Proceeding of the IEEE
International Conference on Robotics and Biomimetics
Dali, China, December 2019

A Structured Light Encoding Method for M-array Technique

1st Zhenzhen Liang
AI Institute, Ecovacs Robotics
Nanjing, China
zhen.liang@ecovacs.com

2nd Yuanlong Yu
AI Institute, Ecovacs Robotics
Nanjing, China
jeff.yu@ecovacs.com

3rd Hongxiang Xue
AI Institute, Ecovacs Robotics
Nanjing, China
hongxiang_xue@126.com

***Abstract*—In this paper, we have designed a structured light encoding algorithm based on the fill-and-test-as-you-go-along method. The algorithm achieves significant improvement for the structured light encoding technology, which could encode a lager resolution pattern with a certain number of symbols for encoding. It is significant to improve the accuracy of 3D recognition using the structured light (SL) technology, and reduce the complexity of the structured light algorithm, which is wildly used in 3D reconstruction, tactile sensor based on vision, grasping and so on. In the comparative experiment, we obtain nearly twice the size of the pattern encoded by the proposed encoding algorithm.**

***Index Terms*—structured light; pattern encoding; three-dimensional reconstruction; tactile sensor**

I. Introduction

Three-dimensional (3D) surface reconstruction is one of the most important research topic in computer vision due to its wild application in various fields, such as face recognition, industrial inspection, autonomous vehicles, and virtual reality [1] [2], etc. And structured light (SL) technology is considered to be one of the most reliable technologies for reconstructing the surface of an object because of its numerous advantages in terms of untouched, accuracy, speed, resolution, flexibility [3] [4]. A structured light system is similar to a stereo system with the difference of replacing one camera with a projector. The projected structured patterns carry encoded information to resolve the fundamentally difficult correspondence problem of the stereo vision technique.

Structured light techniques can be divided, according to different methods of encoding the projected pattern, into three categories [5]: direct coding, time-varying, and spatial neighborhood. The methods based on the time-varying strategy are easy to be implemented and can achieve the highest accuracy and a good resolution [6]. The major drawback to these patterns is the difficulty to solve dynamic scenes because several patterns must be projected precisely. Patterns coded directly offer a great spatial resolution but are unstable to noise and light variations. The spatial neighborhood codification strategies are attractive for low-cost systems, because the 3D reconstruction could be obtained from just a single pattern carried all the codes [2] [7]. More importantly, the last method can deal with dynamic scenes.

The methods based on the spatial neighborhood strategy normally have a single pattern with the entire coding information. The codewords of the structured light patterns usually uniqueness to alleviate any ambiguity while identifying the components. The sizes of the pattern for structured light technique, like the pixes for digital cameras, will determine the resolution of the 3D reconstruction. M-array patterns use the window property to define codewords based on De-Bruijn. By definition, an M-array (perfect map) is an $r \times v$ array over an alphabet of k symbols such that every possible $m \times n$ window appears exactly once [1]. Due to uncertain occlusions, shadows, and surface reflections, some pattern symbols cannot be identified from the captured image, which will lead to holes in the 3D shape reconstruction eventually. To solve this problem, some researchers elaborate patterns with a certain hamming-distance between the codewords [7] [8] [9]. Ref. [8] provided a good solution for the correspondence problem by setting the hamming-distance greater than or equal to 3. However, as introduced in [8] and in [9], used 7 and 8 symbols respectively, most of the related methods used too much symbols to achieve the condition of hamming-distance. The more symbols used in the encoding stage, the more difficult it will get in the decoding stage [9]. Therefore, it will be significant to consider an alphabet with a minimum number of symbols but could achieve lager-size pattern encoding correctly.

In this paper, an optimization algorithm is designed to solve above-mentioned problem, as an improvement of the fill-and-test-as-you-go-along method [8], which aim to produce larger size patterns with a certain number of symbols in the alphabet. The fill-and-test-as-you-go-along method is wildly used to encode an M-array pattern [1] [2]. By the proposed method, a significant improvement is achieved for structured light encoding.

II. Methodology

A. A Pyramid of Pattern Encoding Algorithm

Fig.1 shows an example of encoding a pattern with a size of 6×8 using the fill-and-test-as-you-go-along method. To generate a pattern with a (3×3) window property, the encoding algorithm starts from the top-left (3×3) window with random numbers taken from an alphabet P ($P = \{0, 1, 2\}$). Then (3×1) columns with random numbers are added to the right of the initial window consecutively. Next, (1×3) rows are added beneath the initial window similarly. Then the processes are

978-1-7281-6321-5/19//$31.00 ©2019 IEEE

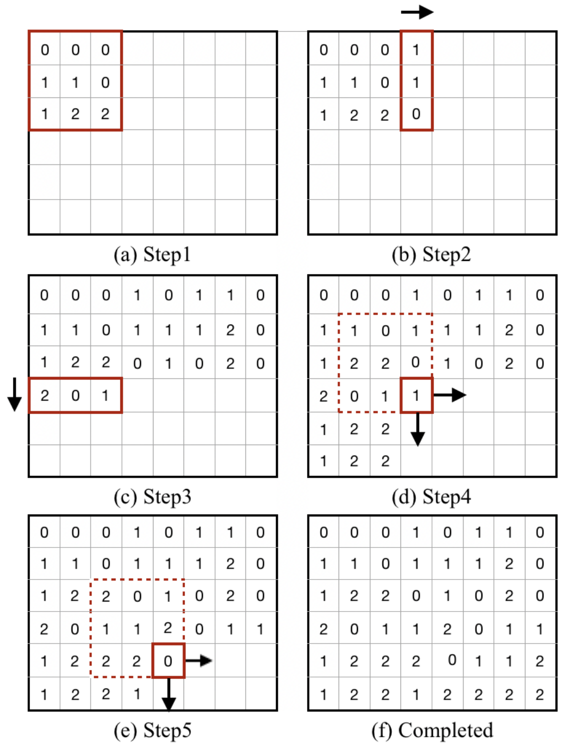


Fig. 1. The process of encoding a 6×8 pattern with (3×3) window property. (a) Step1: Filling the first (3×3) window with random numbers assignment. (b) Step2: Appending (3×1) columns with random elements to the right of the initial window. (c) Step3: Appending (1×3) rows after the first three rows is filled completely. (d) Step4: Appending a single element towards the right and down. (e) Step5: Filling of next row and col.

repeated by adding one number once until the whole pattern is filled. During the above process, the uniqueness of the codeword (window property) will be checked when new (3×3) window appears. The encoding process is deemed to be failed if all the three numbers in the alphabet P cannot satisfy the window property and the process will start over again.

However, there is a problem with the previous encoding process. We find that it is too early to stop the encoding algorithm and start all over again once it encounters that no numbers in the alphabet could fill the pattern. As shown in Fig.2(a), for example, the red box element is filled by one. Then the pattern is filled element by element. Now we assume no number in the alphabet P can satisfy the window property at the place marked by green box. The process will start over from the top-left of the whole pattern but don't try zero (as shown in Fig. 2(b)) or two at the place marked by red box in Fig.2, which will kill many possibilities of encoding process.

An optimization method is designed to solve the above-mentioned problem base on the fill-and-test-as-you-go-along method algorithm. Instead of encoding the pattern from beginning to end, the pattern is divided into several parts that will be encoded respectively. Take 42×42 for example, the pattern is divided into six parts as shown in Algorithm 1. The initial process (gray color area shown in Fig. 3) of

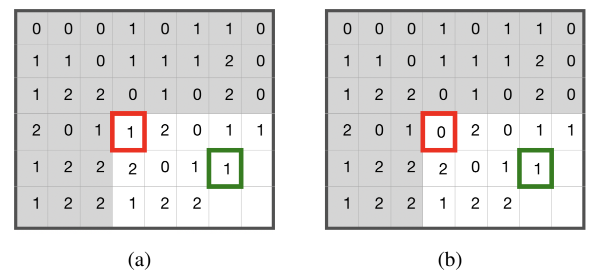


Fig. 2. The problem of traditional encoding algorithm.

pattern accomplished in the external loop, including encoding of first 3×3 window, first three rows and first three cols. The remainder of the pattern is evenly divided into five parts according to the number of rows and cols, corresponding to different steps in Fig. 3. The elements of the blue part (Fig. 3) are filled in loop1. But unlike the fill-and-test-as-you-go-along method algorithm, the encoding process will restart from the beginning of the blue area instead of the top-left place of the whole pattern. Process from loop2 to loop5 is similar to loop1. Then the encoding process of the whole pattern is divided into some different little pattern and encode them individually. When the algorithm encounters none numbers of the alphabet could be correctly filled, the process will first do inner loop encoding certain times, as shown in Algorithm 1. The process will jump to the upper loop until the whole pattern is filled if the current area is not still filled correctly when the current loop completes as shown in Algorithm 1.

B. Pattern Encoding

To verify the performance of our optimization algorithm, we make a comparison between pattern encoded by Udaya et al. [1] and our pattern encoded after being optimized.

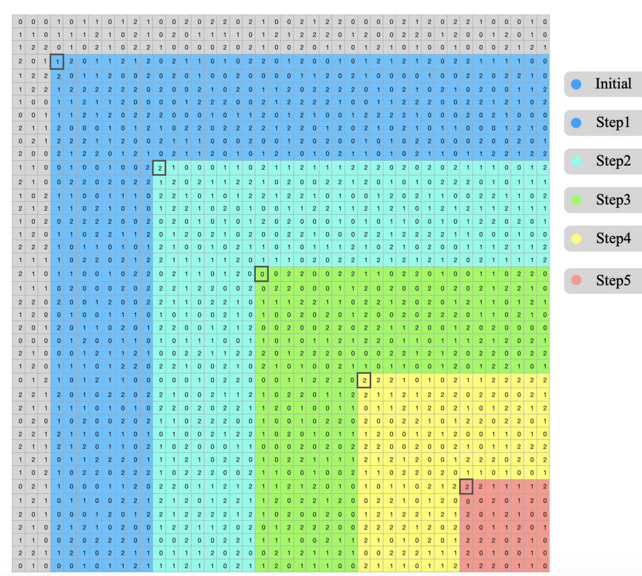


Fig. 3. The process of proposed encoding algorithm.

Algorithm 1 A pyramid of pattern encoding algorithm

However, there is a problem with the previous encoding process. We find that it is too early to stop the encoding algorithm and However, there is a problem with the previous encoding process. We find that it is too early to stop the encoding algorithm and

Input: pattern size
Output: pattern

```
while True do
    Initial
    while True do
        Fill the blue area
        if Failed to fill blue area then
            loop1 ← loop1 + 1
            if loop1 < pow(2, 9) then break
            else continue
        else
            Fill the cyan area
            if Failed to fill cyan area then
                loop2 ← loop2 + 1
                if loop2 < pow(2, 8) then break
                else continue
            else
                Fill the green area
                if Failed to fill green area then
                    loop3 ← loop3 + 1
                    if loop2 < pow(2, 7) then break
                    else continue
                else
                    Fill the yellow area
                    if Failed to fill yellow area then
                        loop4 ← loop4 + 1
                        if loop2 < pow(2, 6) then break
                        else continue
                    else
                        Fill the red area
                        if Failed to fill red area then
                            loop5 ← loop5 + 1
                            if loop5 < pow(2, 5) then
                                break
                            else continue
                            if Succed to fill the whole pattern
    then
                                return pattern
```

Udaya uses a dual pseudorandom array encoding method, which encodes a binary pattern with a (7×7) window property and then encodes a color pattern with (3×3) window property. And he gives (80×60) as the largest possible resolution with (7×7) window property but (60×45) is the highest possible resolution for pattern[(use of six colors)] with (3×3) window property. With the same constraints, we obtain greater than (100×100) (just show 100×100 in this paper) for (7×7) window property and (100×100) for (7×7) window property, as shown in Fig. 4(a) and Fig. 4(b) respectively.

Table I shows the comparable results between the traditional method and our method. A significant improvement can be seen with our proposed encoding method. Then we choose a monochromatic pattern to reconstruct the 3D shape of the object with a self-designed tactile sensor, which will be introduced in the following section.

C. The Pattern and Tactile Sensor Used in This Papaer

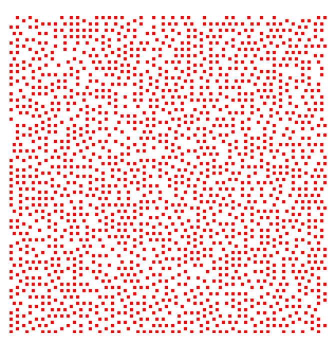
Albitar generates a monochromatic pattern with a (3×3) window property using three symbols. He gives (27×29) as the largest possible resolution when the hamming distance of the codewords is equal to 3. Then, with the same constraints we obtain a similar pattern with (42×42) resolution as shown in Fig. 5. Here, to reduce the difficulty of decoding, we adopt three simple geometrical shapes: line, circle, and triangle to associate with three symbols in the alphabet, as shown in Fig. 6.

The hardware device used in this paper is a tactile sensor designed by ourself. The tactile sensor used in this paper combines the working characteristics of optical tactile sensors and structured light systems, as shown in Fig. 7.

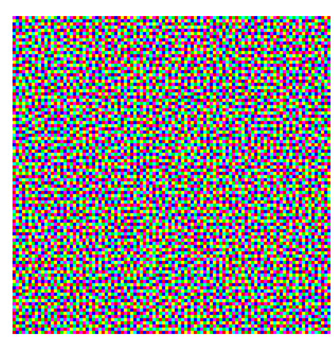
We prepared the projection pattern in the structured light system as an adhesion layer pattern on the surface of the elastomer in the optical tactile sensor. According to the spatial relationship between the two cameras and the transparent elastic body, the three-dimensional structure of the contact object is reconstructed based on the pattern information of the adhesion layer. It is worth noting that the projection pattern can only use the pattern encoded by the spatial encoding method due to device limitations.

D. Segmentation Decoding

The detection of the correspondences between two images captured from stereo cameras is one of the most important steps in 3-D reconstruction. In the pattern decoding precess, the codeword of each pattern symbol in both captured images should be accurately determined to solve the correspondences. In this section, a linked-symbols segmentation algorithm is introduced and CNNs algorithm is applied to classify symbols



(a)



(b)

Fig. 4. The pattern encoded by proposed algorithm with the same constraints with the pattern introduced by Udaya [1].

TABLE I
THE RESOLUTION COMPARISON BETWEEN TRADITIONAL ENDOING ALGORITHM AND THE PROPOSED ALGORITHM.

Encoding Method	Number of symbols	Pattern resolution	Codeword size
Chen et al.[16]	7	38×212	4 neighbors
Morano et al.[17]	3	45×45	3×3 window
Pages et al. [18]	3	20×20	3×3 window
Albitar et al.[19]	3	27×29	3×3 window
Udaya et al.	6	60×45	3×3 window and 7×7 window
Optimized method	6	100×100	3×3 window and 7×7 window

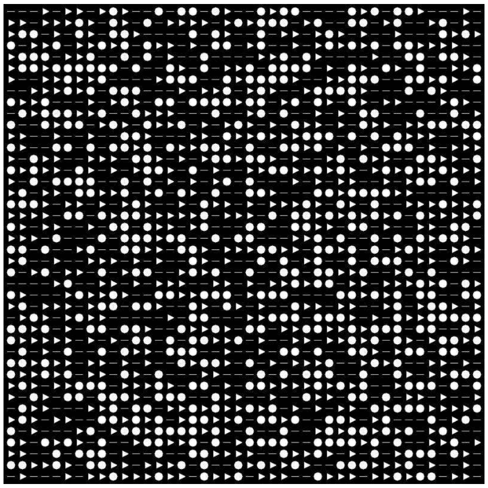
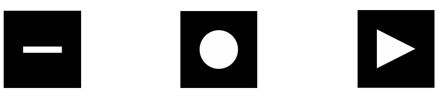


Fig. 5. The pattern used in this paper.



(a) (b) (c)

Fig. 6. Primitives used in pattern. (a) line; (b) circle; (c) triangle

to label each contour by one of the three pattern primitives: line, triangle, and circle.

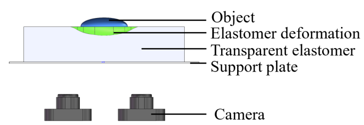
In this stage, all the center grid points that represent the centers of the contours are extracted from the captured image. First, the captured images are converted to the gray image. Then gray image is transformed into a binary image according to the formulae:

$$dst(x,y) = \begin{cases} maxValue & \text{if } src(x,y) > T(x,y) \\ 0 & \text{otherwise} \end{cases}. \quad (1)$$

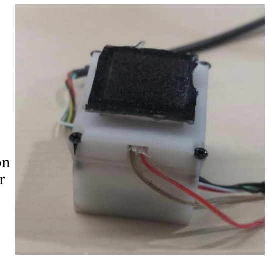
Here $T(x,y)$ is an adaptive threshold calculated individually each pixel. Next, Morphological transformations are applied to remove noise. We use erosion filter defined in (2) to shrink all the segments in the binary image and dilation filter defined in (3) to expand the segments smoothly.

$$dst(x,y) = \min_{(x',y'):element(x',y')\neq 0} src(x+x', y+y') \quad (2)$$

$$dst(x,y) = \max_{(x',y'):element(x',y')\neq 0} src(x+x', y+y') \quad (3)$$



(a)



(b)

Fig. 7. The principle and the device of the tactile sensor based on structured light technology.

Since every symbol is different from its eight neighbors, a contour detection algorithm is applied separately to each binary image to locate the shape symbols in the captured image. The size of each contour is computed and compared with the average size of all contours. Very large or small contours that are not related to the pattern symbol will be removed according to the comparison. Due to surface deformation of the target objects, uneven illumination, two or three contours detected from the captured image may be connected, as in the example shown in Fig.8. Then a method is designed based on image moments to obtain both centers of linked symbols. Fig.9 show a sketch of two linked symbols. Where the red dotted arrow stands for the bounding box of linked-symbols, the pale blue area stands for the gap between two symbols, and the dark blue area stands for symbols. Then the centroids of both symbols can be calculated by:

$$\begin{cases} x_1 = \bar{x} - 3/10 * d_c \, sin(\theta/180 * \pi) \\ y_1 = \bar{y} - 3/10 * d_c \, cos(\theta/180 * \pi) \\ x_2 = \bar{x} + 3/10 * d_c \, sin(\theta/180 * \pi) \\ y_2 = \bar{y} + 3/10 * d_c \, cos(\theta/180 * \pi) \end{cases}. \quad (4)$$

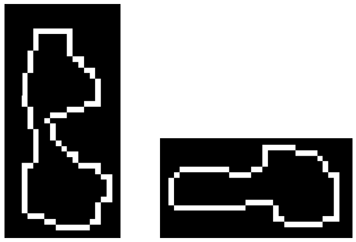
Where, $(\bar{x}, \bar{y})$ is the centroid point of the linked symbols and calculated by:

$$\bar{x} = \frac{m_{1,0}}{m_{0,0}}, \quad \bar{y} = \frac{m_{0,1}}{m_{0,0}}. \quad (5)$$

Then the angle between the long side of bounding box d_w and x axis can be calculated by:

$$\theta = \frac{1}{2} tanh^{-1}\left(\frac{2m_{11}}{m_{20} - m_{02}}\right) + N\frac{\pi}{2} \quad (6)$$

The size of the symbols d_s is obtained by calculating the



(a) (b)

Fig. 8. Linked contours of symbols

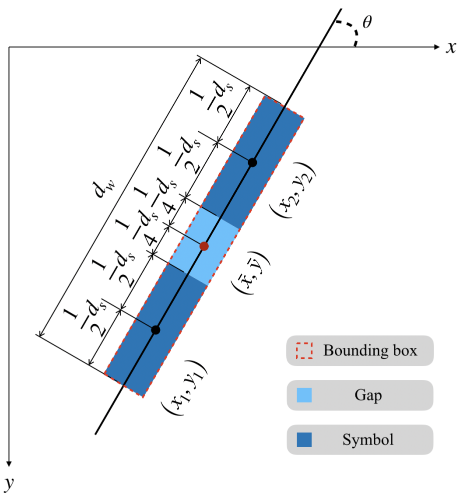


Fig. 9. The principle to segment linked symbols.

mean distance between all symbols of the pattern. The image moments $m_{i,j}$ is calculated by:

$$m_{i,j} = \sum_{x,y} x^i y^i f(x, y) \tag{7}$$

Fig.11 shows the segment process of one captured image, where, Fig. 11(a) is the captured image by the camera of tactile sensor, Fig. 11(b) is the contour detection result of captured image, Fig. 11(c) shows the centroid of each contours shown in Fig. 11(b), then Fig. 11(d) shows the Optimized result by proposed method. By comparing the grid points in Fig. 11(b) and Fig. 11(d), we can see that the holes is well complemented.

E. Classifying the detected symbols

After getting the grid points of each symbol, we need to determine the classification of each grid point representing. However, due to surface textures, surface discontinuities of the target objects or uneven illumination, the geometrical shapes in the acquired images are usually blurred and distorted as shown in Fig.11(a). Therefore, it is difficult to identify the shapes of elements of the pattern through traditional image matching methods. Then we regard the pattern elements identification as a multi-classification problem, and the latter is wildly researched in the domain of computer vision and achieved outstanding success by using convolutional neural networks

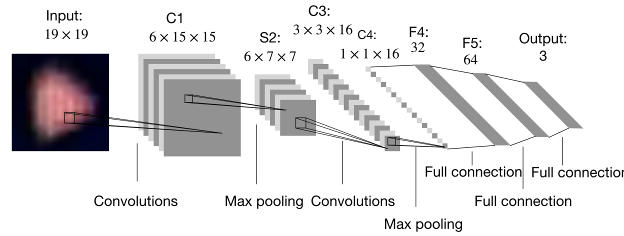


Fig. 10. The principle to segment linked symbols.

(CNNs). For this paper, we need to classify the following three types: line, circle, and triangle.

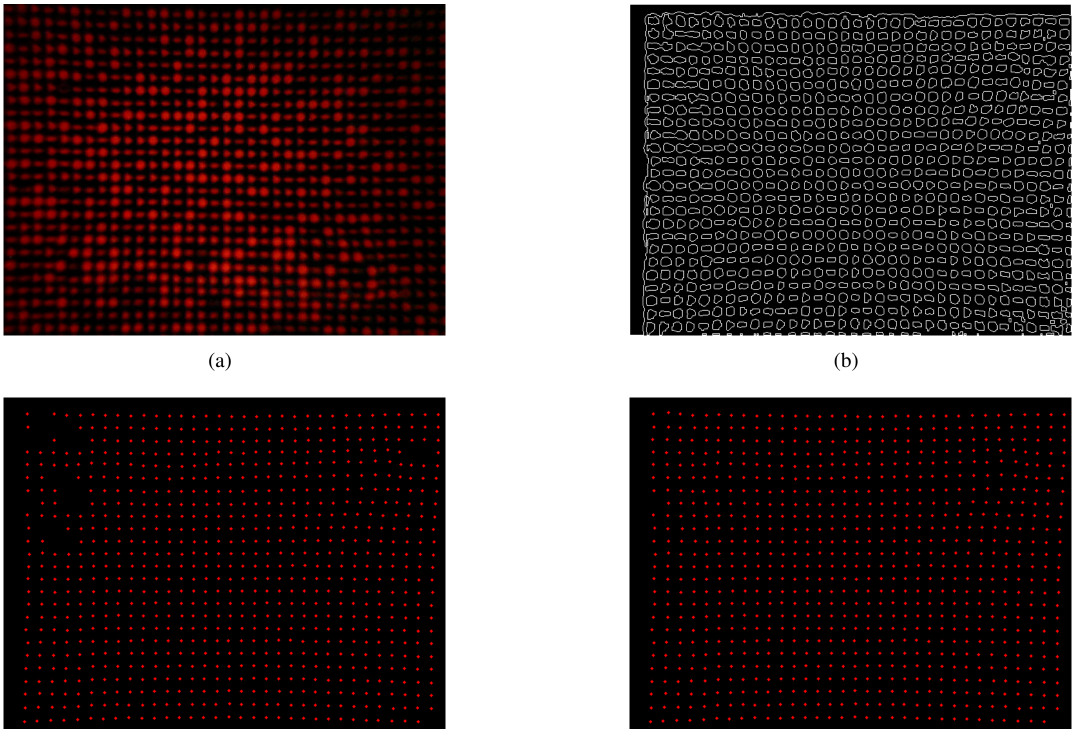
We design a convolutional neural networks based on the Lenet-5 [10] which is well-known for its efficient solution for handwritten digit recognition problem. Fig. 10 shows the CNNs structure of this paper, which includes seven layers: three conventional layers, two max-pooling layers, one fully connected layers, and the output layer. Considering the average distance of pattern elements, the input image size in this CNNs is set to (18×18). The first convolution layer $C1$ uses 6 filter kernels with a size of (5×5) and stride of 1. The second convolution uses 16 filter kernels with a size of (5×5) and stride of 1. Two max-pooling layers F4, F5 is applied after each convolution layer is done. Then two connection layers applied with nodes 32 and 64 respectively.

To improve the performance of the symbols classifier, it is necessary to collect pattern element samples with different objects, distortion, etc. For this paper, we select spoon model, Rome pillar model, bowling model, human finger as objects to collect samples. Different pressure is applied when collecting samples with above objects. 6000 samples are collected totally where 5200 symbols for training and 800 for testing.

III. Experiment Results And Discussion

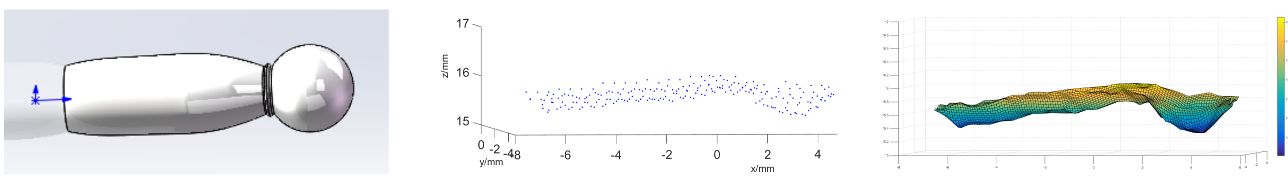
In this paper, we designed a new M-array structured light encoding algorithm, which could encode a larger resolution pattern with a certain number of symbols. Fig. 4 shows the patterns encoded by proposed coding algorithm. Fig. 4(a) and Fig. Under the same constraints with Udaya et al. [1] proposed, we obtain (100×100) a binary pattern (Fig. 4(a)) and a (100×100) color pattern (Fig. 4(b)). But Udaya gives (80×60) as the largest possible resolution for binary pattern and (80×60) resolution for color pattern, which is also discussed in Section II. Then in order to verify the applicability of the algorithm, we performed a 3D reconstruction experiment using a tactile sensor designed by ourself, as shown in Fig. 7.

In the experiment, the tactile system is composed of two wild filed cameras with resolution of 640×480 pixels, and focus of $3.35mm$, which are calibrated by the method proposed by Zhang [11]. The distance of two symbols in the adhesion layer is $0.5mm$. Working distance is about $15mm$ and the measurement area is about $5mm \times 15mm$. To test the 3D reconstruction capability of the designed tactile sensor, we use the roman column model (Fig. 12(a)) as the testing object,



(a) (b) (c) (d)

Fig. 11. Square grid points extraction process: (a) captured image; (b) contour detection result; (c) extracted the grid points with linked-symbols; (d) after segment the linked-symbols



(a) (b) (c)

Fig. 12. Model and the 3D reconstruction result: (a) Bowling model; (b) The reconstructed 3D point cloud model; (c) The reconstructed 3D model of object.

and the 3D reconstruct result is shown in Fig. 12(b) and Fig. 12(c).

IV. Conclusions

We have designed a new tactile sensor that combines the advantages of a structured light system and optical tactile sensor. It can sense the 3D structure of the contact objects. Besides, embedding the structured light technology the designed tactile sensor could solve many problems the traditional structured light technique encountered, such as the sensitivity to color, immovability of the structured light device, etc.

As future work, we are going to improve the performance of the tactile sensor. The tactile sensor in this paper is the first-generation device. We are planning to design the next device to enhance the 3D reconstruction stability of the sensor, and enable the sensor to sense the stiffness of the objects.

References

[1] Wijenayake, Udaya, and Soon-Yong Park. "Dual pseudorandom array technique for error correction and hole filling of color structured-light three-dimensional scanning." Optical Engineering 54.4 (2015): 043109.

[2] Lei, Yang, et al. "Design and decoding of an M-array pattern for low-cost structured light 3D reconstruction systems." 2013 IEEE International Conference on Image Processing. IEEE, 2013.

[3] Salvi, Joaquim, Jordi Pages, and Joan Batlle. "Pattern codification strategies in structured light systems." Pattern recognition 37.4 (2004): 827-849.

[4] Li, Beiwen, et al. "High-accuracy, high-speed 3D structured light imaging techniques and potential applications to intelligent robotics." International Journal of Intelligent Robotics and Applications 1.1 (2017): 86-103.

[5] J. Salvi, J. Pages and J. Battle, "Pattern codification strategies in structured light systems", Pattern Recognition, Vol. 37, pp. 827-849, 2004.

[6] J. Salvi, S. Fernandez, T. Pribanic, and X. Llado, A state of the art in structured light patterns for surface profilometry, Pattern Recognition, vol. 43, no. 8, pp. 26662680, August 2010.

[7] Albitar, Chadi, Pierre Graebling, and Christophe Doignon. "Robust

structured light coding for 3D reconstruction." 2007 IEEE 11th International Conference on Computer Vision. IEEE, 2007.

[8] R. Morano, C. Ozturk, R. Conn, S. Dubin, S. Zietz, and J. Nissanov, "Structured Light Using Pseudo-random Codes", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.20 (3), pp. 322 327, 1998.

[9] A. Adan, F. Molina, and L. Morena, "Disordered patterns projection for 3D motion recovering", In Proceedings of the 2nd Int. Symp. on 3D Data Processing, Visualization, and Transmission, 3DPVT'04, pp. 262-269, 2004.

[10] LeCun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.

[11] Zhang, Z. "Flexible camera calibration by viewing a plane from unknown orientations." Proceedings of the Seventh IEEE International Conference on Computer Vision IEEE, 1999.