

digital-light-processing (DLP) projector is used, the measurement speed is limited to 120 Hz, which is its highest pattern switching rate [2].

To improve the measurement speed, we have proposed the binary defocusing technique [3]. Specifically, this technique is to generate sinusoidal fringe patterns by properly defocusing the 1-bit binary structured ones. Comparing with a 8-bit grayscale sinusoidal fringe pattern, the binary structured one requires smaller data size to represent, thus allows for faster pattern switching rate. For example, the DLP Discovery D4000 can switch images at faster than 32 kHz for 1-bit binary patterns, whilst only 291 Hz for 8-bit grayscale ones. If the binary defocusing and a three-step phase-shifting algorithm are used, the achievable 3D shape measurement speed is faster than 10 kHz. In contrast, the same DLP Discovery can only achieve 97 Hz 3D shape measurement speed if 8-bit sinusoidal fringe patterns are used. With the binary defocusing technique, we have successfully achieved 3D shape measurement at an unprecedentedly high speed: 667 Hz [4]. Instead of 10 kHz, 667 Hz speed was limited by the projection source intensity rather than by the capability of the projector or the camera.

The superfast 3D shape measurement system [4], however, cannot be used to measure multiple moving objects because a single-wavelength three-step phase-shifting algorithm is used. Adding additional coded patterns for fringe order determination has the potential to resolve this problem because it can obtain absolute phase point-by-point [5]. But, because the defocusing technique is used, the challenge becomes more complicated. For example, Zhang has demonstrated that by combing with simple binary coding with phase shifting, the point-by-point 3D shape measurement could be realized [6]. However, because the binary coding technique used was not based on gray code, the decoding process has significant errors due to defocusing. To address this problem, we adopt the gray code [7] in this research. The gray code changes only one bit at one position. However, we found even the gray code has problems for rapid motion capture. This is because when the object is moving rapidly, there might be significant displacement between the first captured pattern and the last one. Thus the precisely designed code will have misalignment, which may introduce severe errors in the final reconstructed results.

In this paper, we propose a new phase unwrapping strategy that combines the conventional spatial phase unwrapping algorithm with the gray code to solve the motion induced phase unwrapping problems. The gray code is first used to segment the whole image into separate regions. Within each region, the phase differences between neighboring pixels are no more than 2π . Then a conventional unwrapping algorithm is applied to unwrap the phase for each region to obtain relative phase. Finally, the gray code is again used to obtain absolute phase by properly shifting the relative phase for each region by adding multiples of 2π . By this means, the phase unwrapping problems caused by motion could be resolved. To verify the performance of the proposed method, a system with a speed of 5,000 Hz was developed to capture the collision process of two balls.

Section 2 explains the principle of the absolute phase retrieval by combining gray coding with phase shifting. Section 3 presents the proposed framework to solve the motion induced phase unwrapping problems. Section 4 shows experimental results, and finally Section 5 summarizes this paper.

2. Principle of phase-shifting technique

2.1. Three-step phase-shifting algorithm

Phase-shifting methods are widely used in optical metrology because of their speed and accuracy [8]. We use a three-step phase-shifting algorithm to find the phase value. Three fringe

images with a phase shift of $2\pi/3$ can be written as:

$$I_1(x, y) = I'(x, y) + I''(x, y) \cos(\phi - 2\pi/3), \quad (1)$$

$$I_2(x, y) = I'(x, y) + I''(x, y) \cos(\phi), \quad (2)$$

$$I_3(x, y) = I'(x, y) + I''(x, y) \cos(\phi + 2\pi/3). \quad (3)$$

where $I'(x, y)$ is the average intensity, $I''(x, y)$ the intensity modulation, and $\phi(x, y)$ the phase to be solved for. The phase can be solved for from these equations as

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

This equation provides the wrapped phase ranging from $-\pi$ to $+\pi$ with 2π discontinuities. The 2π phase jumps can be removed to obtain a continuous phase map by adopting a phase unwrapping algorithm [9]. Specifically, the phase unwrapping is to determine fringe order, integer $k(x, y)$, so that

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

Here, $\Phi(x, y)$ is the unwrapped absolute phase. However, such a single three-step phase-shifting algorithm cannot be used to measure multiple objects at a time.

2.2. Fringe order $k(x, y)$ determination with gray code

To simultaneously measure multiple objects, a binary coding method can be used for absolute phase extraction. Integer number $k(x, y)$ in Eq. (5) can be uniquely defined by a sequence of binary (1-bit) patterns. Figure 1 illustrates two coding examples. It can be seen that for each fringe period, a unique bit sequence (e.g., 0001, 0011) is used. However, due to the projector defocusing and discrete camera sampling, the binary state changes in coded patterns become blurred, and this makes the code boundaries difficult to be determined accurately. This problem will be worse if the state changes at the same place for different coded patterns, as illustrated in Fig. 1(a): all four patterns change from code 0111 to 1000.

To reduce this problem, we adopt a gray code [7] as illustrated in Fig. 1(b). Gray code is to ensure that at each point there is just one bit change for all the coded patterns. Using this new coding method, the codeword-detecting errors could be reduced and the unwrapping result can be improved. However, when the objects are moving fast, the motion could cause additional phase unwrapping problems even if the gray coding patterns are used. There will be misalignment between the extracted wrapped phase and precisely designed codeword. Therefore, severe errors will arise near the jump edges of codeword, which will be illustrated in the Sec. 4.

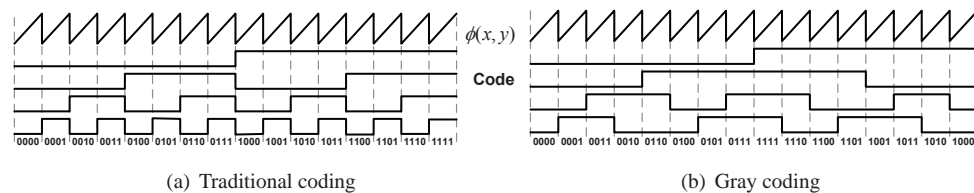


Fig. 1. Gray Vs non-gray coding for point-by-point phase unwrapping.

2.3. Phase-to-depth conversion

After obtaining the absolute phase $\Phi(x, y)$, the depth distribution can be retrieved by adopting a phase-to-height conversion algorithm. In this paper, a reference-plane-based method is adopted

to convert absolute phase to depth z [10]. In brief, assuming $\Phi(x, y)$ is the object phase map, and $\Phi^{rP}(x, y)$ is the reference phase map that is obtained by measuring a uniform flat planar surface, the relationship between the surface height relative to the reference plane and their phase difference, $\Delta\Phi(x, y) = \Phi(x, y) - \Phi^{rP}(x, y)$, is approximately proportional, that is

$$\Delta z \propto \Delta\Phi(x, y) = \Phi(x, y) - \Phi^{rP}(x, y). \quad (6)$$

Assuming the reference plane has a depth of $z_0 = 0$, the absolute depth value for each camera pixel can be represented as

$$z(x, y) = z_0 + c_0 \times [\Phi(x, y) - \Phi^{rP}(x, y)] = c_0 \times [\Phi(x, y) - \Phi^{rP}(x, y)], \quad (7)$$

where c_0 is a constant that can be determined through calibration.

3. Novel phase unwrapping framework

To tackle the phase unwrapping problems caused by motion, a novel phase unwrapping framework is proposed. Specifically, we combine the conventional spatial phase unwrapping method with the gray coding method. The framework can be implemented in the following steps:

Step 1: Use the codeword to segment the whole image into separated regions. For each point, we can detect the codeword jump by comparing it with neighboring points. For each region, the codeword does not change more than 1, which means that the phase difference between neighbor pixels should not be more than 2π . If the jump of one point is more than 1, meaning that the phase difference of the point might be more than 2π , this point is treated as a discontinuous edge. The entire image is then divided into continuous regions separated by the edges.

Step 2: Adopt a conventional phase unwrapping algorithm to unwrap the phase for each region to obtain the relative phase map. When the phase map is continuous in one region, a conventional unwrapping algorithm can be applied to obtain the continuous phase map. Because each region is unwrapped individually relative to one point in that region, the unwrapped phase obtained in this step is called *relative phase*, $\Phi^r(x, y)$.

Step 3: Use the codeword to adjust the relative phase maps to be absolute ones for all regions. Step 2 only provides relative phase for each region. In order to measure multiple objects simultaneously, *absolute phase* is needed. To do so, we first use the conventional approach to extract the absolute phase map $\Phi^0(x, y)$. For each region, the difference between the absolute phase $\Phi^0(x, y)$ and the relative phase $\Phi^r(x, y)$ obtained in Step 2 should be a constant, $\Delta\Phi(x, y) = \Phi^r(x, y) - \Phi^0(x, y) = \Delta k(x, y) \times 2\pi$. Here, $\Delta k(x, y)$ is an integer and constant for each region. The difference values should be nearly the same for each region except for a few noisy points. To obtain $\Delta k(x, y)$, we can first average all the phase difference values in one region after applying a low-pass filter, and then divide the average value by 2π and round it to an integer for each region. Once $\Delta k(x, y)$ is obtained, the relative phase can be converted to absolute one by adding $\Delta k(x, y) \times 2\pi$.

Step 4: Combine different region phase maps into a single one and convert the final unwrapped phase map to 3D shape. This step will generate the final complete absolute phase map $\Phi(x, y)$ by combining all regions together. Once the absolute phase map is obtained, 3D shape can be recovered if the system is calibrated.

4. Experimental results

We developed a superfast system that is composed of a DLP Discovery projection system, a high-speed CMOS camera, and a self-developed synchronization circuit. The DLP Discovery projection system includes a DLP Discovery board (D4000) (Texas Instruments) with an ALP

High Speed (Digital Light Innovations). The camera used in this system is Phantom V9.1 (Vision Research). The synchronization circuit takes the projection timing signal and sends the trigger signal to the camera for simultaneous image acquisition.

Experiments were conducted to capture the collision of two balls. For each 3D frame, four binary coded patterns and three phase-shifted fringe patterns are projected. In this research, the patterns are continuously captured at 5,000 Hz with a 480×480 image resolution. The exposure time used is $200 \mu\text{s}$. In the experiment, both balls are suspended by strings, and initially in contact with one another. Then, one ball is pulled to a certain height and released to swing like a pendulum until collision occurs with the stationary ball, at the bottom of the pendular arc where the moving ball's velocity is at its maximum. Figure 2(a) shows one fringe pattern and Fig. 2(b) shows one of the coded binary patterns. From the four coded binary patterns, the codeword $k(x, y)$, which represents the fringe order of each point, can be extracted as shown in Fig. 2(c). From the three phase-shifted fringe patterns, the wrapped phase map can be obtained as shown in Fig. 2(d). The codeword can be used to unwrap the phase point-by-point using Eq. (5). Figure 2(e) shows the result. It clearly shows some problems on the unwrapped phase (e.g., stripes and spikes). To highlight these problems, Fig. 2(f) shows a view zoomed into a small area. This figure clearly shows that the problematic areas are more than 2 pixels wide. This means that a standard filtering technique cannot be used to eliminate this problem. Since the object is moving rapidly, the coded patterns and the phase-shifted fringe patterns are captured sequentially, and the time lag between the first frame and the last frame is 1.4 ms, the displacement between the first captured fringe pattern and the last one could be significant. Thus the precisely designed codeword and the wrapped phase do not match, and the phase cannot be unwrapped correctly by the coded patterns.

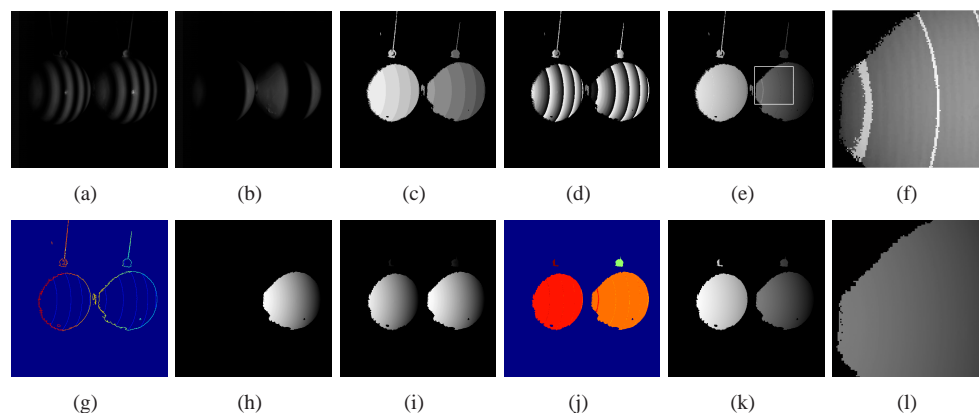


Fig. 2. (a) One fringe pattern; (b) One binary pattern; (c) Fringe order $k(x, y)$ extracted from four coded patterns; (d) Wrapped phase $\phi(x, y)$ extracted from three phase-shifted fringe patterns; (e) Absolute phase $\Phi^0(x, y)$; (f) Zoom-in view of the area within the white window of (e); (g) Codeword change map; (h) Unwrapped phase for one region by a conventional unwrapping algorithm; (i) Unwrapped relative phase of all regions $\Phi^r(x, y)$; (j) Difference between relative phase map in (i) and absolute phase map obtained traditionally (e); (k) Absolute phase $\Phi(x, y)$ after Step 3; (l) The corresponding area shown in (f).

Now we apply the novel phase unwrapping framework we proposed in Sec. 3. Figure 2(g) shows the map of codeword changes which can be used to segment the images into different regions. By applying a conventional phase unwrapping algorithm to each region, the continuous phase map can be obtained. One of the regions is illustrated in Fig. 2(h). However, because

each region is unwrapped relative to one point within that region, we only obtain relative phase instead of absolute one. For example, Fig. 2(i) shows the unwrapped phase for two regions, it clearly shows that they are not absolute because the phase is not monotonic along x -axis. The relative phase map $\Phi^r(x,y)$ can be converted to absolute one by referring to the absolute phase map $\Phi^0(x,y)$ obtained by using a conventional approach as shown in Fig. 2(e). Figure 2(j) shows the difference phase map $\Delta\Phi(x,y) = \Phi^r(x,y) - \Phi^0(x,y)$. This figure indicates that the difference values indeed are nearly the same for each region except for a few noisy points. Once the constants for each region is determined, the relative phase can be converted to be absolute one as show in Fig. 2(k). Because the relative phase map for each region is obtained through a conventional phase unwrapping algorithm, it should be smooth. Figure 2(l) shows the corresponding area of the absolute phase as shown in Fig. 2(f). It can be seen that the noisy areas are gone, and the phase map is smooth.

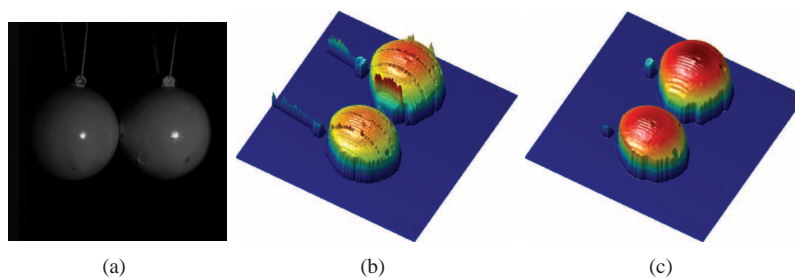


Fig. 3. (a) Photograph of one frame;(b) 3D result using the conventional method; (c) 3D result by the proposed phase unwrapping framework. ([Media 1](#), [Media 2](#), [Media 3](#), and [Media 4](#)).

Figure 3 shows the experimental result for one frame. Figures 3(a), 3(b) and 3(c) respectively show the photograph of the frame, the 3D reconstructed results using the conventional and the proposed technique. It clearly indicates that the spike and large noisy points disappear if the new method is adopted. This means that the new phase unwrapping method can resolve the phase unwrapping problem caused by motion. Several videos are also submitted together with this paper. [Media 1](#) shows the full speed video ($5000/7 = 714$ Hz) of the collision process. [Media 2](#), [Media 3](#), and [Media 4](#) shows the video at 60 Hz viewing from different angles. Our experiment shows that the collision process of two balls are well captured and 3D information are correctly recovered by adopting the new phase unwrapping strategy and thus verifies the success of the proposed method.

5. Conclusion

This paper has presented a novel and efficient phase unwrapping strategy to solve phase unwrapping problems associated with 3D shape measurement of moving objects. Experimental results demonstrate that the proposed method can overcome motion-induced phase unwrapping problems for simultaneously measuring multiple rapidly moving objects. However, the proposed method is not trouble free. The assumption of the framework is that within each region there are no step-height changes between π and 2π . Although this is not likely to happen in most measurement scenarios, if it becomes a problem for some particular applications, various image processing means could be adopted to address it.

It is important to note that this proposed method only solves for the phase unwrapping error caused by motion, but cannot reduce the inherent phase error caused by motion. For this system with 5,000 Hz capture speed and $200 \mu\text{s}$ exposure time, we assume the object is motionless

during three phase-shifted fringe images capture, i.e., the phase error caused by $600 \mu\text{s}$ time lag will not be alleviated. Even with this limitation, the proposed system has achieved a 3D shape measurement with an equivalent shutter speed of 1,667 Hz or $600 \mu\text{s}$ shutter time, which can be used to capture relatively rapidly moving objects.