IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 3, 2021, accepted March 28, 2021, date of publication April 2, 2021, date of current version April 14, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3070655

Projector and Camera Shade Detection Based on Stripe Pattern Sequence for High Quality Structured Light Depth Imaging

SUKHAN LEE[1,2], (Life Fellow, IEEE), AND MUHAMMAD ATIF[2]
[1]Department of Artificial Intelligence, Intelligent Systems Research Institute (ISRI), Sungkyunkwan University, Suwon 440-746, South Korea
[2]College of Information and Communication Engineering, Sungkyunkwan University, Suwon 440-746, South Korea

Corresponding author: Sukhan Lee (lsh1@skku.edu)

This work was supported in part by the "Cross-Sensory Transfer for Visually Impaired" project under Grant 2020R1A2C200956811, in part by the "SMART Patient Isolation Transport Unit System" project of the National Research Foundation (NRF) under Grant HW20C2077020020, in part by the AI Graduate School Program under Grant 2019-0-00421, and in part by the ICT Consilience Program sponsored by the Korea Ministry of Science and ICT (MSIT) under Grant IITP-2020-0-01821.

ABSTRACT Structured light depth imaging offers a high precision of 3D measurement for applications to industrial modeling, inspection, reproduction and archiving. Despite recent progress, however, structured light depth imaging has yet to settle the trade-off between the number of captured 3D points and that of unwanted outliers, in particular, when a low reflectance surface is mixed with projector and camera shades. This paper presents an approach to solving such trade-off by accurately mapping projector and camera shades under the presence of a low reflectance surface. Unlike conventional approaches relying mostly on pattern intensity for detecting shades, the proposed approach first identifies stripe boundaries legitimate in terms of the projected pattern sequence. Based on the list of legitimate boundaries identified, then, the gaps in pixel and boundary addresses are analyzed to define the respective projector and camera shades, possibly, in the presence of address juxtaposition. Experimental results show that the proposed approach shows superior performance in shade detection as well as outlier removal compared to the state-of-the-art approaches, maximizing the number of captured 3D points while minimizing that of unwanted outliers.

INDEX TERMS Structured light depth imaging, camera and projector shade, shade detection, outlier removal.

I. INTRODUCTION

Structured light 3D imaging is capable of delivering a high precision of depth measurement, even to the level of microns, within its dynamic range. As such, it is a preferred option for applications requiring a highly accurate depth measurement as their key feature. To date, the applications of structured light 3D imaging have expanded to cover such areas as robotics [1], endoscopy [2], industrial automation [3], industrial inspection [4], human dental treatment [5], as well as the preservation of cultural heritage [6].

The working principle of structured light depth imaging is similar to that of stereo imaging; depth is defined by triangulation of the lines of sight from a pair of corresponding pixels of a stereo camera. Unlike stereo imaging, however, structured light 3D imaging is based on an active device, and light patterns projected by a projector onto a scene are captured by a camera to compute depth by triangulation. In industry, structured light 3D camera systems are preferred to stereo systems due to the ability of the former to deal with poor illumination and un-textured surfaces, in addition to the accuracy in depth measurement.

More specifically, for a single camera-single projector structured light imaging setup, as illustrated in Fig. 1, light patterns are projected by the digital micro-mirror device (DMD) of the projector and captured by the image sensor of the camera in such a way that the correspondence between the projected pattern stripes and the detected pattern stripes on the image plane can be established. A number of approaches to coding light patterns have been introduced to date based on temporal [16], [29], spatial [40] or hybrid [39] coding approaches, with accurate and efficient decoding of the stripe correspondence as the goal. They include the gray codec [35], phase shift codec [3], [4], [24], spatial codec [40] and their

The associate editor coordinating the review of this manuscript and approving it for publication was Marco Martalo.

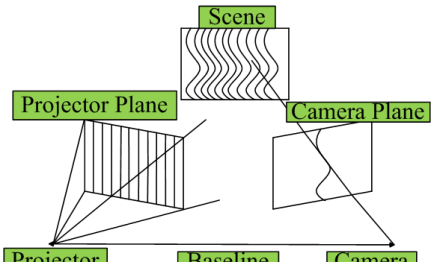


FIGURE 1. A single projector-single camera setup for structured light depth imaging.

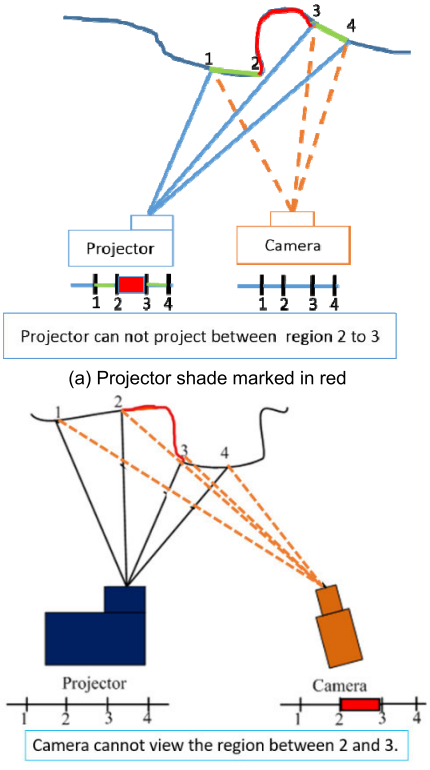


FIGURE 2. Projector and camera shades in a structured light 3D camera system.

many variants [16], [39]. Once decoded for the stripe correspondence, the depth is computed based on a triangulation process applied to the corresponding pixel pairs from the projected and detected stripes.

The performance of structured light depth measurement can be assessed in terms of the number of captured 3D points, the accuracy of the captured points, and the number of outliers included in the captured 3D points, among others. The following factors influence the performance of structured light depth measurement: whether the surface to be measured is shaded or occluded from the projector and the camera, the reflection characteristics of the surface, the illumination condition and the scattering and inter-reflection properties of the surface. One thing to note is that there exists a trade-off between the number of 3D points captured and the number of outliers included in the captured 3D points. This trade-off becomes especially problematic when surfaces with low light reflectance are mixed with surfaces that are shaded from the camera and projector views or, in short, the camera and projector shades. To be more specific, consider region 2-3 marked in red in Fig. 2 (a), which represents the projector shade in the camera view. Although light patterns from the projector cannot reach that region, the camera can receive ambient light from the shade due to light scattering and inter-reflection. Should region 3-4 or 1-2 (in green) be a surface with low light reflectance, it may be difficult to distinguish the surfaces of 3-4 and 1-2 from the shade of 2-3 due to the combined effect of low-intensity readings and inter-reflection. This leads either to a hole, a set of missing points, in depth imaging when a part of the low-reflective surface is identified as part of the projector shade, or to an outlier when a part of the projector shade is identified as the low-reflectance surface.

By the same token, consider region 2-3 marked in red in Fig. 2 (b), representing the camera shade in the projector view. In this case, the projected light can reach the surface but the camera cannot receive the light due to the shade. This results in missing stripe patterns from the projector in the camera frame. Since the camera shade can disturb the correct decoding of the detected stripe boundaries, especially when the shade is bordered by the surface of low reflectance, outliers may occur. Note that an increase in the sensitivity of pattern stripe detection, especially for a low reflectance surface, results in an increase in the number of outliers from the projector and camera shades. To overcome this trade-off, it is necessary to develop an accurate means of detecting boundaries of the projector and camera shades.

A number of approaches to detection of the camera and projector shades have been proposed to date. However, they are mostly based on the intensity of received light such that they perform well only when either the surface under consideration has a high reflection coefficient or the camera receiving the light adopts a long exposure time.

A. RELATED WORK

In this subsection, we introduce previous work related to the detection of shadows or shades, although the work may not directly address the camera and projector shades in a structured light depth measurement setting. Most conventional approaches to detection of shades explore the difference in light reflectance properties between the shade and the object surface, such as the light intensity and spectral response. DiCarlo *et al.* [13] presented an approach to pixel-wise estimation of surface reflectance from two images: one with ambient light and the other with camera flash. Shades can be identified as the region of low reflectance. However, performance degrades in the presence of surfaces with low reflectance as well as in cases with inter-reflection from the environment. Ruiqi Guo *et al.* [14]

presented a region based approach to labeling of shadow and non-shadow regions based on pairwise classification of segmented regions through prediction of relative illumination conditions. Although more robust than pixel or edge-based approaches, it shows a decrease in performance when the size of the segmented regions becomes smaller, and incurs a heavy computational cost. Fredembach *et al.* [9] proposed an approach to shadow detection in a scene that takes advantage of the properties of near-infrared (NIR) light, which is more strongly reflected by dark surfaces than visible light. Shadows are identified by observing NIR and visible light images along with the ratio of their intensities in image pixels. However, this approach has a limitation in that different surface materials alter the properties of NIR reflection. Finlayson *et al.* [10] proposed an approach to estimation of an illumination map based on normal and color-filtered images of the same scene, in which shades are detected using a broadband color filter. Note that the shade detection approaches introduced above mainly identify the shadows caused by ambient or external illumination. Although applicable to the detection of projector and camera shades, to date, they have not been exploited much for detection of projector and camera shades in a structured light depth imaging setting. Instead, the projector and camera shades are discriminated from the stripe patterns projected on object surfaces simply based on the intensity of stripe signals during the process of decoding the camera-projector pixel correspondence. Otherwise, they may employ a multi-camera system that minimizes the size of projector shade regions, as tackled by Park *et al.* [12]. A major problem common to intensity-based shade detection systems is the difficulty of discriminating a surface with low light reflectance from the shade. This problem often causes generation of many unwanted outliers in 3D reconstruction as the stripe signals are scattered and inter-reflected into the shades. To address this problem, Dung *et al.* [8] considered the luminance variation of images captured when different structured light patterns are projected with only limited success for distinguishing between the projector shade and a low-reflectance surface.

B. PROBLEM DEFINITION AND PROPOSED APPROACH

Accurate detection of projector and camera shades is crucial for the quality of structured light depth imaging, as pattern stripes erroneously detected from shades due to scattering and inter-reflection cause outliers. Most of the conventional codecs developed for structured light depth imaging rely on the intensity of received pattern signals to determine the presence of stripes on the surface while excluding those on the shade. However, they perform well only when either the surfaces under consideration have a high reflection coefficient or a long camera exposure time is employed. There exists a trade-off between the number of 3D points that are captured and the number of outliers included in the captured 3D points. This trade-off becomes especially problematic when surfaces with low light reflectance are mixed with surfaces that are shaded by the projector and camera shades. For instance, increase in the sensitivity of pattern stripe detection, especially, for a low-reflectance surface causes an increase in the number of outliers from the projector and camera shades.

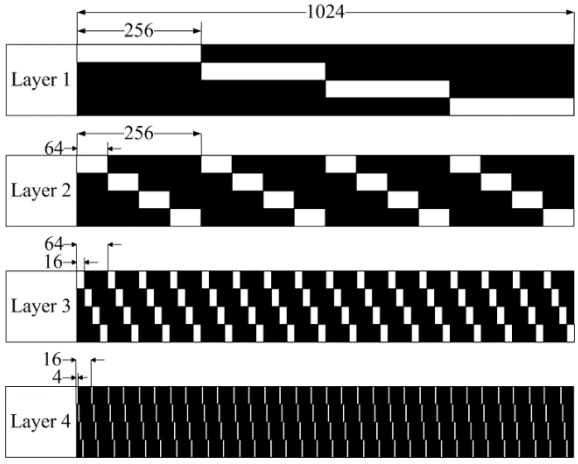


FIGURE 3. The four layers of the HOC pattern along with the stripe width of each layer and the hierarchical sub-division of each stripe.

To overcome this trade-off, we propose a new approach to accurately detecting the projector and camera shades, especially, when the shades are mixed with surfaces with low reflectance, for a structured light 3D camera system. To this end, the proposed approach first identifies stripe boundaries legitimate in terms of the projected pattern sequence. Based on the list of legitimate boundaries identified, then, the gaps in pixel and boundary addresses are analyzed to define the respective projector and camera shades, possibly, in the presence of address juxtaposition. The proposed approach is capable of distinguishing the shade from object surfaces with low reflectance.

II. SHADE DETECTION WITH THE BOUNDARY INHERITANCE CODEC

In this paper, we adopt the boundary inheritance codec (BIC) [17] for the detection of projector and camera shades. Therefore, we briefly introduce BIC briefly in relation to the problem of projector and camera shade detection.

BIC generates a sequence of stripe patterns based on hierarchically orthogonal pattern coding or the Hierarchically Orthogonal Codec (HOC) [29]. HOC is configured with four layers of orthogonal patterns of growing precision, as shown in Fig. 3, where each HOC layer contains four patterns that are orthogonal to each other. The four patterns of a layer represent the four sub-codes of its upper layer. This results in 16 pattern frames to be projected for the 256 stripes at the fourth layer, with each stripe represented by a 16-bit code.

BIC roughly follows four steps: 1) Estimation of the stripe pattern boundaries for each HOC layer; 2) Fine-tuning of the estimated boundaries that are common among layers in such a way as to make them consistent. This fine-tuning is called the boundary inheritance process; 3) Hierarchical sub-division of the region of pattern projection into a number of coded

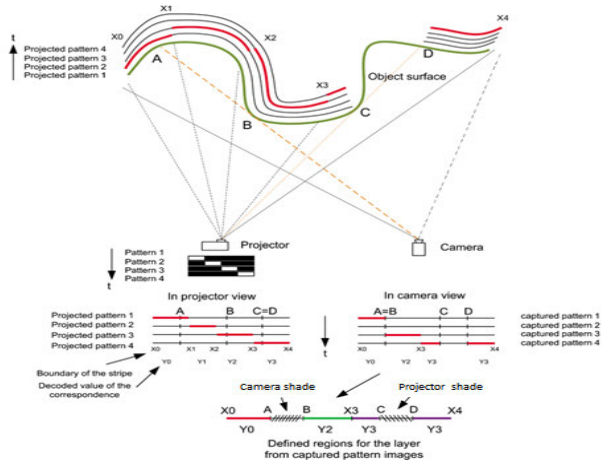


FIGURE 4. Illustration of camera and projector shades in relation to HOC patterns. The surface regions from A to B and from C to D represent, respectively, the camera and projector shades. The four stripe region boundaries are denoted x0, x1, x2, x3 and x4. Notice that region boundaries may be located within the camera shade and a stripe region may appear discontinuous from the camera due to the projector shade.

sub-regions or stripes and validation of their boundaries by the inheritance process; and 4) Outlier removal and application of between-boundary interpolation processes to obtain a high quality, anomaly-free depth estimation.

A. PROJECTOR AND CAMERA SHADES

As mentioned, it is difficult to differentiate the shades from the object surface when the shades are mixed with a surface with low light reflectance. Therefore, it is important to precisely detect the projector and camera shades in order to capture 3D data from the low-reflectance surface while minimizing outliers.

Fig. 4 illustrates the projector and camera shades in relation to the four stripe regions of HOC that are projected onto the surface. The camera and projector shades are illustrated by the surface regions from A to B and C to D, respectively, while the four stripe region boundaries are denoted x0, x1, x2, x3 and x4. As shown in the figure, the stripe region boundaries may be located within the camera shade or a stripe region may be discontinuous from the camera due to the projector shade. More specifically, in the case of the projector shade, shade C-D separates the stripe region, x3-x4, into two segments, even though the boundaries of the stripe pattern remain in sequence, such that any stripe boundaries that appear to be within the shade become outliers, possibly due to inter-reflections. In the case of the camera shade, shade A-B makes the stripe pattern boundaries in the region disappear from the camera image, such that the stripe images captured by the camera experience an abrupt jump in their addresses. The abrupt jump of stripe addresses may cause difficulties in address decoding as serious outliers are created by the unrecovered missing boundaries from the upper layers, which mess up the addresses of lower layers. Therefore, it is important for the quality of 3D imaging to identify whether or not the address sequences around the projector and camera shades are legitimate by precisely detecting the boundaries of the projector and camera shades. To explore how to accurately detect the boundaries of the camera and projector shades, we present various cases that are considered legitimate address sequences around the projector and camera shades in real situations.

B. STRIPE SEQUENCES IN RELATION TO SHADES

1) CASE-1: NO SHADE

Case 1 represents an ideal case in which no camera or projector shade appears during imaging. This situation can occur when the surface is continuous and smooth. As shown in Fig. 5 (a), the sequence of addresses assigned to stripe boundaries (address sequence) that appears in the camera image is exactly same as the one that is projected,

with no address jump or juxtaposition and no pixel jump or holes between neighboring addresses. A typical 4th layer signal of the image pixels observed from a real measurement for Case 1 is exemplified in Fig. 5(b), where the black, red, green and blue signals represent a sequence of four orthogonal stripes captured by the camera.

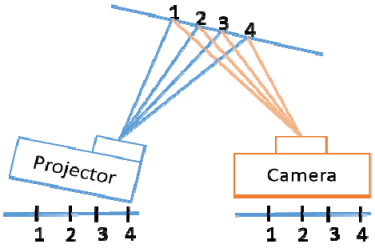
2) CASE-2: CONCAVE SHADE

Case 2 represents the projector and camera shades, marked in green and red, respectively, due to surface concavities that cause occlusions from the projector and camera views, as illustrated in Fig. 5(c). In this case, the address sequence in the camera image shows pixel jump (green) between neighboring stripe addresses due to the projector shade and address jump (red) due to the camera shade. A typical 4th layer signal of image pixels in a real measurement situation is shown in Fig. 5(d). The address jump due to the camera shade is illustrated in Fig. 5(d) by the disruption of the original black-red-green-blue sequence into the black-red-blue sequence with the green missing, while the pixel jump due to the projector shade is shown by the region with no clear stripe pattern, with no jump in the sequence of stripe addresses.

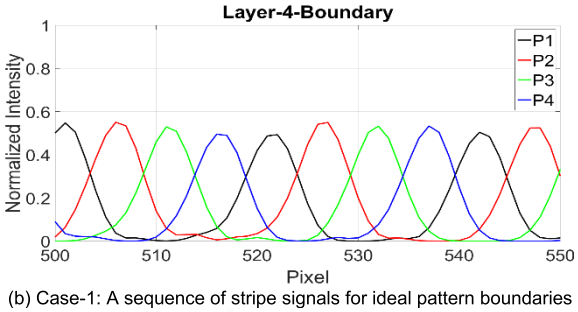
3) CASE-3: ISLAND SHADE

Case 3 represents the projector and camera shades, marked, respectively, in green and red in Fig. 5 (e), due to an obstacle isolated above the surface, referred to here as an island. In this case, the address sequence in the camera image shows not only address and pixel jumps between the neighboring addresses, as in Case 2, but also an address juxtaposition, as shown by the 1-4-3-5 sequence in Fig. 5 (e). Typical 3rd and 4th layer signals of image pixels in a real measurement situation are shown in Fig. 5(f).

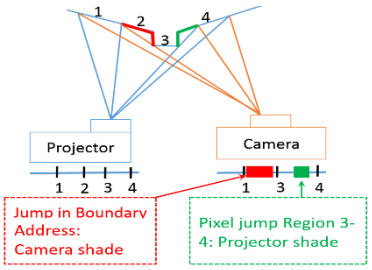
The address jump marked in red in Fig. 5(e) is indicated in Fig. 5 (f) by the missing green and blue signals in the 3rd layer sequence. On the other hand, the pixel jump marked in green in Fig. 5 (e) is indicated in Fig. 5(f) by the undistinguishable pattern signal between the red and the green signals of the 3rd layer sequence. Notice the discontinuous address



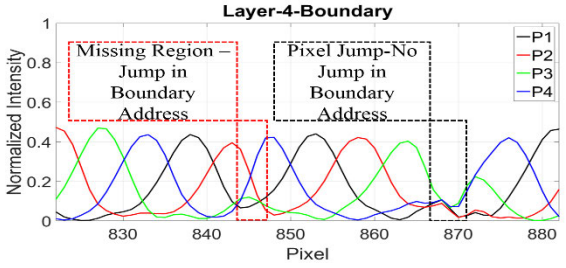
(a) Case-1: Ideal pattern boundary scenario without shades



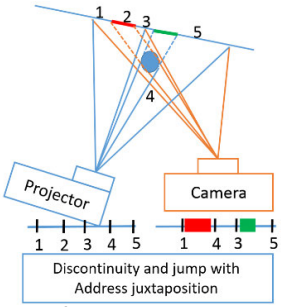
(b) Case-1: A sequence of stripe signals for ideal pattern boundaries



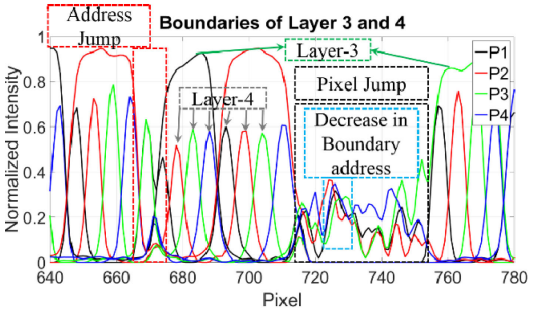
(c) Case-2: The projector(green) and camera(red) shades that cause the respective pixel and address jumps.



(d) Case-2: A sequence of stripe signals with pixel and address jumps in the presence of the projector and camera shades.



(e) Case-3: An island type of shade that causes discontinuity in jump due to boundary juxtaposition.



(f) Case-3: A sequence of stripe signals with discontinuity in jump due to address juxtaposition under the presence of an island type of shades.

FIGURE 5. Different scenarios of projector and camera shades with the respective pixel and address jumps.

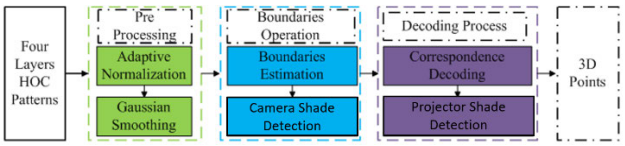


FIGURE 6. The flow diagram of the proposed approach to a structured light 3D camera system based on detection of the projector and camera shades.

jump due to the address juxtaposition between the green and blue of the 4th layer sequence that takes place in the region of the 3rd layer pixel jump.

III. PROPOSED METHOD

We address the problem of identifying the boundaries of the projector and camera shades in structured light depth estimation by detecting the address and pixel jumps in con- junction with the concave and island cases described in Section II. The proposed approach differs from conventional approaches that rely mainly on signal strength to detect the boundaries of shades, without taking the address and pixel jumps into consideration. The overall process of the proposed shade detection algorithm is schematically illustrated in Fig. 6. The algorithm comprises three major steps: 1) Signal pre-processing for normalization and smoothing, 2) Accurate boundary estimation and correction and 3) Detection of pixel and address jumps along with address decoding. Signal normalization and smoothing aims at representing the captured stripe pattern signals independently of the ambient light, the strength of surface reflection and the variation in stripe widths. Accurate boundary estimation and correction aims at accurately detecting stripe pattern boundaries based on the intersection of normalized neighboring stripes and the correction of the layer-wise overlapping boundaries for consistency. The algorithm is implemented based on six camera frames: four frames required to define the fourth layer of the HOC patterns and two additional frames that capture an ambient image (L) and an all-white image (H).

A. SIGNAL NORMALIZATION AND SMOOTHING

$$f_w(x) - f_a(x) = [(\mathrm{H} - \mathrm{L}) \otimes g_p(x, \sigma_P))^* \mathrm{R}_{(\mathrm{X})}] \otimes$$

$$\begin{aligned} & \times g_c(x, \sigma_c) W_a(x) \\ f_s(x) - f_a = & [((s(x) - L) \otimes g_p(x, \sigma_P))^* \mathrm{R}_{(\mathrm{X})}] \otimes \\ & \times g_c(x, \sigma_c) W_b(x) \end{aligned} \quad (1)$$

The signal normalization and smoothing process transforms the captured stripe pattern signal into a canonical representation independent of the ambient light, the strength of surface reflection and the variation in stripe widths. The process of transforming the captured stripe pattern into the canonical representation is described in detail in the following.

Let s(x) be a virtual stripe signal with a boundary at x as an ideal step function:

$$s(x) = \begin{cases} H \; x \geq 0 \\ L \; x < 0 \end{cases} \quad (2)$$

In practice, s(x) is subject to corruption and deformation along the entire optical path [30], as follows:

- The ideal signal, s(x), will be blurred along the optical path of the projector lens system.
- The ideal signal will be attenuated by the reflection, R(x), from the object surface.
- The signal reflected by the surface will be contaminated by the ambient light, A(x).
- The signal will be blurred due to optical passage through the camera lens system to the imaging sensors.
- The signal captured by the imaging sensors will be contaminated by the measurement noise, W(x).

Taking all the above factors that deform and corrupt s(x) into consideration, the captured signal of the pattern received at the camera sensor for surface x, fs(x), can be modeled by:

$$\begin{aligned} f_s(x) = & \Big(\big((s_{(x)}) \otimes g_p(x, \sigma_p) \big) R_{(x)} + A(x) \Big) \otimes \\ & \times (g_c(x, \sigma_c)) + W(x) \end{aligned} \quad (3)$$

Here, we represent the attenuation upon reflection at the local surface corresponding to x by the reflectance index $R(x)$. The projector and camera blurring kernels are represented as $g_p\left(x, \sigma_p\right)$ and $g_c\left(x, \sigma_c\right)$, respectively, where

$$g_j\left(x, \sigma_j\right) = \frac{1}{\sigma_j \sqrt{2\pi}} e^{\left(\frac{x^2}{2\sigma_j^2}\right)} \quad (4)$$

with 'j' representing either 'p' or 'c' and ⊗ representing the convolution process.

One thing to note is that the stripe width becomes narrower as the layer moves down in the pattern hierarchy, causing a decrease in pixel intensity. As such, the stripe boundaries may not be detected once the stripe width is reduced beyond a certain level. This is illustrated in Fig. 7, where the signal intensities decrease as the signals move from layer k to layer k + 1 (k = 1,2,3) due to the reduction in the stripe width. For instance, at pixel 305, the intensities of the signals in layers 1, 2, 3 and 4 are 120, 110, 100 and 65, respectively, as the width

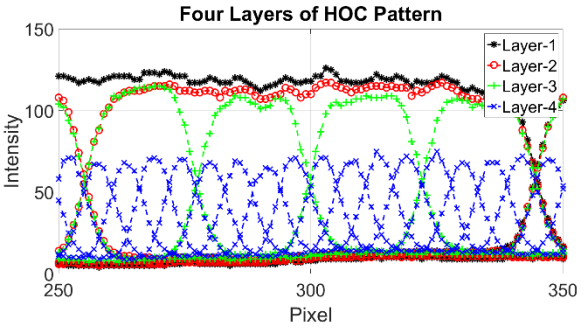


FIGURE 7. Pixel intensity of a selected row of the 4 layers of the HOC patterns.

of the stripes is gradually reduced to a quarter of the original width. To consider the reduction in light intensity due to the decrease in stripe width, or the effect of stripe width on the strength of the captured signal, $f_s(x)$, we introduce the stripe width factor α. The final captured signal at the camera sensor for surface x, $f'_s(x)$, becomes

$$f'_s(x) \approx (1 - \alpha) * f_s(x) \quad (5)$$

We propose that the stripe width factor, α, for the i^{th} layer, L_i, can be determined by the following formula:

$$\alpha = \frac{Slope\ L_i}{Slope\ L_1} \quad (6)$$

where

$$Slope = \frac{Signal\ Intensity}{Shutter\ Time} \quad (7)$$

We validate Eqs. (5) and (7) extensively based on experiments with a variety of ambient conditions, reflection strengths and shutter speeds.

Now, let us consider how to recover, $s(x)$, so as to find the ideal boundary at x from the captured signal, $f'_s(x)$. Note that the recovered $s(x)$ is supposed to be independent of the blurring along the optical path, the ambient light, the strength in surface reflection and the variation in stripe widths. To this end, we first capture the all-bright signal, $f_w(x)$ with s(x) = H, and the ambient signal, $f_a(x)$ with s(x) = L, as follows:

$$\begin{aligned} f_w(x) = & \left(\left(H \otimes g_p\left(x, \sigma_p\right) \right) * R(x) + A(x) \right) \otimes \\ & \times g_c(x, \sigma_c) + W_1(x)) \end{aligned} \quad (8)$$

$$\begin{aligned} f_a(x) = & \left(\left(L \otimes g_p\left(x, \sigma_p\right) \right) * R(x) + A(x) \right) \otimes \\ & \times g_c(x, \sigma_c) + W_2(x)) \end{aligned} \quad (9)$$

where α is set to 0 for the all-bright and ambient frames.

Then, based on $f_w(x) - f_a(x)$ and $f_s(x) - f_a(x)$ from Eqs. (2), (8) and (9), we can derive $s(x)$ with the boundary represented as a step jump at x :

$$\begin{aligned} s(x) - L = & \frac{Deconvolution\left(\left(f_s(x) - f_a(x) \right), \left(\sigma_p + \sigma_c \right) \right)}{Deconvolution\left(\left(f_w(x) - f_a(x) \right), \left(\sigma_p + \sigma_c \right) \right)} \\ & \times (H - L) + W_s(x) \end{aligned} \quad (10)$$

Note that L can be set to 0 without loss of generality and "deconvolution" can be computed based on the Richardson-Lucy [31], [32] deconvolution operator. In Eq. (9), $f_s(x)$ can be replaced by the captured signal, $f_s'(x)$, that takes the stripe width factor into consideration based on Eq. (5).

In practice, the deconvolution operation tends to amplify the noise effect and, thus, is best avoided, if possible. Therefore, to detect the boundary at x accurately but without resorting to the deconvolution operation, we localize the boundary at the intersection of a pair of inversely shaped stripes with a shared boundary at x. More specifically, instead of using a single jump-up stripe, $s(x)$, we use the intersection of two consecutive stripes, one jump-up with $s(x)$ and the other jump-down with $-s(x)$. In this case, the blurring along the optical path is expected to occur equally to the two stripe signals around x such that the intersection can accurately represent the location of x given appropriate compensation for the effects of ambient light, the strength of surface reflection and the variation in stripe width. The process of compensating the captured signal, $f_s'(x)$, for the effects of ambient light, the strength of surface reflection and the variation in stripe width is referred to here as signal normalization. The normalized signal, $\bar{f}_s(x)$, can be obtained from Eqs. (5) and (9) without the deconvolution operation, as follows:

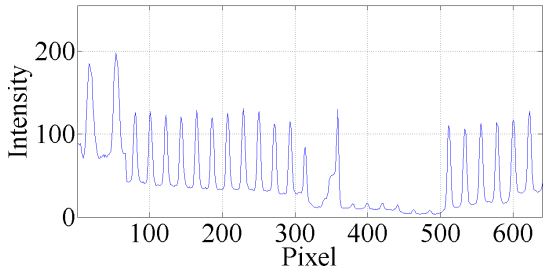
$$\bar{f}_s(x) = \frac{f_s'(x)/(1-\alpha) - f_a(x)}{f_w(x) - f_a(x)} H + W_s(x) \qquad (11)$$

where L is set to 0. Note that $f_w(x) - f_a(x)$ and $(1-\alpha)$ are used to compensate for the reflection and the stripe width variations, respectively. Fig. 8 illustrates the transformation of a captured signal into a normalized signal based on Eq. (11).

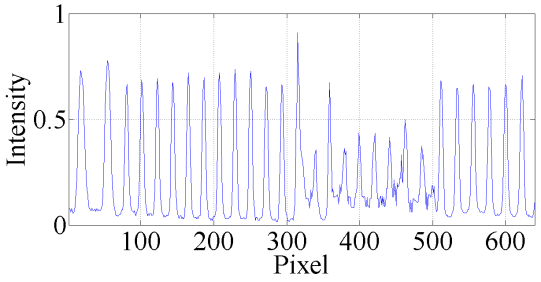
The normalized signal still contains the measurement noise, $W_s(x)$ of Eq. (11), in the imaging pixels. The effect of measurement noise on the captured signal is illustrated by the non-smooth curves in Fig. 10(a). To smooth out the measurement noise, we apply a Gaussian filter to the normalized signal as the final step in signal preprocessing. Fig. 10 (b) illustrates the Gaussian smoothed signal of the normalized signal shown in Fig 10(a).

The normalization and smoothing process improves the signal conditions for accurate boundary localization based on the intersection of two consecutive stripe signals that share the boundary, as shown in Figs. 10 (b) and 9(b), where Fig. 9 emphasizes the effect of compensating for the stripe width factor α.

The overall effect of the normalization and smoothing process can be demonstrated by the real-world experimental result depicted in Fig. 11. The boundaries of the pattern projected on a dark surface have more contrast after normalization and smoothing, compared to the original captured image.

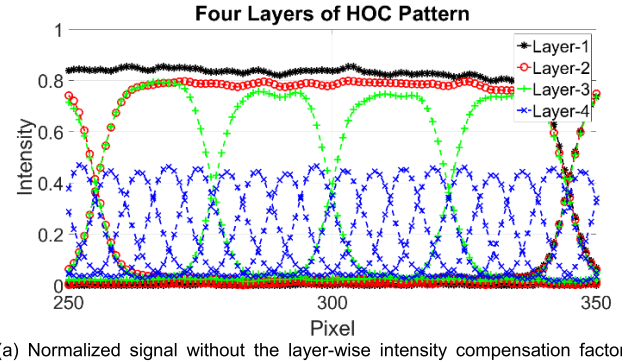


(a) Pixel intensity of a selected row before normalization

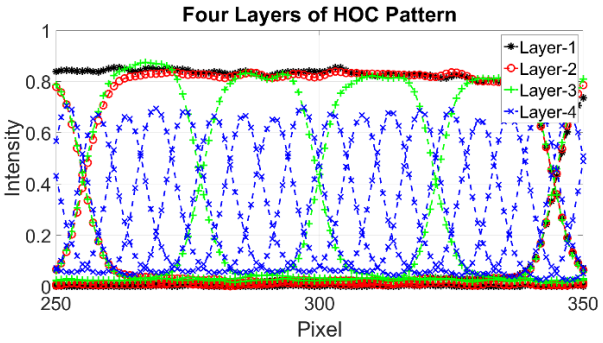


(b) Pixel intensity of a selected row after normalizationtion

FIGURE 8. Pixel intensity of a selected row of captured camera frame before and after signal normalization.



(a) Normalized signal without the layer-wise intensity compensation factor, alpha



(b) Normalized signal with the layer-wise intensity compensation factor, alpha

FIGURE 9. Normalized signal of a selected row with and without the layer-wise intensity compensation factor, alpha. (a) Without alpha, the normalized signal intensity differs among layers with the weak 4th layer signal (b) With alpha, the difference in normalized signal intensity is reduced, which helps in detecting the stripe boundaries.

B. BOUNDARY DETECTION AND LOCALIZATION

Boundary detection identifies valid stripe boundaries in the presence of projector and camera shades. Boundary localization determines the position of detected valid boundaries in subpixel precision by interpolation while making layer-wise common boundaries consistent when applicable. In Fig. 12, four stripes, P1 (black), P2 (red), P3 (green) and P4 (blue),

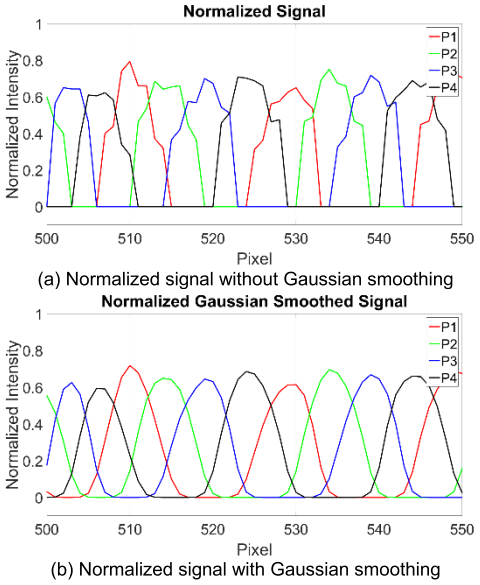


FIGURE 10. Stripe signals of a selected row of the HOC fourth layer before and after applying Gaussian smoothing to normalized signals.

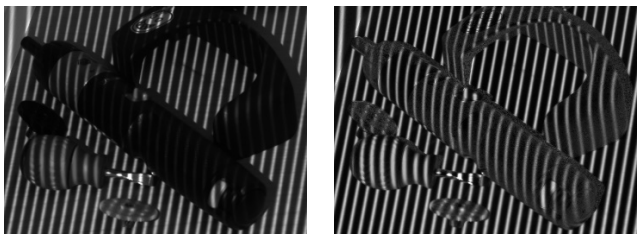


FIGURE 11. Stripe pattern signals of the fourth layer of HOC before and after signal normalization and smoothing. (a) Before normalization and smoothing (b) After normalization and smoothing. Notice that the stripe patterns on a dark surface have more contrast after normalization and smoothing.

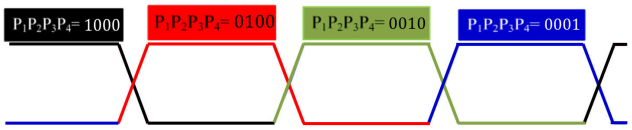


FIGURE 12. Stripe boundaries of HOC patterns from four S-SI pairs, P_1-P_2, P_2-P_3, P_3-P_4 and P_4-P_1.

are illustrated according to their projection sequence. The stripe boundaries are shown at the intersection or crossing of the falling and rising edges of a pair of consecutive pattern stripes, referred to here as a Signal-Signal Inverse (S-SI) pair. Following the sequence of pattern projection, we define four S-SI pairs, P1- P2, P2-P3, P3-P4 and P4-P1, and the corresponding boundaries, as shown in Fig. 12.

Boundary detection intends to distinguish legitimate boundaries from false boundaries that may appear in the shade due to scattering and inter-reflection [30]. We propose that the legitimacy of a boundary be checked by the correctness in S-SI pairing and its edge formation by which the

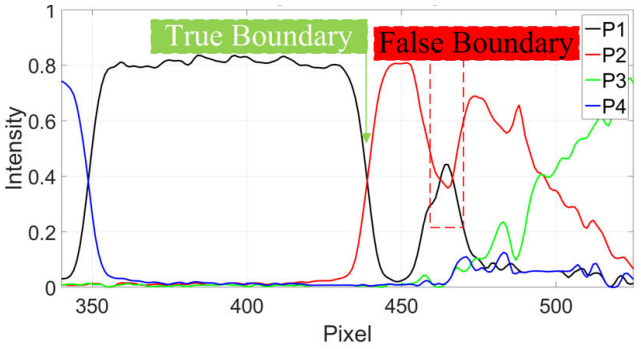


FIGURE 13. False boundaries that are appeared in a shade due to inter-reflection are contrasted with true boundaries that satisfy clear boundary criteria.

boundary is derived and the consistency among layer-wise common boundaries and in address sequencing. We refer boundaries that satisfy the above first and second criteria as ''clear and consistent'' boundaries, respectively.

1) CLEAR BOUNDARIES

A clear boundary is defined as an intersection of the falling and rising edges of S-SI pairs, P_1- P_2, P_2 - P_3, P_3 - P_4 and P_4 - P_1. S-SI pairs are represented by the normalized and smoothed stripe signals described in Section III. In addition, to be qualified as a clear boundary, an intersection should have a support region around it within which the S-SI pair continues maintaining its falling and rising edge status. Detecting clear boundaries based on legitimacy in S-SI pairs represents an important first step for discriminating a shade from a surface with low reflectance. For instance, Fig. 13 illustrates the case where false boundaries, marked in red, are present in a shade due to inter-reflection. The true boundary marked in green is a clear boundary as it meets the legitimacy in S-SI pairing (P_1-P_2) with a definite support region around it. On the other hand, the false boundaries marked in red violate the clear boundary criteria with an incorrect S-SI pairing (P_2-P_1) or an insufficient support region. Note that the false boundaries may not be assigned a legitimate address in terms of valid address sequencing due to address duplication.

Algorithm 1 presents a detailed process for detecting clear and consistent boundaries. For detecting a clear boundary, first, an intersection formed by a legitimate S-SI pair is identified, starting from the first to the last pixel of each row. Once such an intersection is found, we check if the intersection is supported by a support region with a sufficient length of the falling and rising edges. Specifically, we check if the intensity of S or SI is greater than the rest stripe signals on the left or right side of the intersection within the support region. The size of a support region may depend upon the camera pixel resolution and the layer in which clear boundaries are searched for. The higher the resolution or the layer, the larger the size. Intersections that satisfy the above S-SI pairing and support region criteria are declared as clear boundaries.

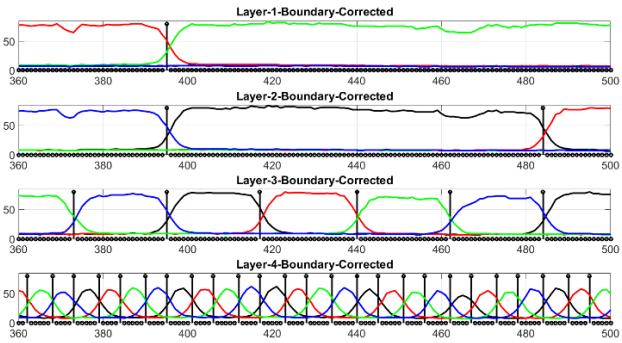


FIGURE 14. Illustration of layer-wise common stripe boundaries in the HOC patterns. Such layer-wise common boundaries should be maintained as consistent in terms of their locations.

2) LAYER-WISE CONSISTENCY

A clear boundary is subject to further validation for consistency among layer-wise common boundaries and in address sequencing. Note that, due to a hierarchical structure of stripe patterns in the HOC, a boundary located at the intersection of P_4-P_1 (Blue-Black) in a layer shares common boundaries with its upper layers, as shown in Fig. 14.

For a clear boundary to be consistent, first, its layer-wise common boundaries should be all clear, in case the clear boundary has common boundaries in its upper layers. In addition, a legitimate address can be assigned to a clear boundary based on layer-wise stripe codes at the boundary location, where the assigned address represents no anomaly in address sequencing such as address duplication.

Refer to Algorithm 1 for more details of identifying clear and consistent boundaries. Algorithm 1 outputs a set of clear and consistent boundaries that are detected and localized with their valid addresses assigned based on layer-wise stripe codes. A clear and consistent boundary supported by the layer-wise common boundaries that are clear and the valid boundary address assigned helps enhance the reliability in boundary detection and localization. For instance, validating layer-wise consistency can compensate for the clear boundary criteria in identifying illegitimate boundaries in a shade by picking up those inter-reflected boundaries in a shade that the clear boundary criteria somehow fail to detect. The set of clear and consistent boundaries obtained by Algorithm 1 serves as the basis for the detection of projector and camera shades, as explained in the following subsections.

C. DETECTION OF PROJECTOR SHADES

A projector shade represents those regions that the projected stripe patterns cannot reach but that a camera can view them. As described in Section II.B, a projector shade incurs a jump in the pixel gap between the neighboring boundaries. As an example, Fig. 15 illustrates the 4^{th} layer stripe signals when a projector shade partially overlaps with a surface with low reflectance or a black surface. The pixel region from

Algorithm 1 Detection of Clear and Consistent Boundaries for HOC Based Structured Light Imaging

Input:

1) Image pixel coordinate (u, v), u = 1,…,h, v = 1, …, w.
2) HOC layer number L, L = 1, 2, 3, 4.
3) Normalized stripe signals, ^{L}S or ^{L}SI, in the layer L, where ^{L}S or ^{L}SI ε (P1, P2, P3, P4) with the projection sequence of P1, P2, P3, P4.
4) Correct stripe signal pairs, S-SI, by which boundaries are defined: (S, SI) ε (P1, P2), (P2, P3), (P3, P4), (P4, P1).
5) The size of a supporting region, q.

Output:

1) A set of the 4th layer clear and consistent boundaries, $\{C_k\}$, k = 1, …, n_c, where $\{C_k\} = \{(u, v')_k, (S, SI)_k, A_k\}$ with v' and A_k represent, respectively, the pixel position and boundary address of the k^{th} boundary. The pixel position, v', is in subpixel precision and A_k is with the 4 layer stripe codes: $A_k = (^1S, ^2S, ^3S, ^4S)_k$.

Method:

Step 1. (Initialization) u = 1, v = 1 and k = 1
Step 2. At v, obtain S(u,v) and SI(u,v) from the normalized signal strengths of the 4^{th} layer signal.
Step 3. (Clear boundary) Check if S(u,v-1) ≥ SI(u,v-1) and SI(u,v + 1) ≥ S(u,v + 1). If yes, then, in the support region, check if S(u, v-i) and SI(u, v + i) are greater than all the rest stripes for i = 1,2,…, q. If yes, save (u, v) as a candidate of clear boundary and go to Step 4. Otherwise, go to Step 8.
Step 4. (Sub-pixel precision) Refine (u,v) to (u,v') in sub-pixel precision by intersecting S and SI signals in the support region, {S(u,v-i), …,S(u,v + i)} and {SI(u,v-i), …, SI(u, v + i)} for i = 0, 1,2,…, q. Go to Step 5.
Step 5. For (u, v') with (S, SI), if (S, SI) = (P4, P1), go to Step 6. Otherwise, if (S, SI) ≠ (P4, P1), go to Step 7.
Step 6. (Consistency) Check if the layer-wise common boundaries of (u, v') are clear based on the process of Step 3. If yes, make the pixel position of upper layer common boundaries of (u, v') same as (u, v'). go to Step 7. Otherwise, go to Step 8.
Step 7. (Addressing) Define the address, A, of (u, v') by obtaining its upper layer stripe codes: ^{3}S, ^{2}S and ^{1}S. Check if A is free of anomalies such as an unsupportive duplication or outlier. If yes, save (u, v'), (S, SI) and A as the k^{th} element in the set of clear and consistent boundaries, $\{C_k\}$. Set k = k + 1 and go to Step 8. Otherwise, go to Step 8.
Step 8. if v < w, set v = v + 1 and go to Step 2. If v = w, go to Step 9.
Step 9. If u < h, set u = u + 1 and go to Step 1. If u = h, output the set of clear and consistent boundaries $\{C_k\}$.

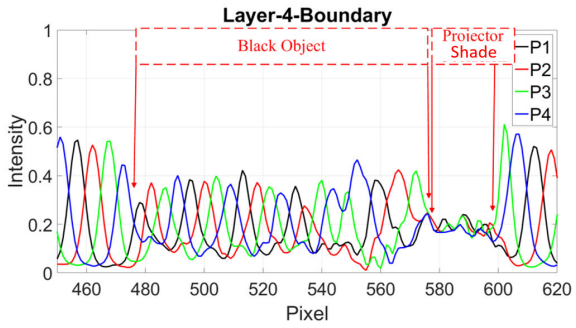


FIGURE 15. Illustration of the stripe pattern signals appeared on a projector shade and on a surface with low reflectance. Notice that pattern signals on the projector shade fail in satisfying the clear and consistent boundary criteria unlike those on the low reflectance surface.

475-575 belongs to a black surface, while the region from 576-600 to a projector shade. As shown in Fig. 15, the stripe signals that appear on the black surface generate clear and consistent boundaries in a legitimate stripe pattern sequence. However, in the projector shade, although a number of signal crossings exist due to scattering and inter-reflection, they do not meet the criteria for clear and consistent boundaries described in Algorithm 1. In other words, a pixel jump takes place. This represents an important feature for distinguishing a surface with low reflectance from a projector shade by exposing a pixel jump that crosses over the projector shade.

For the detection of a projector shade, we examine if there occurs any pixel jump between a neighboring boundary pair based on the set of clear and consistent boundaries generated by Algorithm 1. A pixel jump is detected by identifying an excessive pixel gap between a neighboring boundary pair in comparison with the pixel gaps of adjacent boundary pairs. As described in Section II. B, a pixel jump between a pair of consecutive boundary addresses with no address jump represents a projector shade. Note, however, that we need to take into account an address juxtaposition that may occur when a projector shade is associated with an island shade. With such an address juxtaposition, a pixel jump between a neighboring boundary pair with an address jump can also be a projector shade. To deal with this issue, we reorder a set of clear and consistent boundaries to follow the sequence in boundary addresses, so as to find a consecutive address pair with a pixel jump between the pair based on the reordered set. Then, any boundary pairs the pixel gaps of which reside inside the pixel jump from the reordered set represent an address juxtaposition. Finally, a projector shade is identified by removing the pixel gaps inside the pixel jump while adjusting the address gap to represent the corrected pixel jump. Refer to Algorithm 2 for more details of how to identify a projector shade with and without the presence of address juxtaposition.

D. DETECTION OF CAMERA SHADES

A camera shade represents those regions that projected stripe patterns can reach but a camera cannot view them.

Algorithm 2 Detection of Projector and Camera Shades for the HOC Based Structured Light Imaging

Input:

1) A set of the 4th layer clear and consistent boundaries, $\{C_k\} = \{(u, v')_k, (S, SI)_k, A_k\}$, $k = 1,.., n_c$, with the address, A_k, $A_k = ({}^1S, {}^2S, {}^3S, {}^4S)_k$, where ${}^LS \in (P_1, P_2, P_3, P_4)$ at layer L.
2) α: pixel jump threshold, β: address jump threshold

Output:

1) A set of camera pixel jumps representing projector shades: $\{P_i\} = \{u, (v_1', v_2')_i, (A_1, A_2)_i\}$, $i = 1, \ldots, n_p$.
2) A set of stripe address jumps representing camera shades: $\{Q_j\} = \{u, (v_1', v_2')_j, (A_1, A_2)_j, j = 1, \ldots, n_q$
   Note: A_1 and A_2 represent the boundary addresses for v_1' and v_2', respectively.

Method:

Step 1. (Reordering of $\{C_k\}$) To deal with address juxtaposition, obtain $\{C'_j\}$, $j = 1,\ldots, n_c$, by reordering $\{C_k\}$ in the boundary address sequence, instead of the pixel sequence. $\{C'_j\}$ is same as $\{C_k\}$ in case of no address juxtaposition.
Step 2. (Initialization) Set $k = 1$
Step 3. For a neighboring boundary pair, (C_k, C_{k+1}), obtain its pixel and address gaps, (v'_k, v'_{k+1}) and (A_k, A_{k+1}), respectively. Check if $(v'_{k+1} - v'_k) > \alpha\, g_{max}$ (v'_k, v'_{k+1}) for a pixel jump, where g_{max} represents the max. pixel gap among m boundary pairs adjacent to (C_k, C_{k+1}). If no pixel jump detected at (v'_k, v'_{k+1}), go to Step 4. Otherwise, if a pixel jump detected, go to Step 7.
Step 4. (Shade-free) If (A_k, A_{k+1}) is non-consecutive with an address jump, go to Step 5. Otherwise, if (A_k, A_{k+1}) is consecutive, put (C_k, C_{k+1}) in the set of shade free pairs. Set $k = k + 1$ and go to Step 3, if $k < n_c$. Stop, if $k = n_c$.
Step 5. (Camera shade) Find C'_j and C'_{j+r} in $\{C'_j\}$ such that $A_j = A_k$ and $A_{j+r} = A_{k+1}$. If C'_j and C'_{j+r} are a neighboring boundary pair in $\{C'_j\}$, i.e., $r = 1$, put (A_k, A_{k+1}) and (v'_k, v'_{k+1}) in the set of camera shades, $\{Q_j\}$, for (A_1, A_2) and (v_1', v_2'), respectively. Set $k = k + 1$ and go to Step 3, if $k < n_c$. Stop, if $k = n_c$. Otherwise, if $r > 1$, i.e., there are boundaries addressed in between C'_j and C'_{j+r}, go to Step 6.
Step 6. (Juxtaposition) Pick up the boundary addresses, $A_k < A'_{j+1}.. A'_{j+r-1} < A_{k+1}$, in between C'_j and C'_{j+r} and find their corresponding pixel locations, $v_{j+1}', \ldots, v_{j+r-1}'$, in $\{C_k\}$. Verify that $v_{j+1}', \ldots, v_{j+r-1}'$ are outside of (v'_k, v'_{k+1}) as the result of juxtaposition. Put (A_k, A'_{j+1}) and (v'_k, v'_{k+1}), as well as (A'_{j+r-1}, A_{k+1}) and (v'_k, v'_{k+1}), in the set of camera shades, $\{Q_j\}$, for (A_1, A_2) and (v_1', v_2'), respectively. Set $k = k + 1$ and go to Step 3, if $k < n_c$. Stop, if $k = n_c$.

Algorithm 2 *(Continued:)* Detection of Projector and Camera Shades for the HOC Based Structured Light Imaging

Step 7. (Projector shade) If (A_k, A_{k+1}) is non-consecutive with an address jump, go to Step 8. Otherwise, if (A_k, A_{k+1}) is consecutive, put (v'_k, v'_{k+1}) and (A_k, A_{k+1}) in the set of projector shades, $\{P_i\}$, for (v_1', v_2') and (A_1, A_2), respectively. Set $k = k + 1$ and go to Step 3, if $k < n_c$. Stop, if $k = n_c$.

Step 8. (Juxtaposition) Find C'_j and C'_{j+r} in $\{C'_j\}$ such that $A_j = A_k$ and $A_{j+r} = A_{k+1}$. Pick up the boundary addresses, $A_k < A'_{j+1}.. A'_{j+r-1} < A_{k+1}$, between C'_j and C'_{j+r}. Verify that A'_{j+r-1} or A'_{j+1} is consecutive to A_k or A_{k+1} such that the addresses included in (A'_{j+r-1}, A_{k+1}) or (A_k, A'_{j+1}) in $\{C_k\}$ represent a juxtaposition. Put (v'_k, v'_{k+1}) and (A_k, A_{k+1}) in the set of projector shades, $\{P_i\}$, for (v_1', v_2') and (A_1, A_2), respectively. Set $k = k + 1$ and go to Step 3, if $k < n_c$. Stop, if $k = n_c$.

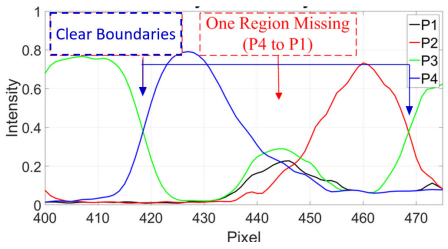


FIGURE 16. Illustration of a missing region due to a camera shade: no clear and consistent boundaries are present in between the boundaries of (P_3-P_4) and (P_2-P_3) with stripe pattern P1 (Black) is completely disappeared.

As described in Section II.B, a camera shade causes a jump in the boundary address. Note, however, that a camera shade is not exposed explicitly in the camera image as no definite pixel gap is incurred by a camera shade. For instance, Fig. 16 illustrates the case where the black stripe is disappeared from the camera image, causing an address jump without a definite pixel jump. Specifically, two stripe boundaries, one between P_4 and P_1 and another between P_1 and P_2, are disqualified as clear and consistent boundaries due to the appearance of an incorrect stripe signal, P_3, between the two clear boundaries, P_3-P_4 and P_2-P_3. Recall that a clear and consistent boundary should have a correct S-SI pairing with the intensities of their falling and rising edges greater than the rest stripe signals within a support region.

A camera shade can be detected by identifying an address jump from the set of clear and consistent boundaries generated by Algorithm 1. We can declare a camera shade for an address jump between a pair of neighboring boundaries with no pixel jump. Note, however, that, same as the case of a projector shade, we need to take into account an address juxtaposition that may occur when a camera shade is involved in an island shade. With an island shade, an address jump between a neighboring boundary pair can partly be disturbed by an address juxtaposition. In this case, we check the set of clear and consistent boundaries to find the boundaries the addresses of which reside inside the address jump under consideration by a juxtaposition. A camera shade is then defined by removing out those addresses present inside an address jump due to a juxtaposition. For more details on detecting a camera shade, refer to Algorithm 2. Note that pixel and address jumps other than the camera and projector shades detected by Algorithm 2 are regarded as surfaces with failed pattern recognition due to extremely low or specular surface reflectance.

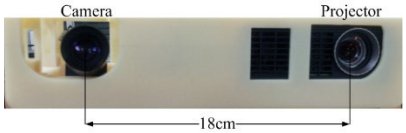


FIGURE 17. The configuration of a structured light 3D camera used for experiments.

E. REMOVAL OF OUTLIERS IN 3D RECONSTRUCTION

The proposed approach to projector and camera shade detection differs from the conventional approaches in that not only it presents a notion of clear and consistent boundaries in terms of the legitimacy in stripe sequences as a means of effectively distinguishing a shade from a surface with low reflectance. But also, it deals with an island shade that produces an address juxtaposition that complicates the detection of projector and camera shades. The proposed approach offers capability of capturing a high quality of 3D point clouds with much less outliers than conventional approaches. This is because incorrectly picked up boundaries in the projector and camera shades cause outliers in the reconstruction of a 3D point cloud. Note that 3D reconstruction is based on pairs of the projector and camera pixel addresses corresponding to the same boundaries. The proposed approach to camera and projector shade detection effectively removes out false boundaries caused by the shades mixed with a surface with low reflectance, as shown in more detail in the following experimental results.

IV. EXPERIMENTAL RESULTS

A. EXPERIMENTAL SETUP

The following equipment was used to test the proposed algorithm.

- Embedded Camera: Point Grey Flea3 FireWire
- Commercial Projector: Optoma ML 750
- Camera Lens: 12 mm 1:1.3 TV Lens
- Baseline: 18 cm
- Projector Resolution: 1024 × 768 pixels
- Camera Resolution: 640 × 480 pixels
- Projected Patterns: HOC, GCI, BCI, GCI (LS/PS), GC(LS/PS)
- Pattern Resolution: 1024 × 768 pixels

Fig. 17 shows the structured light 3D camera system developed to evaluate the proposed algorithm. Patterns are

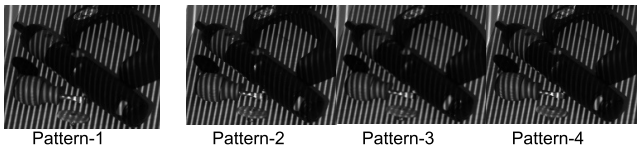


FIGURE 18. 4 stripe signals, P_1, P_2, P_3, and P_4, captured in sequence for the 4th layer HOC patterns before normalization.

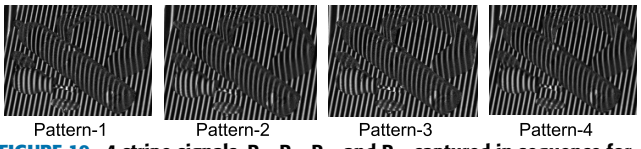


FIGURE 19. 4 stripe signals, P_1, P_2, P_3, and P_4, captured in sequence for the 4th layer HOC patterns after normalization.

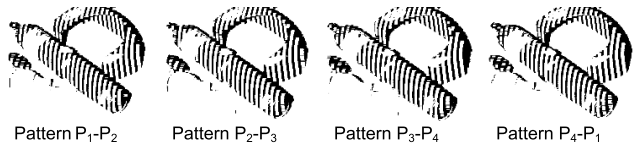


FIGURE 20. Images of four S-SI pairing to detect clear boundaries.

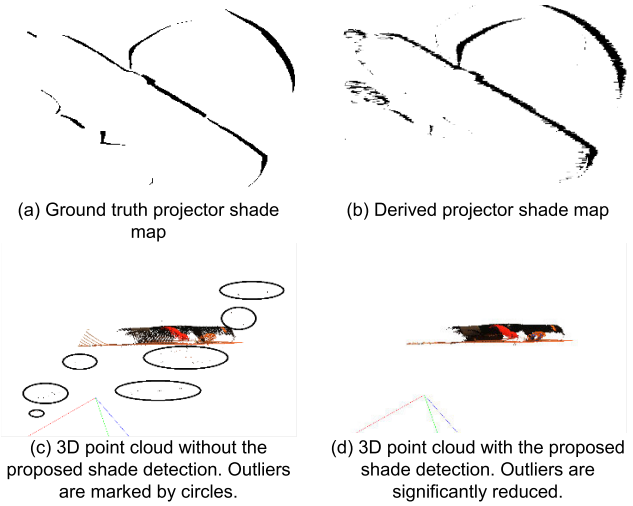


FIGURE 21. A projector shade map obtained by the proposed shade detection in comparison with the ground truth: (a) Ground truth shade map (b) Derived projector shade map (c) 3D point cloud by without proposed shade detection (d) 3D point cloud with the proposed shade detection Notice that, in (c), a significant number of outliers are generated due to inter-reflected stripe boundaries in the shade.

projected through the computer, and the camera is connected to the same computer to receive the captured frames for 3D reconstruction.

B. PROJECTOR SHADE DETECTION

The output of each step of projector shade detection is presented here to enable visualization of the results. Fig. 18 shows the fourth layer images captured by individually projecting P_1, P_2, P_3, and P_4 stripes, onto an industrial object with a dark surface.

After signal normalization, the captured images of Fig. 18 are transformed into the normalized images shown in Fig. 19. Notice that, in the normalized images, the stripe patterns on the dark surface become much clearer than those in the non-normalized images. This verifies that the proposed signal normalization process improves the signal conditions for boundary detection.

Fig. 20 illustrates the four S-SI pairings, P_1-P_2, P_2-P_3, P_3-P_4 and P_4-P_1, used to detect the clear boundaries.

The projector shade maps are then obtained from the individual S-SI pairings based on the detected clear boundaries after checking the legitimacy of the stripe boundary sequence with a possible pixel jump. By combining the shade maps from the four S-SI pairings, we obtain the final projector shade map, as illustrated in Fig. 21 (b). Fig. 21 (a) represents the ground truth projector shade map obtained by manually annotating the ground truth projector shade. Fig. 21 (a) and (b) provide a qualitative assessment of the performance of the proposed projector shade detection process and the ground truth shade for comparison. The details of the quantitative evaluation of the proposed method in comparison with the conventional method are given in Section IV. F. Fig. 21 (c) and (d) show a comparison of the two 3D point cloud outputs that are generated without and with, respectively, the proposed projector shade detection process for outlier removal. The comparison demonstrates that the proposed projector shade detection process significantly reduces the number of outliers in 3D reconstruction.

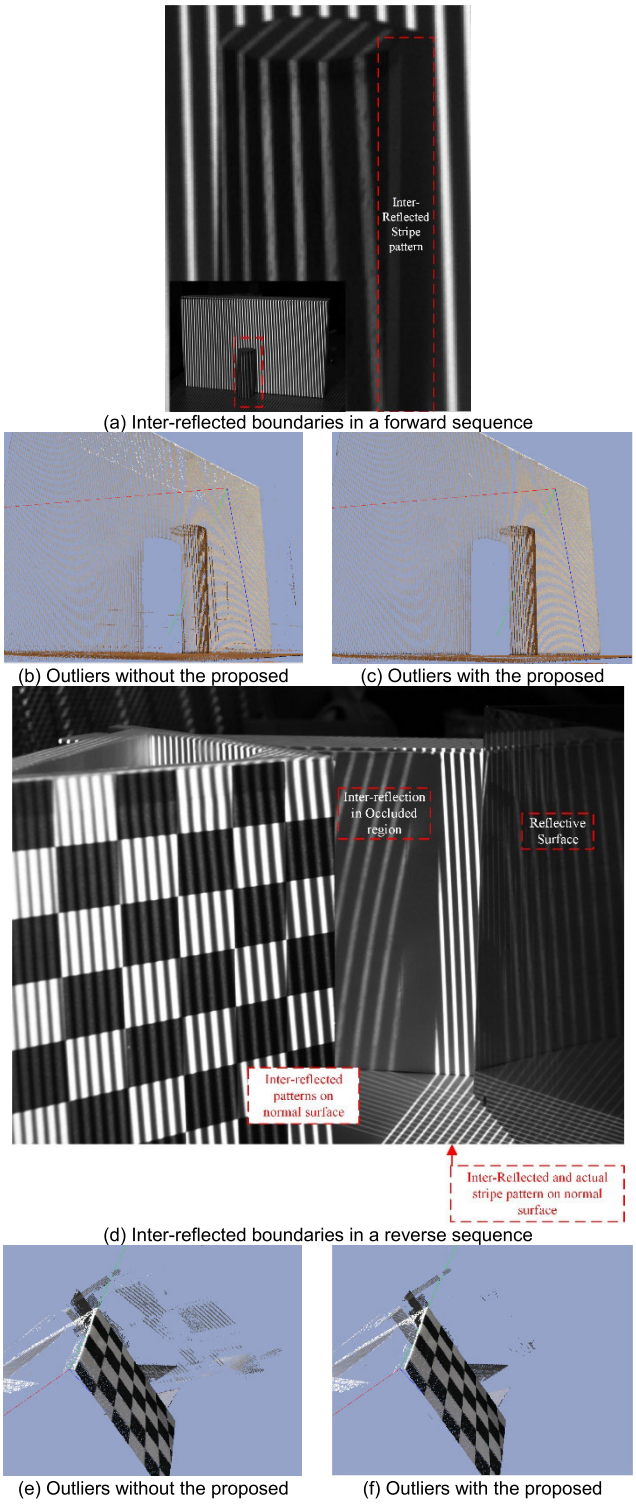
C. INTER-REFLECTION IN PROJECTOR SHADE

The outliers in Fig. 21 (c) are due to the appearance of clear stripe boundaries in the projector shade. These stripe boundaries are caused by inter-reflection, as illustrated in Fig. 22. Note that the inter-reflected boundaries can be sequenced either in the forward order or in the reverse order; inter-reflected boundaries that are sequenced in the forward order generate clear boundaries in the projector shade. However, the proposed projector shade detection algorithm is capable of identifying and removing the inter-reflected boundaries, regardless of their order in the stripe sequence by checking their legitimacy in terms of the S-SI pairing and the stripe boundary sequence.

Experiments were conducted to test the performance of the proposed projector shade detection method for identifying and removing the inter-reflected boundaries, as illustrated in Fig. 22. The performance of outlier removal in 3D point cloud reconstruction is evaluated with and without the use of the proposed shade detection method for comparison. The comparison is carried out for the case of a forward sequence as shown in Fig. 22 (b) and (c) as well as for the case of a reverse sequence as shown in Fig. 22 (e) and (f). The findings clearly demonstrate the power of the proposed shade detection method for identifying and removing outliers. For more in-depth investigation of inter-reflections in structured light depth imaging, refer to [33].

D. CAMERA SHADE DETECTION

The number of outliers present in the reconstructed 3D point cloud is influenced by the incorporation of not only the



(a) Inter-reflected boundaries in a forward sequence

(b) Outliers without the proposed

(c) Outliers with the proposed

(d) Inter-reflected boundaries in a reverse sequence

(e) Outliers without the proposed

(f) Outliers with the proposed

FIGURE 22. Boundaries appeared in the shades due to inter-reflection can be identified and corrected by the proposed shade detection. (a) Appearance of a forward sequence of the inter-reflected stripe boundaries: ordered same as the projected patterns. (b) and (c) 3D point clouds obtained, respectively, without and with the proposed shade detection: much of the outliers seen clearly in (b) are removed significantly in (c). (d) Appearance of a reverse sequence of the inter-reflected stripe boundaries: reverse-ordered as the projected patterns. (e) and (f) 3D point clouds, respectively, without and with the proposed shade detection: much of the outliers seen clearly in (d) are removed significantly in (e).

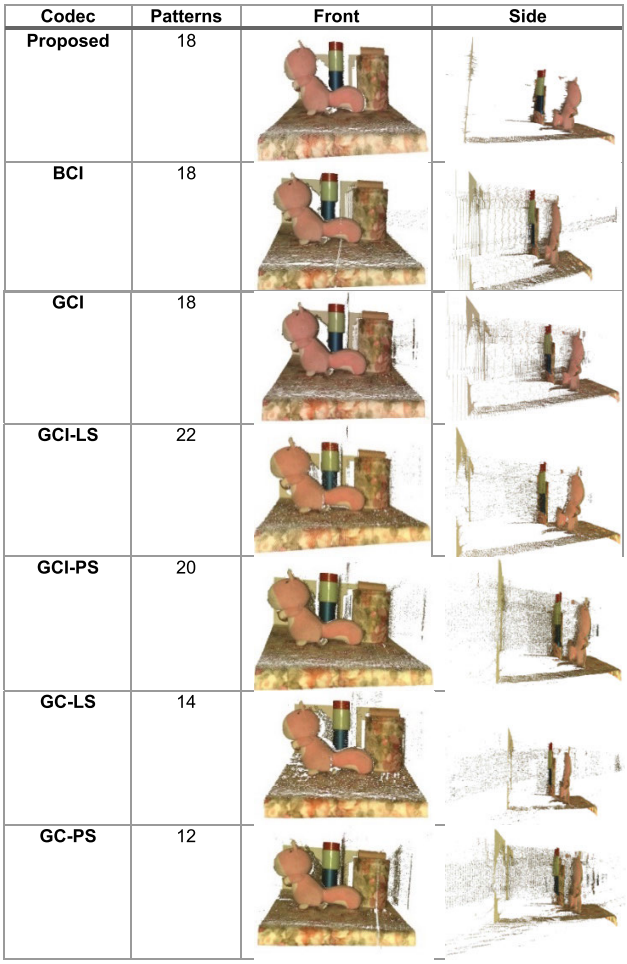


FIGURE 23. 3D point clouds of a scene with colored toys and texture-less objects, captured by different decoding schemes, are shown by their front and side views. Notice the superior performance of the proposed decoding with shade detection for outlier removal.

projector shade but also the camera shade into the decoding of the projector-camera pixel correspondence. As stated, a camera shade causes a jump in the stripe addresses in the camera image. Correct identification of the location and range of the address jump is essential for avoiding decoding errors in order to reduce the number of outliers in 3D point cloud reconstruction. The proposed camera shade detection method ensures accurate identification of an address jump in terms of its location and range. This is done by checking the presence of an address jump in individual layers from the first to the last layer in such a way as to correctly gauge the range of the jump. Note that the discontinuity in the sequence of clear boundaries in the fourth layer alone cannot determine the range of the address jump, as the same P_1-P_2-P_3-P_4 sequence is repeated periodically in space.

To illustrate the significance of incorporating the camera shade into the decoding process to reduce the number of outliers in the point cloud reconstruction, we compare the performance of the proposed coding-decoding(codec), which incorporates the detection of camera shades, with that of conventional codecs in terms of the number of outliers

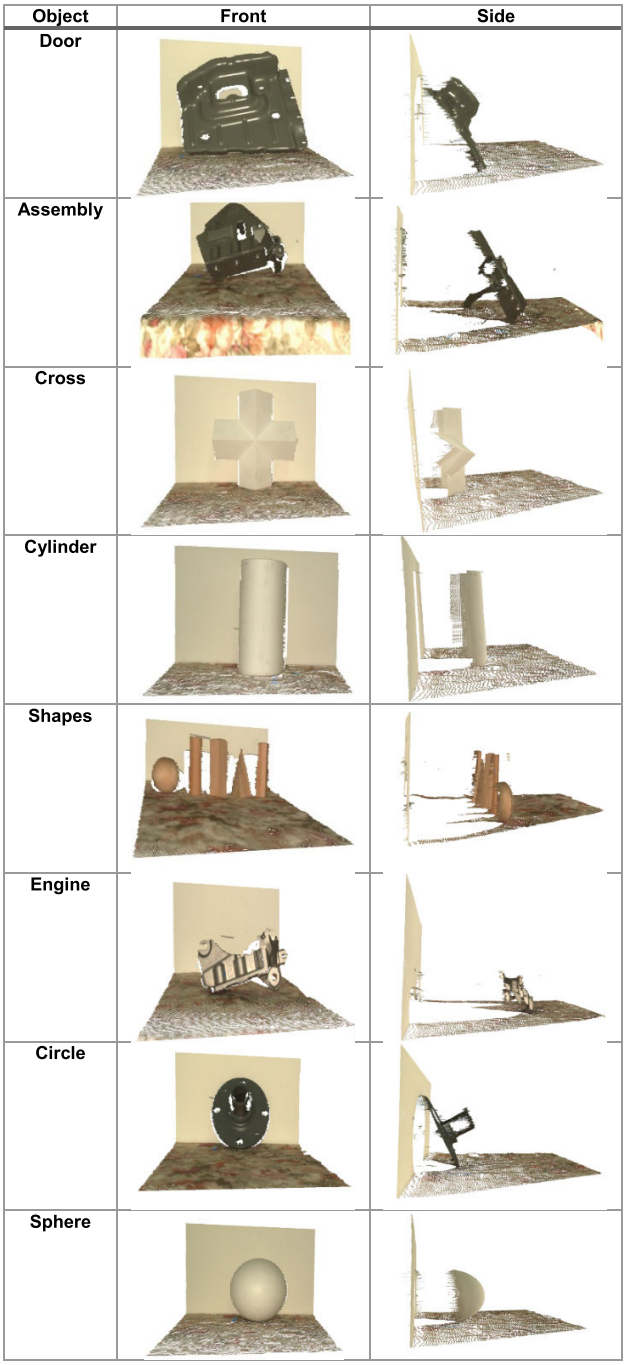


FIGURE 24. 3D point clouds of a scene with industrial parts and un-textured objects, captured by the proposed decoding with shade detection, are shown by their front and side views for quantitative evaluation.

generated. The conventional codecs included here are Binary Code Inverse (BCI), Gray Code Inverse (GCI), Gray Code Inverse with Line and phase shift (GCI + LS/PS) and Gray Code with Line and Phase shift (GC +L S/PS) [34], [35], [36], [37]. To enable a fair comparison, the 3D point cloud of a scene is obtained by projecting the patterns and decoding the projector-camera pixel correspondence based on individual codecs while maintaining the values of the other parameters, including illumination, camera shutter time and distance from

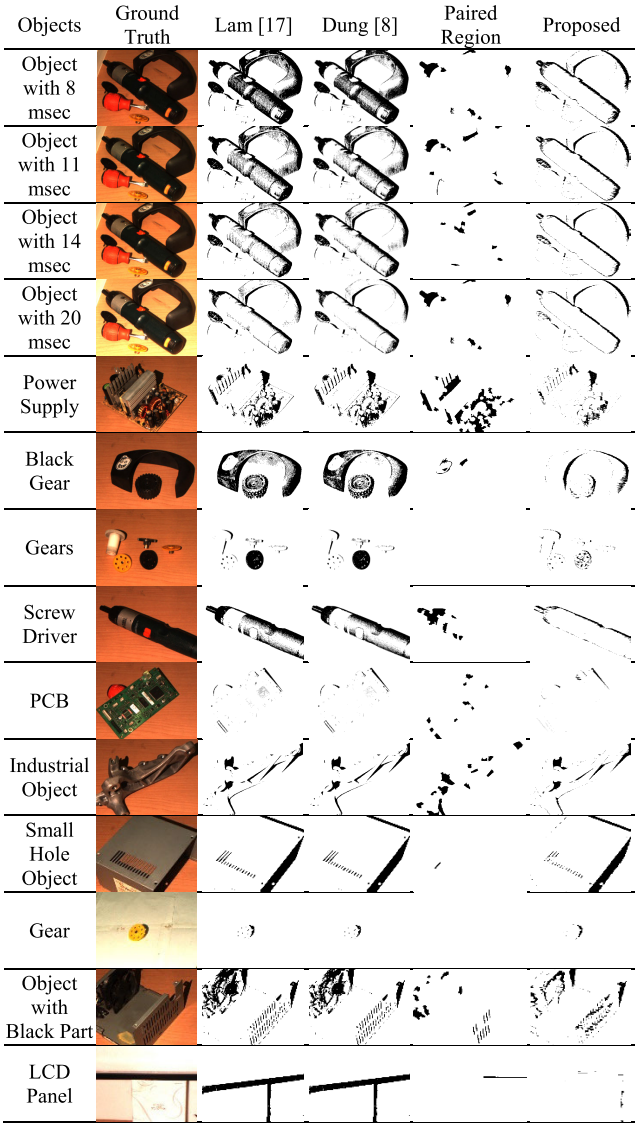


FIGURE 25. Projector shade maps of various objects generated for the comparative performance analysis among different methods of shade detection.

the scene to the camera. Fig. 23 shows the result, which indicates a dramatic reduction in the number of outliers with use of the proposed codec incorporating camera shade detection.

For the quantitative comparison, we compute the percentage of outliers among different codecs for the industrial objects shown in Fig. 24. The percentage is computed as the ratio between the number of outliers and the total number of 3D points obtained.

To this end, the outliers in the resulting 3D point clouds are carefully identified and counted one by one. The results are shown in Table 1, where the proposed codec incorporating the detection of camera shades outperforms the conventional codecs in terms of the percentage of outliers.

E. PERFORMANCE ANALYSIS ON SHADE DETECTION

Here, in order to more precisely assess the performance of the proposed projector shade detection method, we investigate

TABLE 1. The performance of the proposed decoding with shade detection in terms of outlier removal in comparison with conventional decoding approaches (The number inside represents the percentage of outliers).

Object	Proposed	BCI	GCI	GCI		GC	
				LS	*PS*	*LS*	*PS*
Door	0.15	2.31	3.75	1.13	4.1	1.49	6.55
Assembly	0.16	4.82	1.81	1.37	3.33	0.9	6.52
Cross	0.03	3.08	1.81	1.53	1.47	1.76	7.11
Cylinder	0.24	4.93	5.11	0.87	1.65	1.35	8.04
Shapes	0.18	7.59	8.09	2.25	6.08	2.47	7.96
Engine	0.09	5.64	5.54	1.13	3.05	1.30	5.23
Circle	0.35	4.89	3.94	2.07	2.30	2.40	6.86
Sphere	0.15	2.89	3.19	0.36	1.46	1.10	7.88
Toys	0.03	3.61	5.40	2.48	6.26	2.25	8.36
Average	0.15	4.41	4.29	1.46	3.3	1.66	7.16

TABLE 2. True positive (TP): In-Shade rate of shade pixels.

The percentage of Pixels in Shade detected as In-Shade				
Objects	*Lam [17]*	*Dung [8]*	*Paired Region[14]*	*Proposed*
Object 8 msec.	98.40	98.35	0.61	95.32
Object 11 msec.	97.21	96.98	3.16	93.49
Object 14 msec.	96.44	96.17	9.35	92.38
Object 20 msec.	95.90	95.53	0.57	92.77
Power Supply	94.47	94.36	3.48	88.09
Black Gear	96.78	96.74	0.00	92.39
Gears	86.63	86.63	0.00	79.65
Screw Driver	98.00	97.57	8.92	91.86
PCB	99.79	99.79	0.00	95.23
Engine Part	99.96	99.89	18.86	98.63
Small Hole Object	92.90	92.84	1.15	88.51
Gear	90.76	88.57	0.00	86.67
Object with Black part	87.49	87.32	14.59	82.80
LCD Panel	71.06	70.68	0.00	99.49

the effect of the camera exposure time and the reflection coefficient of the object surface on projector shade detection. To this end, we experiment with the camera exposure time of 8, 11, 14 and 20 msec. under the same object as well as with 10 industrial objects with different reflectance coefficients under the same camera exposure time. Then, we compare the proposed projector shade detection method with several state-of-the-art approaches [14], [17], [8] for the various cases of camera exposure time and surface reflectance.

The result is illustrated in Fig. 25. Qualitatively, we can see that the proposed projector shade detection method provides high detection accuracy by precisely discriminating the surface with low reflectance from the projector shade, while the others [8], [17] show poor performance. Note that the proposed projector shade detection method performs better even when no object surface with low reflectance is present. Furthermore, the proposed detection method is able to detect a smaller shade which others fail to detect and performs well independently of the camera exposure time.

To back up the above qualitative analysis, we carry out a quantitative performance assessment based on the following, statistically well-founded, performance metrics: precision, accuracy, sensitivity(recall), specificity and F-measure, as:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{12}$$

TABLE 3. True negative (TN): Not-In-Shade rate of signal pixels.

The percentage of Pixels not in Shade detected as Not-In-Shade				
Objects	*Lam [17]*	*Dung [8]*	*Paired Region [14]*	*Proposed*
Object 8 msec.	36.99	41.62	90.40	78.87
Object 11 msec.	52.96	58.53	91.09	80.65
Object 14 msec.	67.19	71.97	94.31	85.27
Object 20 msec.	79.64	81.91	90.91	85.81
Power Supply	60.27	61.20	51.60	72.77
Black Gear	43.13	49.26	98.37	80.62
Gears	25.58	26.27	100.00	52.34
Screw Driver	53.46	57.16	84.25	84.59
PCB	96.19	96.72	94.04	98.45
Engine Part	73.65	73.92	78.84	73.57
Small Hole Object	53.19	53.22	93.21	60.45
Gear	60.28	62.48	91.82	66.27
Object with Black part	57.76	60.28	88.94	73.94
LCD Panel	31.29	43.70	99.32	91.21

TABLE 4. Performance comparison in various Statistical metrics.

Performance Metrics	*Lam [17]*	*Dung [8]*	*Paired Region[14]*	*Proposed*
Precision	68.21	69.86	34.17	80.20
Accuracy	74.90	76.44	48.01	84.36
Sensitivity	93.27	92.99	4.46	91.23
Specificity	56.54	59.89	81.42	77.48
F-Measure	78.79	79.78	78.96	85.36

TABLE 5. Total number of 3D points generated by state-of-the-art decoding approaches with shade detection.

Objects	*Lam [17]*	*Dung [8]*	*Proposed*
Object 8 msec	173275	184034	242422
Object 11 msec	191872	208315	245735
Object 14 msec	215663	229911	249375
Object 20 msec	237636	242697	250916
Power Supply	224171	224721	231994
Black Gear	216245	228966	268294
Gears	276171	276328	280814
Screw Driver	254088	259458	283562
PCB	287074	288072	290695
Engine Part	261144	261467	262719
Small Hole Object	253470	253431	253857
Gear	297885	297961	298340
Black Object	187136	190463	205441
LCD Panel	270275	272579	290127
Average	239007	244171	261020
Percentage	77.8%	79.48%	84.96%

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{13}$$

$$\text{Sensitivity} = \frac{TP}{TP + FN} \tag{14}$$

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{15}$$

$$\text{F} - \text{Measure} = \frac{2 * TP}{2 * TP + FP + FN} \tag{16}$$

where TP and FN represent, respectively, the true positive and the true negative pixels associated with shade detection.

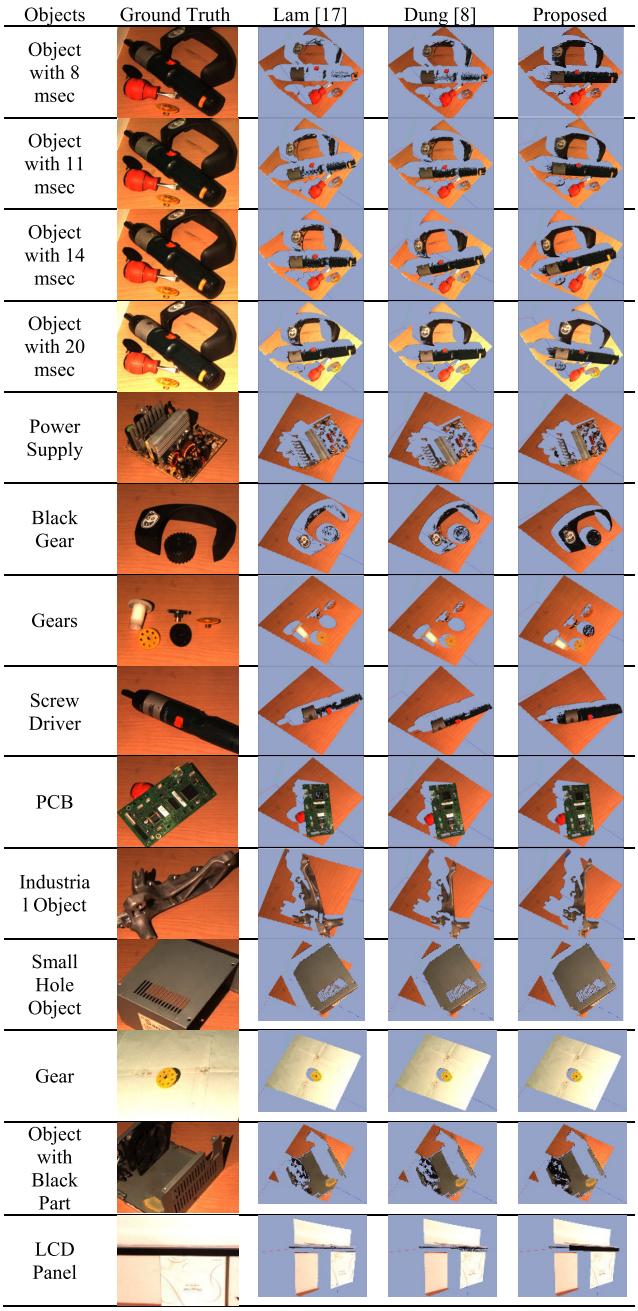


FIGURE 26. Illustration of 3D point clouds generated by state-of-the-art decoding approaches with shade detection for performance comparison. The proposed shade detection outperforms in terms not only of the smallest number of outliers but also of the largest total 3D points captured as shown in Table 5.

To this end, we count on the depth map, the total number of 3D points and carefully compute the percentage of pixels in the shade that are correctly detected as in-shade Pixels (TP) as well as the percentage of pixels not in the shade that are correctly detected as not-in-shade-pixels (TN). Tables II and III show, respectively, the true positive and the true negative detection rates for the experiment illustrated in Fig. 25. Based on Tables II and III, we obtain the values of the performance metrics defined by Eqs. (12) – (16), as shown in Table 4. Table 4 quantitatively verifies that the proposed shade detection method is superior to state-of-the art approaches as it shows improvements in precision, accuracy and F-measure by, at least, 10.34, 7.92 and 5.58%, respectively.

Fig. 26 shows the 3D point clouds obtained by the decoding process based on the proposed and state-of-the-art shade detection approaches. It turns out that the decoding process based on the proposed shade detection method generates 7% more total 3D points than the state-of-the-art approaches, as shown in Table 5.

V. CONCLUSION

This paper demonstrates that the quality of 3D point cloud reconstruction based on structured light depth imaging can significantly be enhanced by incorporating projector and camera shade detection into the decoding of projector-camera pixel correspondence. The key to success in shade detection lies in correctly distinguishing the pixels in the shade from those in a surface with low reflectance. The proposed shade detection method exploits the legitimacy of stripe boundaries in terms of the correct stripe sequence as well as the association of the pixel and address jumps existing in neighboring boundary pairs with the projector and camera shades even under the presence of address juxtaposition. This allows the proposed shade detection method to outperform conventional methods in distinguishing the shade from the surface with low reflectance. We qualitatively and quantitatively illustrate, in extensive experiments using industrial objects, that the proposed shade detection method is shown to significantly decrease the number of outliers while increasing the number of inliers, independently of the camera and projector exposure time. All in all, the decoding process that incorporates the proposed shade detection method improves the quality of 3D point cloud reconstruction beyond that offered by many conventional approaches to accurate shade detection, removal of outliers and correct identification of inliers on the surface with low reflectance. Note that the shade detection approach presented here can be applicable to other forms of binary patterns, such as GCI and BCI. In the future, we plan to apply the proposed approach to other types of pattern encodings while assessing its impact on modeling 3D objects and environments based on extensive experiments.

REFERENCES

[1] J. Liu and Y. Li, "Performance analysis of 3-D shape measurement algorithm with a short baseline projector-camera system," *Robot. Biomimetics*, vol. 1, no. 1, p. 1, Dec. 2014.

[2] J. Geng and J. Xie, "Review of 3-D endoscopic surface imaging techniques," *IEEE Sensors J.*, vol. 14, no. 4, pp. 945–960, Apr. 2014.

[3] K. Zhong, Z. Li, X. Zhou, Y. Li, Y. Shi, and C. Wang, "Enhanced phase measurement profilometry for industrial 3D inspection automation," *Int. J. Adv. Manuf. Technol.*, vol. 76, nos. 9–12, pp. 1563–1574, Feb. 2015.

[4] S. Zhang and S.-T. Yau, "High-resolution, real-time 3D absolute coordinate measurement based on a phase-shifting method," *Opt. Exp.*, vol. 14, no. 7, p. 2644, Apr. 2006.

[5] H. T. Yau, T. J. Yang, and Y. K. Lin, "Comparison of 3-D printing and 5-axis milling for the production of dental E-models from intra-oral scanning," *Comput. Aided. Des. Appl.*, vol. 13, no. 1, pp. 32–38, Jan. 2016.

[6] F. Buchón-Moragues, J. Bravo, M. Ferri, J. Redondo, and J. Sánchez-Pérez, "Application of structured light system technique for authentication of wooden panel paintings," *Sensors*, vol. 16, no. 6, p. 881, Jun. 2016.
[7] E. J. Olaya, F. Berry, and Y. Mezouar, "A robotic structured light camera," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics*, Jul. 2014, pp. 727–734.
[8] H. T. N. Dung and S. Lee, "Outlier removal based on boundary order and shade information in structured light 3D camera," in *Proc. IEEE 7th Int. Conf. Cybern. Intell. Syst. (CIS), IEEE Conf. Robot., Automat. Mechatronics (RAM)*, Jul. 2015, pp. 124–129.
[9] C. Fredembach and S. Süsstrunk, "Automatic and accurate shadow detection from (potentially) a single image using near-infrared information," EPFL, Lausanne, Switzerland, Tech. Rep. 165527, 2010.
[10] G. Finlayson, C. Fredembach, and M. S. Drew, "Detecting illumination in images," in *Proc. IEEE 11th Int. Conf. Comput. Vis.*, Oct. 2007, pp. 1–8.
[11] S. K. Nayar and M. Gupta, "Diffuse structured light," in *Proc. IEEE Int. Conf. Comput. Photogr. (ICCP)*, Apr. 2012, pp. 1–11.
[12] J. Park, G. N. DeSouza, and A. C. Kak, "Dual-beam structured-light scanning for 3-D object modeling," in *Proc. 3rd Int. Conf. 3-D Digit. Imag. Modeling*, 2001, pp. 65–72.
[13] J. M. DiCarlo, F. Xiao, and B. A. Wandell, "Illuminating illumination," in *Proc. Color Imag. Conf.*, 2001, vol. 2001, no. 1, pp. 27–34.
[14] R. Guo, Q. Dai, and D. Hoiem, "Single-image shadow detection and removal using paired regions," in *Proc. CVPR*, 2011, pp. 2033–2040.
[15] M. Atif and S. Lee, "Boundary based shade detection," in *Proc. IEEE Int. Conf. Multisensor Fusion Integr. Intell. Syst. (MFI)*, Sep. 2016, pp. 635–640.
[16] J. Salvi, S. Fernandez, T. Pribanic, and X. Llado, "A state of the art in structured light patterns for surface profilometry," *Pattern Recognit.*, vol. 43, no. 8, pp. 2666–2680, Aug. 2010.
[17] L. Q. Bui and S. Lee, "Boundary inheritance codec for high-accuracy structured light three-dimensional reconstruction with comparative performance evaluation," *Appl. Opt.*, vol. 52, no. 22, p. 5355, Aug. 2013.
[18] H. Nguyen, D. Nguyen, Z. Wang, H. Kieu, and M. Le, "Real-time, high-accuracy 3D imaging and shape measurement," *Appl. Opt.*, vol. 54, no. 1, p. A9, Jan. 2015.
[19] M. Atif and S. Lee, "FPGA based pattern generation and synchonization for high speed structured light 3D camera," *Telecommun. Comput. Electron. Control.*, vol. 15, no. 1, pp. 926–933, 2017.
[20] S. Lee, J. Choi, S. Oh, J. Ryu, and J. Park, "A real-time 3D IR camera based on hierarchical orthogonal coding," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, May 2006, pp. 2035–2040.
[21] Y. Gong and S. Zhang, "Ultrafast 3-D shape measurement with an off-the-shelf DLP projector," *Opt. Exp.*, vol. 18, no. 19, p. 19743, Sep. 2010.
[22] M. Atif and S. Lee, "FPGA based adaptive rate and manifold pattern projection for structured light 3D camera system," *Sensors*, vol. 18, no. 4, p. 1139, Apr. 2018.
[23] S. Zhang, J.-S. Hyun, and B. Li, "High-speed 3D imaging using digital binary defocusing method vs sinusoidal method," *Proc. SPIE*, vol. 10117, Feb. 2017, Art. no. 1011707.
[24] Z. Wang, "Three-dimensional surface imaging by multi-frequency phase shift profilometry with angle and pattern modeling for system calibration," *Meas. Sci. Technol.*, vol. 27, no. 8, Aug. 2016, Art. no. 085404.
[25] Z. Wang, "An imaging and measurement system for robust reconstruction of weld pool during arc welding," *IEEE Trans. Ind. Electron.*, vol. 62, no. 8, pp. 5109–5118, Aug. 2015.
[26] Y. Wang, K. Liu, Q. Hao, D. Lau, and L. G. Hassebrook, "Multicamera phase measuring profilometry for accurate depth measurement," *Proc. SPIE*, vol. 6555, May 2007, Art. no. 655509.
[27] M. Weinmann, C. Schwartz, R. Ruiters, and R. Klein, "A multi-camera, multi-projector super-resolution framework for structured light," in *Proc. Int. Conf. 3D Imag., Modeling, Process., Vis. Transmiss.*, 2011, pp. 397–404.
[28] Z. Song, R. Chung, and X.-T. Zhang, "An accurate and robust strip-edge-based structured light means for shiny surface micromeasurement in 3-D," *IEEE Trans. Ind. Electron.*, vol. 60, no. 3, pp. 1023–1032, Mar. 2013.
[29] S. Lee, J. Choi, D. Kim, J. Na, and S. Oh, "Signal separation coding for robust depth imaging based on structured light," in *Proc. IEEE Int. Conf. Robot. Automat.*, Apr. 2005, pp. 4430–4436.
[30] S. Lee and L. Q. Bui, "Accurate estimation of the boundaries of a structured light pattern," *J. Opt. Soc. Amer. A, Opt. Image Sci.*, vol. 28, no. 6, p. 954, Jun. 2011.
[31] W. H. Richardson, "Bayesian-based iterative method of image restoration," *J. Opt. Soc. Amer.*, vol. 62, no. 1, p. 55, Jan. 1972.
[32] L. B. Lucy, "An iterative technique for the rectification of observed distributions," *Astronomical J.*, vol. 79, p. 745, Jun. 1974.
[33] L. Q. Bui and S. Lee, "A method of eliminating interreflection in 3D reconstruction using structured light 3D camera," in *Proc. Int. Conf. Comput. Vis. Theory Appl. (VISAPP)*, vol. 3, 2014, pp. 640–647.
[34] J. Guehring, "Dense 3D surface acquisition by structured light using off-the-shelf components," *Proc. SPIE*, vol. 4309, pp. 220–231, Dec. 2000.
[35] D. Zheng and F. Da, "Self-correction phase unwrapping method based on gray-code light," *Opt. Lasers Eng.*, vol. 50, no. 8, pp. 1130–1139, Aug. 2012.
[36] G. Sansoni, A. Patrioli, and F. Docchio, "OPL-3D: A novel, portable optical digitizer for fast acquisition of free-form surfaces," *Rev. Sci. Instrum.*, vol. 74, no. 4, pp. 2593–2603, Apr. 2003.
[37] G. Wiora, "High-resolution measurement of phase-shift amplitude and numeric object phase calculation," *Proc. SPIE*, vol. 4117, pp. 289–299, Oct. 2000.
[38] H. Gärtner, P. Lehle, and H. J. Tiziani, "New, highly efficient, binary codes for structured light methods," *Proc. SPIE*, vol. 2599, pp. 4–13, Jan. 1996.
[39] F. Blais, "Review of 20 years of range sensor development," *J. Electron. Imag.*, vol. 13, no. 1, pp. 231–240, Jan. 2004.
[40] J. L. Posdamer and M. D. Altschuler, "Surface measurement by space-encoded projected beam systems," *Comput. Graph. Image Process.*, vol. 18, no. 1, pp. 1–17, Jan. 1982.
[41] T. Chen, H.-P. Seidel, and H. P. A. Lensch, "Modulated phase-shifting for 3D scanning," in *Proc. CVPR*, 2008, pp. 1–8.
[42] S. Zhang, D. V. D. Weide, and J. Oliver, "Superfast phaseshifting method for 3-D shape measurement," *Opt. Exp.*, vol. 18, no. 9, pp. 9684–9689, 2010.

SUKHAN LEE (Life Fellow, IEEE) received the B.S. and M.S. degrees in electrical engineering from Seoul National University, South Korea, in 1972 and 1974, respectively, and the Ph.D. degree in electrical engineering from Purdue University, West Lafayette, IN, USA, in 1982. From 1983 to 1997, he was with the Department of Electrical Engineering, University of Southern California, and the Department of Computer Science, University of Southern California. From 1990 to 1997, he was with the Jet Propulsion Laboratory, California Institute of Technology. From 1998 to 2003, he was an Executive Vice President and a Chief Research Officer with the Samsung Advanced Institute of Technology. Since 2003, he has been a Professor of Information and Communication Engineering with Sungkyunkwan University. His research interests include cognitive robotics, intelligent systems, deep learning, and micro/nano electro-mechanical systems. He is currently a Life Fellow of the Korea National Academy of Science and Technology.

MUHAMMAD ATIF received the bachelor's degree from Bahauddin Zakariya University (NFC IET), Multan, Pakistan, in 2006, the M.E. degree in telecommunication engineering from NUCES FAST, Islamabad, Pakistan, in 2010, and the Ph.D. degree from Sungkyunkwan University, Suwon, South Korea, in 2019. He has been working as a Postdoctoral Research Associate with the Korea Institute of Science and Technology (KIST), Seoul, South Korea, since 2019.