Beyond the interference problem: hierarchical patterns for multiple-projector structured light system

Zengqiang Yan,1,2 Li Yu,1,2 You Yang,1,2,* and Qiong Liu1,2
1Department of Electronics and Information Engineering, Huazhong University of Science and Technology, Wuhan 430074, China
2Wuhan National Laboratory for Optoelectronics, Wuhan 430074, China
*Corresponding author: yangyou@hust.edu.cn

Received 20 February 2014; revised 25 April 2014; accepted 28 April 2014; posted 29 April 2014 (Doc. ID 205609); published 4 June 2014

Three-dimensional reconstruction of a dynamic object based on the structured light (SL) technique has attracted intensive research. Since a single projector only covers a limited area of the scene, multiple projectors should be employed simultaneously to extend the imaging area for a 3D object. However, patterns projected by different projectors superpose each other in a light-intersected area, which makes it difficult to recognize the original patterns and obtain the correct depth maps. To solve such a problem, we propose a method to design hierarchical patterns that can be separated from each other. In the proposed patterns, each pixel in binary patterns based on the de Bruijn sequence is replaced by a different bin with limited size. Then the proposed patterns can be separated by identifying distributions of colors in each bin in superposed patterns, and depth maps are obtained by decoding the separated patterns. To verify the performance of the proposed method, we design two hierarchical patterns and conduct several experiments in different scenes. Experimental results demonstrate that the proposed patterns can be separated in a multiple-projector SL system to obtain accurate depth maps, and they are robust for different conditions. © 2014 Optical Society of America

OCIS codes: (110.6880) Three-dimensional image acquisition; (120.6650) Surface measurements, figure; (150.6910) Three-dimensional sensing; (100.5010) Pattern recognition; (330.5000) Vision - patterns and recognition.

http://dx.doi.org/10.1364/AO.53.003621

1. Introduction

3D reconstruction of dynamic object has been extensively researched for applications such as motion detection and facial expression tracing. The critical step is a speed, dense, and accurate measurement to obtain the depth maps. Many passive and active methods have been proposed to obtain depth information of 3D object from single viewpoint, especially stereo vision (SV) based techniques [1] and structured light (SL) based techniques [2]. Since SV techniques have difficulty in reconstructing texture-less surfaces densely and accurately, SL systems have been extensively studied for 3D reconstruction.

A typical SL system contains one projector and one camera as shown in Fig. 1. The projector emits an SL pattern, and the camera captures the scene illuminated by the pattern. Thus the depth map can be obtained by analyzing the deformation of the pattern on the surface of the object. The pattern used in the SL system is designed so that code words are assigned to a set of pixels, thus there is direct mapping from every code word to its corresponding coordinates of the pixel in the projected pattern. Based on the codification strategies, SL can be roughly
divided into two categories: time multiplexing (TM) and spatial neighborhood (SN) codifications. Phase shifting [3] has been recognized as an efficient TM technique that allows automated fringe-pattern processing and analysis. The main idea is to produce different fringe patterns with a constant phase shift between consecutive frames. By collecting all frames by the camera, mapping between the phase and the corresponding coordinates is formed. Many schemes of the phase-stepping algorithms have been developed [4–6], and accurate depth maps can be obtained. However, TM is not suitable for a dynamic object due to that TM calls for multiple frames to calculate depth values. SN utilizes spatial neighbor pixels in the pattern to encode each pixel, so that SN needs only one pattern to obtain depth maps. Though accuracy of depth maps obtained by SN is not as good as that obtained by phase shifting, SN has been widely used for 3D reconstruction of a dynamic object such as a color code pattern [7], pseudorandom binary-array based pattern [8], and others [9].

Since that single depth map only contains parts of a scene, several depth maps from different viewpoints are needed to make complete 3D reconstruction. Furthermore, for a dynamic object, several SL based systems are required to work simultaneously to capture the multiview depth videos. However, patterns from different projectors superpose each other in light-intersected areas as shown in Fig. 2. The interference in the superposed patterns makes it extremely difficult to extract the original patterns from the captured image. Once correct original patterns on the surface of the object are not available, depth maps cannot be obtained correctly. Thus the interference problem has hindered multiple-projector structured light (MSL). This motivates our work presented in this paper.

Several methods that are based on hardware modification have been proposed to address the interference problem. By having the SL patterns masked outside the outlines of the target object, Griesser et al. [10] established a system to eliminate interference from opposite patterns. Maimone et al. [11] applied a small amount of motion to a subset of the sensors, thus interference from other patterns became blurred and could be removed. However, hardware modification based methods make the SL system complex. Some postprocessing methods also have been used to enhance the depth maps obtained by the MSL system. Among these methods, Wang et al. [12] presented a plane sweeping based algorithm to reduce the interference, and the method could effectively improve the quality of depth maps. The main disadvantage of a plane sweeping algorithm is its high time complexity, which is not suitable for dynamic objects. Besides, the authors of [13] used color-stripe permutation patterns and eliminated the interference by analyzing the behavior of superposed-light colors in a chromaticity domain. Since color-stripe patterns contain several different colors, such patterns are sensitive to the illumination. These works make a significant achievement in solving the interference problem. Different from the above researches, we focus on the root of the interference problem: superposed patterns.

In this paper, we propose a method to design hierarchical patterns that can be separated when such patterns superpose each other. We first design binary patterns based on de Bruijn sequences [14] so that coordinates of each pixel can be labeled by its neighbors. Then each corresponding pixel in the binary patterns is replaced by a different bin with limited but uniform size. In the proposed method, each bin is a pseudorandom array [15] with limited size. The proposed patterns can be separated from the superposed patterns by identifying different distribution of colors in each bin. Then the depth of each bin can be calculated through its neighbors in the de Bruijn sequence, and the depth of each pixel in the bin can be obtained based on its neighbors in a pseudorandom array. Furthermore, the proposed method can be used to generate multiple hierarchical patterns by adjusting the sizes of the de Bruijn sequence and bin to satisfy the requirement of N-views depth video. In this way, there is no cost for the hardware modification, and we avoid the depth artifacts from the possible postprocessing for interference reduction. Besides, in the proposed patterns both binary patterns and bins consist of few colors, so that proposed patterns have relatively strong robustness to the illumination. To validate the proposed method, we design two hierarchical patterns and conduct several different experiments. Experimental results also show that the hierarchical patterns can be separated, and depth maps from multiple viewpoints are obtained. We have made contributions on two aspects as follows:

1. We propose a method to design hierarchical patterns for interference cancellation in the MSL system. The proposed hierarchical patterns can be used for 360 deg 3D data acquisition, especially for dynamic objects.
2. Different from traditional SL patterns, we utilize multiple coding strategies to design the
hierarchical patterns including the de Bruijn sequence and pseudorandom array. This coding strategy improves the robustness of the proposed hierarchical patterns.

The rest of this paper is organized as follows. In Section 2, we analyze the interference problem in the MSL system. Section 3 presents our method to design the hierarchical patterns. An algorithm to decode the proposed patterns and obtain depth information is provided in Section 4. Experimental results are shown in Section 5. We conclude this paper in Section 6.

2. Problem Statement

In this section, we analyze the measurement principle of the SL technique first to clarify the interference problem. Figure 3 shows the measurement principle of the SL technique in [16]. The SL system is based on projection pattern onto an object and measures its surface by imaging the illuminated object with camera. Once disparity of a certain pixel between the captured image and the original pattern is obtained, Eq. (1) is used to calculate its depth information:

\[ h = \frac{\text{Disparity} \times S}{L + \text{Disparity}}. \]  

In order to obtain disparities, the pattern projected in the SL system is well designed so that each pixel has a code word. Thus there is direct mapping from every code word to its corresponding coordinates of the pixel in the original pattern. Once we get the code word of a certain pixel in the captured image, its coordinates in the original pattern can be obtained by decoding its code word. Then the disparity of the pixel is figured out by calculating the distance between its coordinates in the captured image and that in the original pattern.

However, when several patterns superpose each other, the interference from other patterns makes it difficult to identify the code words of pixels. Then the depth value cannot be obtained correctly without accurate code words of these pixels.

Figure 4 shows the severe errors caused by the interference problem. In this scene, two projectors and two cameras are placed at different positions.

The interference problem mainly exists on the left parts of the teapot. A depth map obtained from one viewpoint is shown in Fig. 4. When two patterns superpose each other, the code word of a certain pixel in the captured image might be ambiguous, therefore the depth information of the pixel cannot be obtained. Depth values of the pixels are manually set to be 0 to distinguish such pixels. For example, the depth map in Fig. 4(b) contains more holes caused by the interference problem because the ambiguous code words of the pixels are impossible to decode. According to the analysis above, the misidentification of code words of pixels in the superposed patterns is the source of the interference problem in MSL system. Thus the best way to eliminate the interferences is to design patterns in which each code word can be identified correctly from the superposed patterns.

3. Pattern Encoding

In this section, the method to design the hierarchical patterns is provided. Our design method can be divided into three steps as follows:

1. Binary pattern encoding: Two binary patterns based on the de Bruijn sequence are designed. In binary patterns, coordinates of each pixel can be calculated through its five neighbor pixels in the horizontal direction.

2. Bin-level binary patterns encoding: Two pairs of bins with different distributions of gray values are designed to replace corresponding binary pixels in the binary patterns above. In the bin-level binary patterns, coordinates of each bin can be obtained through its five neighbor bins.

3. Hierarchical patterns encoding: Each bin in the bin-level binary patterns is assigned with an unique code word that corresponds to its position in the pattern. Besides, pseudorandom array is introduced to improve the precision of the proposed hierarchical patterns.

More hierarchical patterns can be obtained by increasing the sizes of the de Bruijn sequence and the bin. Next we show our method to design two hierarchical patterns. In the rest of Section 3, we use left
pattern and right pattern to represent the patterns projected by left and right projectors.

A. Binary Pattern Encoding

A de Bruijn sequence of order \( n \) has a window property that each subsequence of length \( n \) appears only once. Thus coordinates of each element in the whole de Bruijn sequence can be labeled by its \( n \) neighbor element. Based on the algorithm in [14], a de Bruijn sequence \( V_n \) with length of 31 is generated:

\[
V_n = [100011111010100100011011011101]. \tag{2}
\]

In binary sequence \( V_n \), each subsequence \( V_i \), which consists of five consequent elements, appears only once. Thus the position of each element can be figured out through its five sequent elements. In order to generate two different binary patterns that contain the same sequence \( V_n \) in the horizontal direction, two different sequences, \( V_{\text{left}} \) and \( V_{\text{right}} \), are applied as vertical constraints to the patterns.

If \( i \) is the row number and \( j \) is the column number, then gray value \( a_{i,j} \) in the binary patterns can be calculated by Eq. (3):

\[
a_{i,j} = 255 \times \text{mod}(V_{i-\lfloor(j-1)/31\rfloor \times 31} + V'_{j-\lfloor(i-1)/4\rfloor \times 4}, 2), \tag{3}
\]

where \( V_{(k)} \) is the value of \( k \)th element in \( V_n \), and \( V'_{(k)} \) represents the value of \( k \)th element in \( V_{\text{left}} \) or \( V_{\text{right}} \). The binary patterns based on Eq. (3) are shown in Fig. 5.

B. Bin-Level Binary Pattern Encoding

In this stage, we first design two pairs of bins with different distributions of gray values. Since the \( 3 \times 3 \) matrix has an unique central pixel to identify the position of each element, it is enough to generate four different bins to replace corresponding pixels in two binary patterns above. Thus the size of bins in the proposed patterns is set up to \( 3 \times 3 \). Size of the bin should be increased when more patterns are needed. We first design two pairs of bins with different distributions of gray values as shown in Fig. 6.

Then each binary pixel in the predesigned left and right binary patterns is replaced by a corresponding bin above. \( \text{left} \_0 \) and \( \text{right} \_0 \) are used to replace the pixels with gray value 0 in the left and right binary patterns, while \( \text{left} \_1 \) and \( \text{right} \_1 \) are used to replace the pixels with gray value 255. Parts of the left and right bin-level binary patterns are shown in Fig. 7(a).

As shown in Fig. 7(b), no matter how the left and right bin-level patterns superpose each other, the interference from the right pattern can never change \( \text{left} \_1 \) into \( \text{left} \_0 \). Then \( \text{left} \_1 \) and \( \text{left} \_0 \) can be separated by identifying different distributions of gray values. Thus coordinates and depth values of the bins can be obtained by referring its five neighbor bins in the horizontal direction.

When the pixels in the superposed pattern are not aligned, the interference problem is changed into the subpixel interference problem. In the subpixel interference problem, the gray value of certain pixel cannot be directly classified into 0 or 255, which is based on a fixed threshold because the gray value might be
in the range from 0 to 255. In the proposed method, the threshold to classify the gray value of each pixel is set as a variable according to the average gray value of the bin. Then the pixels whose gray values belong to the top three minimum gray values, and below the threshold are classified as the pixels with gray value 0. Otherwise, the pixels are classified as the pixels with gray value 255 in the bin. After that, the current bin can be identified based on its distributions of gray values. Besides, the two sequences $V_{\text{left}}$ and $V_{\text{right}}$ in Section 3.A are used to ensure the correctness of the identification of each bin. Taking the left bin-level binary pattern for example, once the current bin is identified as $\text{left}_0$, the binary values represented by its three neighbor bins in the vertical direction are calculated and a four-tuple binary sequence is obtained. If the four-tuple binary sequence is equal to $V_{\text{left}}$, the current bin can be identified as $\text{left}_0$. Otherwise, the current bin is identified as $\text{left}_1$.

C. Hierarchical Pattern Encoding

According to the analysis above, each bin in the superposed pattern can be separated based on its distributions of gray values. Then the position of each bin in the bin-level patterns can be calculated through its five neighbor bins. However, in some regions of the captured images, available neighbors for a certain bin may be less than five. Positions of such bins cannot be calculated through neighbors. To address such a problem, each bin in the patterns is assigned with an unique code word. In the proposed method, four pixels according to Eqs. (4) and (5) as well as the central pixel of a bin are used to form its code word. Then five pixels of each bin are used to form its code word and total $2^5$ code words are available. By assigning different sequences $P_{\text{left}}$ and $P_{\text{right}}$ to the bins in the left and right bin-level binary patterns in Fig. 7(a), the bin-level patterns are changed into the hierarchical patterns as shown in Fig. 8(b). In the hierarchical patterns, the code word of each bin is different in the horizontal direction:

$$P_{\text{left}} = \{[a, b, c, d]|a, b, c, d \in \{\text{white, red}\}\},$$

$$P_{\text{right}} = \{[a', b', c', d']|a', b', c', d' \in \{\text{white, blue}\}\}.$$  

However, if only the position or coordinates of the central pixel in a bin is obtained, the resolution of the proposed patterns would be rather low. Then the pseudorandom array is introduced to guarantee that positions of all the pixels in a bin can be calculated independently.

Since the pseudorandom array has a property in which the position of each element can be labeled by its neighbors in a small window, it can be used to improve the resolution of the proposed patterns. In our method, the color red is assigned to the lower left pixel in each bin of the left hierarchical pattern while the color blue is assigned to the lower right pixel in each bin of the right pattern. Then each $2 \times 2$ window in a certain bin appears only once. The relative position between certain pixel and the central pixel can be calculated according to its neighbor pixels in $2 \times 2$ window. Besides, any $2 \times 2$ windows in two neighboring bins are different from each other to improve the robustness of the proposed patterns. Parts of the proposed hierarchical patterns are shown in Fig. 8(b).

4. Pattern Decoding

In this section, the algorithm to obtain the separated patterns and depth maps is explained in detail. Flow chart of the proposed method to separate the hierarchical patterns and obtain depth maps is shown in Fig. 9. Each step in our algorithm is based on the property of the hierarchical patterns shown in Section 3. Similar to the encoding process, the decoding process also is divided into the bin-level decoding process and pixel-level decoding process.

A. Bin-Level Decoding

In our algorithm, colors red and blue are first separated from the superposed patterns. After that, the captured images are converted into gray scale. Then the left and right hierarchical patterns are separated by identifying different distributions of gray values in each bin. Once the original patterns are separated from the superposed patterns, the position of a certain bin can be obtained based on its five neighbor bins. In some small regions where available neighbors for a certain bin are less than five, the code word of the bin is used to identify its position.
B. Pixel-Level Decoding

Once the position of a bin is obtained, coordinates of all pixels in the bin are figured out based on each $2 \times 2$ window. However, in separated patterns, there exists some incomplete bins whose size is less than nine, and these bins cannot be decoded through its neighbors or code words. Then a refinement to the separated patterns decoding is proposed to solve such a problem. Since all $2 \times 2$ windows in two neighboring bins are different from each other, original patterns without deformation are used to refine such incomplete bins based on its $2 \times 2$ windows.

Comparing the depth maps without refinement (decoding results) with the depth maps after refinement (depth maps) in Fig. 9, we can find that the refinement processing can effectively obtain depth values of the most incomplete bins in the separated pattern.

5. Experiments and Results

In this section, several experiments have been conducted in different scenes established by 3ds Max [17]. Since the models created by the Stanford University Computer Graphics Laboratory have been widely used as the input for surface reconstruction algorithms, they can effectively verify the performance of the proposed method. In our experiments, the Stanford Bunny, Stanford Lucy, and Stanford Happy Buddha are involved. Besides, MeshLab software [18] is used to make 3D point cloud reconstruction.

In the following subsections, we show our experimental results in different scenes. The arrangement of projectors and cameras for a dynamic object is shown in Fig. 10. Subjective and objective contrast experimental results are shown in Section 5.C.
the whole object, depth maps obtained by decoding
the two patterns are taken for merging.
As shown in depth maps (c) and (f) in Fig. 14, there
still exists several pixels with the wrong depth infor-
mation, especially in regions with a sudden change.
Code words of such bins are changed into other code
words. Thus depth values of such pixels turn out to
be wrong. This problem can be avoided by increasing
the Hamming distance between different code words
of bins.

C. Contrast Experiment
In our contrast experiments, we mainly focus on the
measurement range and accuracy of depth maps for
different methods. We conduct experiments using
two synthetic scenes, Stanford Bunny and Bonsai,
rendered by software 3ds Max to validate the perfor-
mance of different methods. In such two experimen-
tal scenes, Bunny contains complex depth changes
due to its curved surfaces, and Bonsai has some
details, especially in some leaves. Thus the contrast
experiments can effectively test the feasibilities
and the robustness of different methods. The ar-
rangement of projectors and cameras is shown in
Fig. 15. In each scene, the projector is simulated
by a spotlight, and the camera is placed 7 cm away
from the projector.
We implemented some existing methods for perfor-
mance comparison as follows:

1. SL based SV method by computing the mean-
removed cross correlation (MRCC) between the pro-
jected pattern and the captured image [14]: In the
MRCC algorithm, the size of sliding window in the
stereo matching algorithm is 7 × 7, and the minimum
correlation required is 0.5. The patterns and the
captured images in the MRCC algorithm are shown
in Fig. 16.

2. Plane sweeping algorithm [14]: In order to re-
cover the depth information for both overlapped and
nonoverlapped regions in the MSL system, the plane
sweeping algorithm calculates the correlation be-
tween multiple projectors and the captured images
as well as the correlation between the captured im-
gages. In our contrast experiment, the patterns and
the captured images in two viewpoints are available
to reconstruct depth maps obtained by the above
MRCC algorithm.

Figures 17 and 18 show the performances of
different methods. Figures 17(a) and 18(a) are depth
maps obtained by the MRCC algorithm without
the interference brought by the other projector.
Figures 17(b) and 18(b) show depth maps obtained by the proposed hierarchical patterns when there exists no interferences. Thus depth maps of high quality can be obtained from both viewpoints. However, when different patterns superpose each other in light-intersected areas, MRCC scores between the captured image and the known projector pattern, for those light rays might below a fixed threshold 0.5. As a result, depth maps in Figs. 17(c) and 18(c) contain plenty of holes that cannot be used for 3D reconstruction directly. Figures 17(d) and 18(d) show the results by applying a hole-filling method based on a plane sweeping algorithm to depth maps in Figs. 17(c) and 18(c). By comparing the depth maps in Figs. 17(d) and 18(d) with those in Figs. 17(c) and 18(c), we find that the majority of holes can be effectively filled by the depth information from the other viewpoint, and the quality of depth maps is significantly improved. But there still exists a considerable number of pixels without depth values; due to this, the mean-removed cross correlations between different
captured images or patterns are still rather low in regions with a sudden change of depth. Depth maps obtained by the proposed hierarchical patterns are shown in Figs. 17(e) and 18(e), and these depth maps contain fewer pixels without depth values than those in Figs. 17(d) and 18(d). However, depth maps in Figs. 17(e) and 18(e) still contain some pixels without depth values mainly in edge regions. Such problem can be improved by applying a simple median filter as shown in Figs. 17(f) and 18(f).

We also calculate the peak signal-to-noise ratio (PSNR) and bad point ratio (BPR) of the depth maps as shown in Tables 1–4. Experimental results show that the proposed patterns can effectively eliminate the interference and obtain depth maps of high quality. PSNR of the depth maps obtained by the
The proposed hierarchical patterns is better than those by plane sweeping algorithm. When calculating BPR between each depth map and the ground true depth map, the maximum gray value difference allowed for each pixel is set as 2. By comparing the BPR results between different depth maps, we can find that the proposed hierarchical patterns can effectively improve the depth maps in the MSL system.

Figure 19 shows the depth maps obtained by the hierarchical patterns and the MRCC algorithm. In the pattern used for the MRCC algorithm, the depth value of each pixel is calculated based on its neighbor pixels in a $7 \times 7$ window. In other words, the precision of the pattern is $7 \times 7$. Once the depth value of a certain pixel is different from most of its neighbor pixels in the $7 \times 7$ window, the correlation calculated...
by MRCC algorithm is lower than the minimum threshold 0.5, and its depth value cannot be obtained. Compared with the patterns used in the MRCC algorithm, the minimum precision of the hierarchical pattern is much higher as discussed in Section 3. In the hierarchical pattern, the depth value of a certain pixel is missing only when its depth value is different from its neighbor pixels in a $2 \times 2$ window. As a result, depth maps labeled by the white boxes in Fig. 19(b) contain more pixels without depth values. Thus the accuracy of a depth map obtained by the pattern used in the MRCC algorithm is lower than that obtained by the hierarchical pattern.

By comparing the results between the proposed hierarchical patterns and the plane sweeping algorithm, our method has the following advantages:

1. Better quality of depth maps: As shown in contrast experiments, depth maps obtained by the proposed patterns contain fewer wrong pixels than those by the plane sweeping algorithm.

2. Lower complexity: The proposed method is based on the spatial-neighborhood SL technique, and depth maps are obtained by decoding the captured images directly. Plane sweeping algorithm can be regarded as a stereo matching algorithm that consumes more time to calculate correlation of each pixel. Thus the proposed method has better real-time quality, which is useful for a dynamic object.

3. Better flexibility: Plane sweeping algorithm makes use of the captured images and the patterns from other viewpoints to reconstruct depth maps of the current viewpoint. Thus the quality of depth maps depends on the depth maps from other viewpoints. Different from the plane sweeping algorithm, each proposed hierarchical pattern can be separated

<table>
<thead>
<tr>
<th>Depth maps</th>
<th>(a)</th>
<th>(b)</th>
<th>(c)</th>
<th>(d)</th>
<th>(e)</th>
<th>(f)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left view</td>
<td>28.0820</td>
<td>31.0097</td>
<td>20.8105</td>
<td>30.5400</td>
<td>30.0092</td>
<td>33.0351</td>
</tr>
<tr>
<td>Right view</td>
<td>28.4416</td>
<td>30.9140</td>
<td>21.5653</td>
<td>29.7832</td>
<td>30.1531</td>
<td>32.9125</td>
</tr>
</tbody>
</table>

Table 2. BPR Results of Different Depth Maps in Fig. 17

<table>
<thead>
<tr>
<th>Depth maps</th>
<th>(a)</th>
<th>(b)</th>
<th>(c)</th>
<th>(d)</th>
<th>(e)</th>
<th>(f)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left view</td>
<td>0.1100</td>
<td>0.1088</td>
<td>0.1295</td>
<td>0.1093</td>
<td>0.1091</td>
<td>0.1084</td>
</tr>
<tr>
<td>Right view</td>
<td>0.1253</td>
<td>0.1246</td>
<td>0.1413</td>
<td>0.1259</td>
<td>0.1248</td>
<td>0.1244</td>
</tr>
</tbody>
</table>

Table 3. PSNR Results of Different Depth Maps in Fig. 18

<table>
<thead>
<tr>
<th>Depth maps</th>
<th>(a)</th>
<th>(b)</th>
<th>(c)</th>
<th>(d)</th>
<th>(e)</th>
<th>(f)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left view</td>
<td>25.7689</td>
<td>27.4489</td>
<td>20.0816</td>
<td>26.9382</td>
<td>27.2474</td>
<td>29.2093</td>
</tr>
<tr>
<td>Right view</td>
<td>25.8021</td>
<td>27.6526</td>
<td>20.2141</td>
<td>27.3488</td>
<td>27.5787</td>
<td>29.5938</td>
</tr>
</tbody>
</table>

Table 4. BPR Results of Different Depth Maps in Fig. 18

<table>
<thead>
<tr>
<th>Depth maps</th>
<th>(a)</th>
<th>(b)</th>
<th>(c)</th>
<th>(d)</th>
<th>(e)</th>
<th>(f)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left view</td>
<td>0.1486</td>
<td>0.1398</td>
<td>0.1549</td>
<td>0.1484</td>
<td>0.1433</td>
<td>0.1431</td>
</tr>
<tr>
<td>Right view</td>
<td>0.1547</td>
<td>0.1350</td>
<td>0.1562</td>
<td>0.1541</td>
<td>0.1325</td>
<td>0.1321</td>
</tr>
</tbody>
</table>
and decoded independently to obtain a corresponding depth map.

6. Conclusion
The MSL system for complete 3D reconstruction of a dynamic object has been a challenge due to the interference problem when patterns superpose each other. By analyzing the measurement principle of the SL technique, we proposed a method to design hierarchical patterns by combining the de Bruijn sequence and pseudorandom array. The proposed patterns can be separated by identifying different distribution of colors in each pseudorandom array. Then depth maps can be obtained by decoding separated patterns. Experimental results also demonstrate the availability and the robustness of the proposed patterns.

This work was supported by Natural Science Foundation of China (NSFC) (Grant Nos. 61170194, 61231010, and 61202301). The 3D models Lucy, Happy Buddha, and Bunny are courtesy of the Stanford Computer Graphics Laboratory.

References