applied optics

Computer-aided-design-model-assisted absolute three-dimensional shape measurement

BEIWEN LI,[1,*] TYLER BELL,[2] AND SONG ZHANG[2,3]

[1]*Department of Mechanical Engineering, Iowa State University, Ames, Iowa 50011, USA*
[2]*School of Mechanical Engineering, Purdue University, West Lafayette, Indiana 47907, USA*
[3]*e-mail: szhang15@purdue.edu*
**Corresponding author: beiwen@iastate.edu*

Received 22 June 2017; revised 20 July 2017; accepted 20 July 2017; posted 21 July 2017 (Doc. ID 300740); published 14 August 2017

Conventional three-dimensional (3D) shape measurement methods are typically generic to all types of objects. Yet, for many measurement conditions, such a level of generality is inessential when having the preknowledge of the object geometry. This paper introduces a novel adaptive algorithm for absolute 3D shape measurement with the assistance of the object computer-aided-design (CAD) model. The proposed algorithm includes the following major steps: (1) export the 3D point cloud data from the CAD model; (2) transform the CAD model into the camera perspective; (3) obtain a wrapped phase map from three phase-shifted fringe images; and (4) retrieve absolute phase and 3D geometry assisted by the CAD model. We demonstrate that if object CAD models are available, such an algorithm is efficient in recovering absolute 3D geometries of both simple and complex objects with only three phase-shifted fringe images.

OCIS codes: (120.0120) Instrumentation, measurement, and metrology; (100.5088) Phase unwrapping; (110.5086) Phase unwrapping; (100.5070) Phase retrieval.

https://doi.org/10.1364/AO.56.006770

1. INTRODUCTION

Three-dimensional (3D) shape measurement technologies have been exciting for a variety of applications, including industrial inspection, entertainment, and medicine.

For decades, scientists have been conducting extensive research on different types of 3D shape measurement methods. In general, among different types of 3D shape measurement techniques, phase analysis techniques have advantages over intensity analysis techniques owing to their high spatial resolutions and insensitiveness to surface reflectivity variations. Some popular phase analysis techniques include the Fourier transform method [1], the Windowed Fourier transform method [2,3], and the phase-shifting methods [4]. Typically, a phase analysis technique produces a wrapped phase map with 2π discontinuities. Then, a spatial [5] or temporal phase unwrapping algorithm [6–14] is necessary to create a continuous phase map by determining the integer k (i.e., the *fringe order*) multiples of 2π to be added at each phase's discontinuous point.

Essentially, the phase analysis approaches accompanied by a phase unwrapping algorithm can be generically applied to the measurements of different types of objects. To achieve such generality, however, requires extra effort to increase the efficiency and robustness of the algorithms. For instance, a typical spatial phase unwrapping method locally searches for 2π discontinuities from the phase map itself. Despite numerous efforts (e.g., using a quality- [15] or reliability-guided method [16]) to increase the robustness of spatial phase unwrapping, it is still challenging for such methods to deal with cases with abrupt spatial discontinuities, not to mention that the generated phase map is only *relative* to a point on a single connected component. Although temporal phase unwrapping techniques are able to address the aforementioned challenges, the projection of extra images, such as multiple frequency patterns [11–14], or intensity-coded [6,7] or phase-coded patterns [8–10], are typically inevitable to perform unwrapping pixel by pixel. An *et al.* [17] has recently proposed a method based on geometric constraints that does not require the projection of additional images. However, such a method can recover only object geometries within a depth range of 2π in phase domain, something undesirable when the period of projected patterns is narrow [17].

Nowadays, computer-aided design (CAD) has been an important mean to assist the building and optimization of a design [18]. For an object that is produced by a modern manufacturing machine, its CAD model is typically available from the designers. Such a valuable resource, however, has rarely been utilized to assist 3D shape measurements in previous research works. In fact, this preknowledge of object geometry, as strong extra information, could bring about the potential to develop more efficient 3D shape measurement algorithms.

This paper thus proposes a novel adaptive algorithm that can measure objects with a known CAD model. Using the pre-knowledge of the object geometry from the CAD model, we can recover absolute 3D geometry of the object with only three phase-shifted patterns. In our proposed computational framework, we first export the 3D point cloud data from the CAD model, which is then transformed into the real camera perspective to assist phase unwrapping by segmenting the measured object into different depth volumes. We will explain the principles in details along with experimental validations to demonstrate that such a method can well recover both simple and complex 3D geometries given that the original CAD model is available.

Section 2 introduces principles relevant to this research as well as our proposed computational framework; Section 3 demonstrates the experimental validations of our algorithm; and Section 4 summarizes the contributions of our proposed research.

2. PRINCIPLES

This section introduces some theoretical background related to this research. Specifically, we will explain the basics of a phase-shifting algorithm, the modeling of a digital fringe projection (DFP) system, and our proposed measurement method assisted by CAD model.

A. Three-Step Phase-Shifting Algorithm

There are a variety of phase analysis techniques in optical metrology. Within which, the phase-shifting algorithms have the advantages of high accuracy and robustness. Among all phase-shifting techniques, the three-step phase-shifting algorithm requires the minimum number of images for phase computation, making it preferable for high-speed applications. The three phase-shifted patterns that have equal phase shifts can be modeled as

$$I_1(x,y) = I'(x,y) + I''(x,y)\cos[\phi - 2\pi/3], \tag{1}$$

$$I_2(x,y) = I'(x,y) + I''(x,y)\cos[\phi], \tag{2}$$

$$I_3(x,y) = I'(x,y) + I''(x,y)\cos[\phi + 2\pi/3], \tag{3}$$

where $I'(x,y)$ represents the average intensity, $I''(x;y)$ denotes the intensity modulation, and ϕ is the phase to be extracted. Simultaneously solving Eqs. (1)–(3), the phase ϕ can be computed as follows:

$$\phi(x,y) = \tan^{-1}\left[\frac{\sqrt{3}(I_1 - I_3)}{2I_2 - I_1 - I_3}\right]. \tag{4}$$

The computed phase map ranges from $-\pi$ to π owing to the nature of an arctangent function. A phase unwrapping algorithm is necessary to produce a continuous phase map by adding integer $k(x,y)$ multiples of 2π, as follows:

$$\Phi(x,y) = \phi(x,y) + 2\pi \times k(x,y). \tag{5}$$

The integer $k(x,y)$ is typically referred to as the *fringe order.*

Besides phase information, from Eqs. (1)–(3) we can also compute the texture information $I_t(x,y)$ as follows:

$$I_t(x,y) = \frac{I_1 + I_2 + I_3}{3} + \frac{\sqrt{(I_1 - I_3)^2 + (2I_2 - I_1 - I_3)}}{3}. \tag{6}$$

The texture information looks like an actual photograph of the imaged scene that can be used for applications such as object recognition or feature extraction.

B. DFP System Model

The imaging lenses (e.g., camera lens, projector lens) in a DFP system respect the well-known pinhole model [19], which describes the projection from 3D world coordinate (x^w, y^w, z^w) to 2D image coordinate (u, v), as follows:

$$s\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_u & \gamma & u_0 \\ 0 & f_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix}\begin{bmatrix} x^w \\ y^w \\ z^w \\ 1 \end{bmatrix}. \tag{7}$$

In this model, s represents the scaling factor; f_u and f_v are, respectively, the effective focal lengths of the lens along u and v directions; γ stands for the skew of u and v axes; (u_0, v_0) denotes the principle point; and r_{ij} and t_i, respectively, represent the rotation and translation parameters. We then define a projection matrix $\mathbf{P}$ to simplify the expression

$$\mathbf{P} = \begin{bmatrix} f_u & \gamma & u_0 \\ 0 & f_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix}, \tag{8}$$

$$= \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix}. \tag{9}$$

This projection matrix $\mathbf{P}$ can be estimated using some well-developed software toolboxes (e.g., MatLab, OpenCV, etc.).

The camera and the projector share the same pinhole model despite their mutually inverted optics. We can physically correlate the two imaging lenses under the same world coordinate system (x^w, y^w, z^w), as follows:

$$s^c[u^c \quad v^c \quad 1]^t = \mathbf{P}^c[x^w \quad y^w \quad z^w \quad 1]^t, \tag{10}$$

$$s^p[u^p \quad v^p \quad 1]^t = \mathbf{P}^p[x^w \quad y^w \quad z^w \quad 1]^t. \tag{11}$$

Here, superscript p, c, and t, respectively, denote the projector, the camera, and the matrix transpose. For simplification, we aligned the world coordinate with the camera lens coordinate, and we used a linear model for both the camera and the projector.

Since Eqs. (10) and (11) above provide six equations, yet with seven unknowns $(s^c, s^p, x^w, y^w, z^w, u^p, v^p)$, an additional equation is necessary that can be provided by the linear constraint between phase Φ and a projector pixel line u^p assuming fringe stripes vary along u^p direction, as follows:

$$u^p = \Phi \times T/(2\pi). \tag{12}$$

Here T is the fringe period in projector pixels, and Φ is required to be the *absolute* phase that is continuous in camera image space. Given that the absolute phase Φ starts from 0 and is monotonically increasing in projector space, Eq. (12) simply converts absolute phase into projector pixels. Once the absolute phase Φ is obtained, by simultaneously solving Eqs. (10)–(12), one can extract 3D coordinates (x^w, y^w, z^w) for each camera pixel (u^c, v^c). However, 2π discontinuities are present on the extracted phase ϕ from the phase-shifting algorithm. The next two sections will introduce our proposed phase unwrapping framework assisted by the CAD model.

C. Proposed Measurement Method Assisted by CAD Model

Figure 1 shows the conceptual idea of our proposed method assisted by CAD model. Suppose the measured object has a known CAD model and can be transformed into the camera perspective, we can uniformly segment the object by different depth planes z_i so that each Δz between two adjacent depth planes will not exceed 2π in the phase domain. The details of the determination of Δz can be found in the research reported by An *et al.* [17]. The result of segmentation can be stored in several mask images M_i correspondingly,

$$\mathbf{M}_i(u^c, v^c) = \begin{cases} 1 & z_i < z(x, y) <= z_{i+1} \\ 0 & \text{otherwise} \end{cases}, \tag{13}$$

where

$$z_i = z_1 + (i - 1) \times \Delta z. \tag{14}$$

Also, for a calibrated DFP system, any z_i plane actually corresponds to an artificial absolute phase map $\Phi^i_{\min}$ [17]. The theoretical foundation lies in the following: the calibration process for a DFP system has determined all parameters in projection matrices P^c and P^p. With a given z_i, by simultaneously solving Eqs. (10) and (11), we can compute x^w and y^w for each camera pixel (u^c, v^c)

$$\begin{bmatrix} x^w \\ y^w \end{bmatrix} = A^{-1} b, \tag{15}$$

where

$$A = \begin{bmatrix} p^c_{31} u^c - p^c_{11} & p^c_{32} u^c - p^c_{12} \\ p^c_{31} v^c - p^c_{21} & p^c_{32} v^c - p^c_{22} \end{bmatrix}, \tag{16}$$

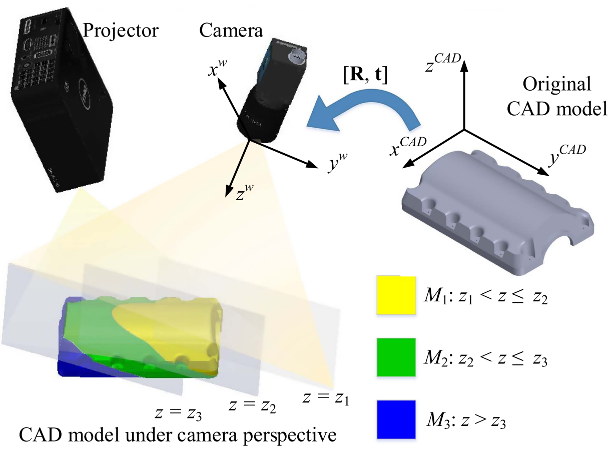

Fig. 1. Illustration of depth segmentation assisted by the CAD model. The CAD model is transformed into the camera perspective to be used as a reference to perform depth segmentation. The depth interval Δz between each plane is less than 2π in the phase domain.

$$b = \begin{bmatrix} p^c_{14} - p^c_{34} u^c - (p^c_{33} u^c - p^c_{13}) z_i \\ p^c_{24} - p^c_{34} v^c - (p^c_{33} v^c - p^c_{23}) z_i \end{bmatrix}. \tag{17}$$

Here, p^c_{ij} denotes the parameter in matrix P^c at the ith row and jth column. Once (x^w, y^w) is computed, for each camera pixel (u^c, v^c), the corresponding projector pixel (u^p, v^p) can be solved by

$$s^p [u^p \quad v^p \quad 1]^t = P^p [x^w \quad y^w \quad z_i \quad 1]^t. \tag{18}$$

Resultantly, suppose the intensity of projected fringe patterns vary along u^p direction with a fringe period of T pixels, the artificial absolute phase $\Phi^i_{\min}$ defined on camera pixel can be determined from

$$\Phi^i_{\min}(u^c, v^c) = u^p \times 2\pi / T. \tag{19}$$

Since each of our segmented depth volume Δz does not exceed 2π in the phase domain, for each region marked by M_i, we can therefore determine the fringe order k_i using the artificial absolute phase map $\Phi^i_{\min}$ corresponding to its lower bound depth plane z_i,

$$k_i - 1 < \frac{\Phi^i_{\min} - \phi}{2\pi} < k_i, \tag{20}$$

or explicitly

$$k_i = \mathrm{ceil}\left[\frac{\Phi^i_{\min} - \phi}{2\pi}\right], \tag{21}$$

where the ceiling operator ceil[] rounds up a floating point number to its closest upper integer.

Finally, the absolute phase map can be computed by separately unwrapping the phase in different regions marked by M_i, and then merging all segments together. Suppose the entire object is divided into n segments, the final absolute phase map can be computed as follows:

$$\Phi(u^c, v^c) = \sum_{i=1}^{n} [(\phi + 2\pi \times k_i) \times \mathbf{M}_i(u^c, v^c)]. \tag{22}$$

This section has conceptually presented the entire computational framework. However, to make this conceptual idea a real practice, one needs to develop detailed step-by-step procedures. The next section will elaborate on the entire working pipeline of our proposed method using a simple step-height object as an example.

D. Procedures

In this section, we use a simple step-height object shown in Fig. 2(a) as an example to step-by-step elucidate our proposed computational framework assisted by the CAD model. The detailed procedures as presented as follows:

- *Step 1: 3D data exportation from the CAD model.* The first step is to extract x, y, and z coordinates for the entire CAD model, denoted as $(x^{\text{CAD}}, y^{\text{CAD}}, z^{\text{CAD}})$. The CAD model of the step-height object is shown in Fig. 2(b). The CAD model is first rendered within OpenGL in shaded mode; this fills in all the geometries between the CAD model's points. The 3D coordinates for each pixel on the rendered CAD model are then encoded via the multiwavelength depth encoding method [20] resulting in a 2D RGB image, as shown in Fig. 2(c). This 2D image is then decoded to recover all the 3D coordinates for the shaded CAD model. The decoded 3D point cloud geometry of the CAD model is shown in Fig. 2(d). Worth noting is that our depth encoding adopts an orthographic projection model, which makes the bottom plane invisible and thus cannot be recovered in the decoded 3D geometry shown in Fig. 2(d). Therefore, it should be noted that our algorithm cannot handle those sides of the object that are invisible in the orthographic projection.

- *Step 2: Extraction of wrapped phase map and texture.* Figure 3(a) shows a representative captured fringe image. Using the three-step phase-shifting algorithm as introduced in Section 2.A, we can compute a wrapped phase map ϕ and the texture image $I_t(x, y)$ as shown in Figs. 3(b) and 3(c).
- *Step 3: Feature extraction.* After obtaining the 3D point cloud from the CAD model, the wrapped phase map, and the texture image, we can then transform the CAD 3D data into the real camera perspective to assist depth segmentation. To perform this task, we can extract some feature points (corners, centers, etc.) both on the camera image (denoted as (u^f, v^f)) and on the CAD model (denoted as (x^f, y^f, z^f)). The camera image that we use is the texture image $I_t(x, y)$ shown in Fig. 3(c) obtained from three-step phase shifting. Figures 4(a) and 4(b), respectively, show the extracted feature points both on the camera image and on the CAD model. The feature extraction can be performed using some well-established feature extraction algorithms (e.g., Harris corner detection, Hough transform circle detection, etc.). In our experimental practices, we found that even a rough estimation through manually selecting the feature points is sufficient for the entire algorithm to succeed.

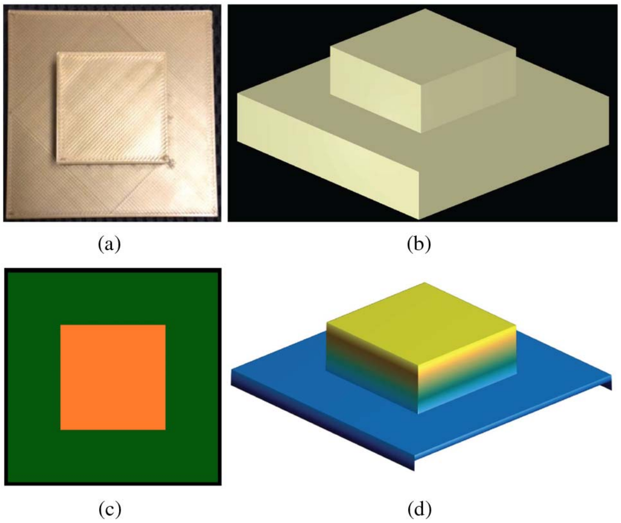

Fig. 2. Three-dimensional data exportation from the CAD model. (a) A photograph of a 3D printed step-height object; (b) the CAD model of the object; (c) the compressed image of the CAD model with multiwavelength depth encoding; (d) the decoded 3D geometry from the CAD model.

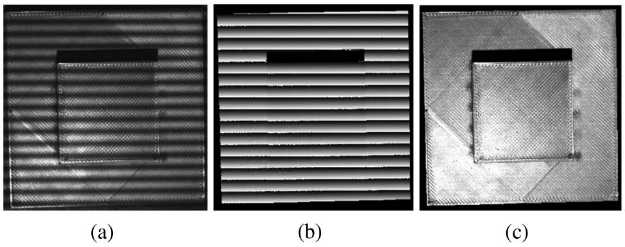

Fig. 3. Computation of wrapped phase ϕ and texture image $I_t(x, y)$ using a three-step phase-shifting approach. (a) One of the captured fringe images; (b) the computed wrapped phase ϕ; (c) the computed texture image $I_t(x, y)$.

- *Step 4: Transforming the CAD 3D data into the real camera perspective.* Once we have extracted the feature points, the next step is to transform the CAD 3D data into the real camera perspective. Given that the world coordinate is aligned with the camera lens coordinate, the key to this step is to estimate the transformation $[\mathbf{R}, \mathbf{t}]$ from CAD 3D data $(x^{\text{CAD}}, y^{\text{CAD}}, z^{\text{CAD}})$ to the world coordinate (x^w, y^w, z^w),

$$\begin{bmatrix} x^w \\ y^w \\ z^w \\ 1 \end{bmatrix} = [\mathbf{R}, \mathbf{t}] \begin{bmatrix} x^{\text{CAD}} \\ y^{\text{CAD}} \\ z^{\text{CAD}} \\ 1 \end{bmatrix}, \tag{23}$$

where $\mathbf{R}$ is a 3×3 rotation matrix and $\mathbf{t}$ is a 3×1 translation vector. This transformation matrix $[\mathbf{R}, \mathbf{t}]$ can be estimated using the extracted feature points (u^f, v^f) and (x^f, y^f, z^f) from the previous step. This 3D to 2D point correspondence can be estimated using an OpenCV function *solvePnP*. It inherently uses an iterative Levenberg–Marquardt optimization to estimate the transformation $[\mathbf{R}, \mathbf{t}]$ with the following functional:

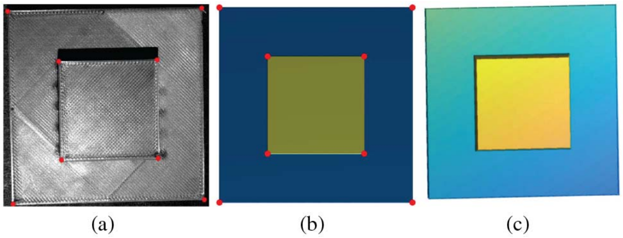

Fig. 4. Feature extraction on both the 2D texture image $I_t(x, y)$ and the 3D CAD model. (a) Extracted 2D feature points on texture image $I_t(x, y)$; (b) extracted feature points on the 3D CAD model; (c) transformed CAD model under world coordinate.

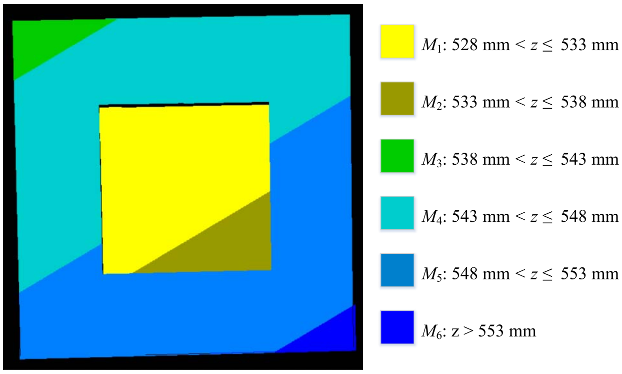
M_1: 528 mm < $z \leq$ 533 mm
M_2: 533 mm < $z \leq$ 538 mm
M_3: 538 mm < $z \leq$ 543 mm
M_4: 543 mm < $z \leq$ 548 mm
M_5: 548 mm < $z \leq$ 553 mm
M_6: z > 553 mm

Fig. 5. Object mask images obtained by depth z segmentation of the transformed 3D data shown in Fig. 4(c); all units are in mm.

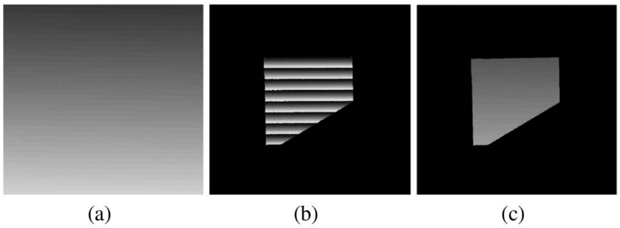

Fig. 6. Example of phase unwrapping inside of M_1. (a) The artificial absolute phase map $\Phi^1_{\min}$ extracted at the depth plane z_1 for the segmented region M_1 in Fig. 5; (b) the wrapped phase map inside of M_1; (c) the unwrapped phase map of (b) inside of M_1.

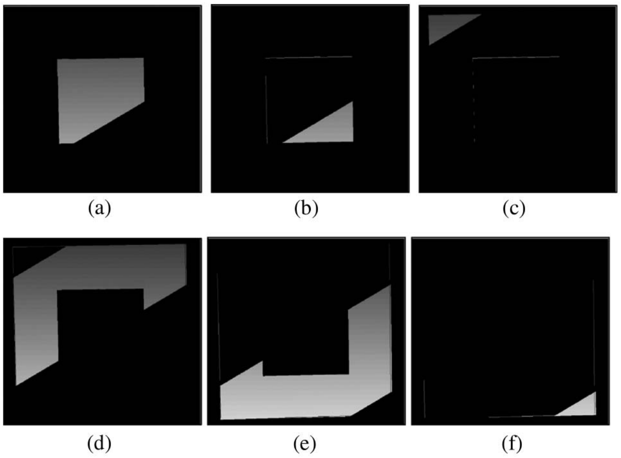

Fig. 7. Unwrapped phases for different depth ranges using the corresponding $\Phi^i_{\min}$ map for each region in Fig. 5. (a)–(f) Unwrapped phases for the region inside of M_1–M_6.

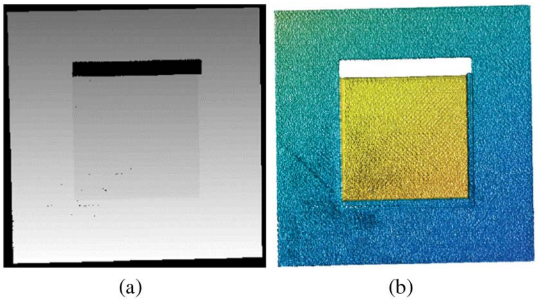

Fig. 8. Final result after merging all phase maps in Fig. 7. (a) Final absolute phase map; (b) reconstructed 3D result.

$$\min_{\mathbf{R},\,\mathbf{t}} \left\| s^c \begin{bmatrix} u^f \\ v^f \\ 1 \end{bmatrix} - \mathbf{P}^c[\mathbf{R},\mathbf{t}] \begin{bmatrix} x^f \\ y^f \\ z^f \\ 1 \end{bmatrix} \right\|. \tag{24}$$

Once the transformation $[\mathbf{R}, \mathbf{t}]$ is estimated, we transform the CAD 3D data $(x^{\mathrm{CAD}}, y^{\mathrm{CAD}}, z^{\mathrm{CAD}})$ into the world coordinate (x^w, y^w, z^w) using Eq. (23). The transformed CAD model is shown in Fig. 4(c).

- *Step 5: Using the transformed CAD data to perform depth segmentation.* The transformed CAD 3D data in Fig. 4(c) can then be used to perform depth segmentation. Figure 5 shows the created mask images $\mathbf{M}_1$–$\mathbf{M}_6$ for the measured object according to the object depth $z(x, y)$ in Fig. 4(c). In this specific case, we set the depth interval Δz as 5 mm, which is within 2π in the phase domain. In total, there are $n = 6$ segmented depth ranges by six different depth planes.
- *Step 6: Phase unwrapping and 3D reconstruction.* After depth segmentation, for the ith region labeled by M_i, we first find its lower bound depth plane at $z_i = z_1 + (i - 1) \times \Delta z$. Following the principles described in Section C, we create an artificial absolute phase map $\Phi^i_{\min}$ at z_i. Then, following Eq. (21), we can unwrap the phase for each segment in M_i by finding the corresponding fringe order k_i. Figure 6 shows an example unwrapping procedure for the region labeled in M_1. Figures 6(b)–6(c), respectively, show the wrapped and unwrapped phase map inside of M_1. The same unwrapping procedure is also applicable to other regions (i.e., M_2–M_6) as well. Figure 7 shows all the unwrapped phase maps within M_1–M_6, respectively, clearly indicating that the unwrapping is successful for the phases in all different regions. Finally, following Eq. (22), we can merge all segments in Fig. 7 into a complete absolute phase map. The final absolute phase map $\Phi(u^c, v^c)$ is shown in Fig. 8(a), from which we can reconstruct the 3D geometry of the object, and the 3D result is shown in Fig. 8(b).

3. EXPERIMENT

We set up a DFP system to further verify the performance of our proposed phase unwrapping framework. A digital CCD camera (the Imaging Source DMK 23UX174) with a resolution of 1280 × 1024 pixels is used as the image acquisition device, and a digital light processing projector (Dell M115HD) with a resolution of 1280 × 800 pixels is used for pattern projection. The camera is attached with a 16 mm focal length lens (Fujinon HF16HA-1B).

To validate the performance of our proposed algorithm, we measured a complex mechanical part as shown in Fig. 9(a).

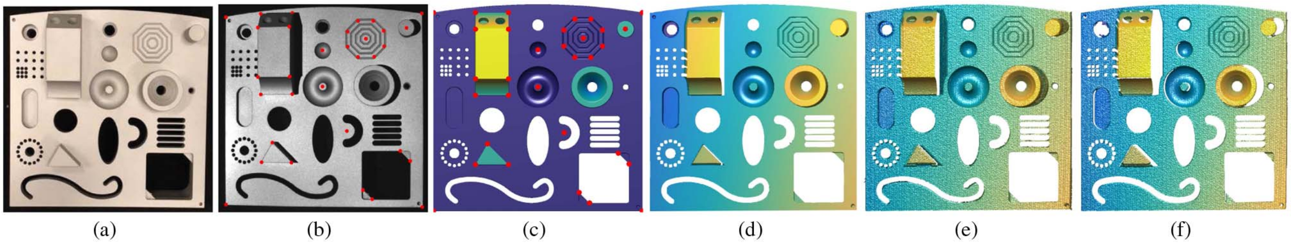

Fig. 9. Measurement of a mechanical part with complex 3D geometry. (a) A photograph of the part; (b) the texture image with feature extraction; (c) the original CAD 3D data with feature extraction; (d) the CAD 3D transformed into camera perspective; (e) reconstructed 3D geometry using the proposed method; (f) reconstructed 3D geometry using conventional temporal phase unwrapping (gray coding [6]) method.

As aforementioned, before implementing our proposed computational framework, the exported CAD 3D data needs to be transformed into the camera perspective by extracting feature points. Figures 9(b) and 9(c) show the extracted feature points (i.e., corners, circle centers) on both the 2D texture image and the CAD 3D data, respectively. Figure 9(d) shows the CAD 3D geometry that has been transformed into the camera perspective using the previously introduced method in Section 2.D. To demonstrate that our method can indeed recover absolute 3D geometry, we reconstructed the 3D geometry using both our proposed method and a conventional temporal phase unwrapping method (i.e., the gray coding [6] method).

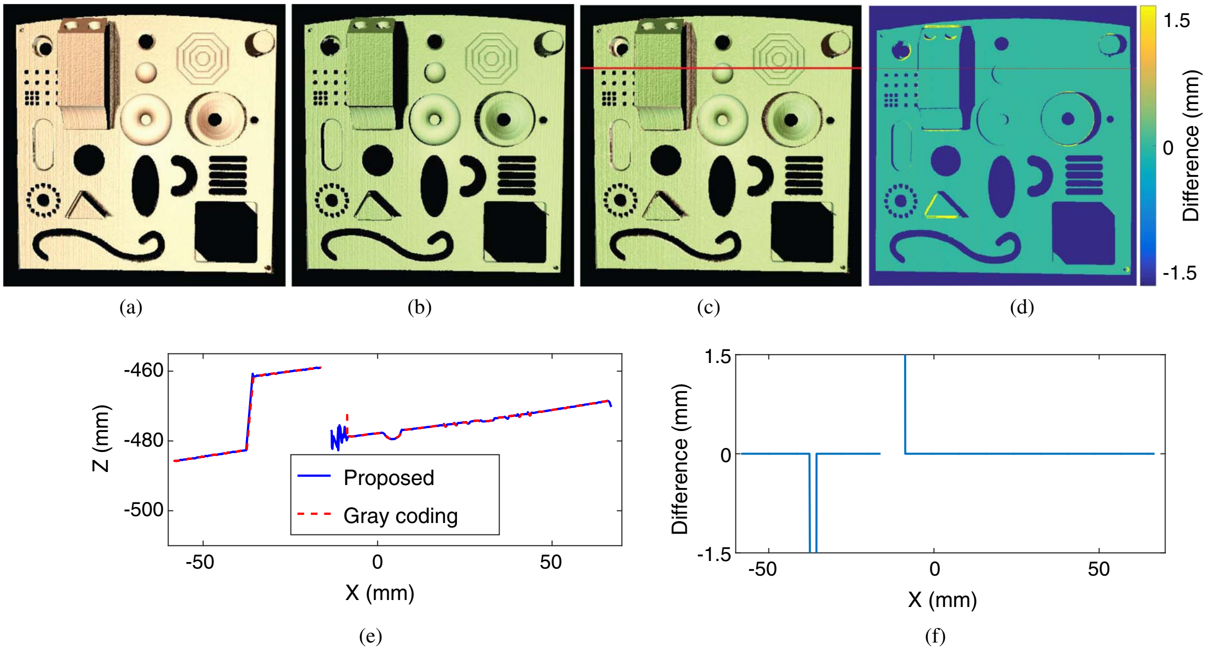

Fig. 10. Verification of absolute 3D recovery by comparing to conventional temporal unwrapping (gray coding [6]) method. (a) Reconstructed 3D using proposed method; (b) reconstructed 3D using gray coding unwrapping method; (c) overlay of 3D geometries in (a) and (b); (d) difference map of (a) and (b) (mean, 0.03 mm; RMS, 0.93 mm); (e) corresponding cross-sections (marked in red) of (c); (f) a cross-section of (d).

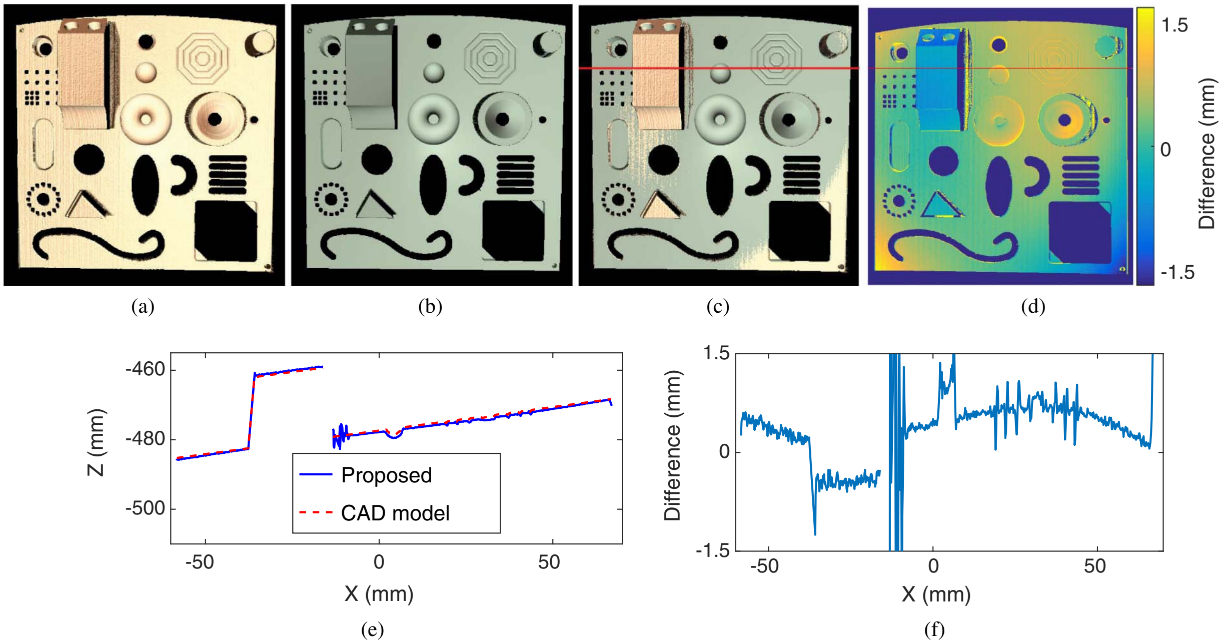

Fig. 11. Comparing reconstructed 3D geometry to the CAD model. (a) Reconstructed 3D geometry using proposed method; (b) geometry exported from CAD model; (c) overlay of 3D geometries in (a) and (b); (d) difference map of (a) and (b) (mean, 0.26 mm; RMS, 0.63 mm); (e) corresponding cross-sections (marked in red) of (c); (f) a cross-section of (d).

Figures 9(e) and 9(f), respectively, show the reconstructed 3D geometry from our proposed method and the conventional gray coding method. To visualize their difference, we rendered the two geometries using different shading colors in Figs. 10(a) and 10(b). The two geometries are overlaid in Fig. 10(c) and their difference map in shown in Fig. 10(d). For better visualization, we also plotted their corresponding cross-sections in Figs. 10(e) and 10(f), from which we can see that the two methods can generate almost identical results. The majority values on the difference map are exactly 0. Worth noting is that there are some differences on the boundaries of some sharp changing geometries caused by the different pixel resolutions and sampling between the camera and the CAD model data.

To further validate our proposed method, we also performed similar analysis by comparing our reconstructed 3D geometry with the transformed CAD 3D geometry under camera perspective, and the result is shown in Fig. 11. From the result, we can see that our reconstructed 3D geometry overall aligns well with the object CAD model with a mean difference of 0.26 mm and a root-mean-square (RMS) difference of 0.63 mm, further proving that our method indeed recovers absolute 3D geometry. The difference may come from several sources. First, manufacturing uncertainties would introduce some difference between the actual manufactured part and the CAD model. Second, the transformation of the CAD model with respect to camera perspective is estimated by a finite number of feature points, possibly still leaving room for making a much more accurate estimation. However, our method does not actually require a perfect alignment between the CAD model and the actual object. In fact, as long as the difference does not yield an incorrect fringe order determination in Eq. (21) (i.e., the difference does not exceed π in the phase domain), we can still reconstruct absolute 3D geometry with high quality.

4. CONCLUSION

We presented a novel absolute 3D shape measurement framework assisted by the CAD model of the measured objects. With the assistance of the CAD model, our proposed method uses only three phase-shifted fringe images to recover the absolute phase map and thus absolute 3D geometry of the measured object. Our method is different from existing methods in that it adaptively performs 3D reconstruction according to the pre-knowledge of the object geometry provided by the CAD model. Experimental results have demonstrated the success of our proposed framework in recovering 3D geometries of both a simple step-height object and a complex mechanical part with rich geometric variations.

Funding. Directorate for Engineering (ENG) (CMMI-1531048).

Acknowledgment. We thank our team members at Purdue University for their support and discussions. Particularly, we thank Jae-Sang Hyun for his assistance in Solidworks drawing and Bogdan Vlahov for his assistance in software development. The entire work presented in this paper was performed when Beiwen Li was with Purdue University.

REFERENCES

1. M. Takeda and K. Mutoh, "Fourier transform profilometry for the automatic measurement of 3-D object shapes," Appl. Opt. **22**, 3977–3982 (1983).
2. Q. Kemao, "Windowed Fourier transform for fringe pattern analysis," Appl. Opt. **43**, 2695–2702 (2004).
3. Q. Kemao, "Two-dimensional windowed Fourier transform for fringe pattern analysis: principles, applications and implementations," Opt. Lasers Eng. **45**, 304–317 (2007).
4. D. Malacara, ed., *Optical Shop Testing*, 3rd ed. (Wiley, 2007).
5. D. C. Ghiglia and M. D. Pritt, eds., *Two-Dimensional Phase Unwrapping: Theory, Algorithms, and Software* (Wiley, 1998).
6. G. Sansoni, M. Carocci, and R. Rodella, "Three-dimensional vision based on a combination of gray-code and phase-shift light projection: analysis and compensation of the systematic errors," Appl. Opt. **38**, 6565–6573 (1999).
7. S. Zhang, "Flexible 3D shape measurement using projector defocusing: extended measurement range," Opt. Lett. **35**, 931–933 (2010).
8. Y. Wang and S. Zhang, "Novel phase coding method for absolute phase retrieval," Opt. Lett. **37**, 2067–2069 (2012).
9. C. Zuo, Q. Chen, G. Gu, S. Feng, F. Feng, R. Li, and G. Shen, "High-speed three-dimensional shape measurement for dynamic scenes using bi-frequency tripolar pulse-width-modulation fringe projection," Opt. Lasers Eng. **51**, 953–960 (2013).
10. Y. Xing, C. Quan, and C. Tay, "A modified phase-coding method for absolute phase retrieval," Opt. Lasers Eng. **87**, 97–102 (2016).
11. Y.-Y. Cheng and J. C. Wyant, "Two-wavelength phase shifting interferometry," Appl. Opt. **23**, 4539–4543 (1984).
12. Y.-Y. Cheng and J. C. Wyant, "Multiple-wavelength phase shifting interferometry," Appl. Opt. **24**, 804–807 (1985).
13. D. P. Towers, J. D. C. Jones, and C. E. Towers, "Optimum frequency selection in multi-frequency interferometry," Opt. Lett. **28**, 887–889 (2003).
14. Y. Wang and S. Zhang, "Superfast multifrequency phase-shifting technique with optimal pulse width modulation," Opt. Express **19**, 5143–5148 (2011).
15. M. Zhao, L. Huang, Q. Zhang, X. Su, A. Asundi, and Q. Kemao, "Quality-guided phase unwrapping technique: comparison of quality maps and guiding strategies," Appl. Opt. **50**, 6214–6224 (2011).
16. X. Su and W. Chen, "Reliability-guided phase unwrapping algorithm: a review," Opt. Lasers Eng. **42**, 245–261 (2004).
17. Y. An, J.-S. Hyun, and S. Zhang, "Pixel-wise absolute phase unwrapping using geometric constraints of structured light system," Opt. Express **24**, 18445–18459 (2016).
18. M. M. M. Sarcar, K. M. Rao, and K. L. Narayan, *Computer Aided Design and Manufacturing* (PHI Learning Pvt. Ltd., 2008).
19. Z. Zhang, "A flexible new technique for camera calibration," IEEE Trans. Pattern Anal. Mach. Intell. **22**, 1330–1334 (2000).
20. T. Bell and S. Zhang, "Multiwavelength depth encoding method for 3D range geometry compression," Appl. Opt. **54**, 10684–10691 (2015).