics.

He is a fellow of HKIE, a fellow of BCS, CEng of the Engineering Council of U.K., and a member of MENSA. He has served the academic communities in various capacities including serving as the Chairman of the IEEE Hong Kong Section Joint Chapter on Robotics and Automation Society and Control Systems Society in 2001-2003.