MVA'94 IAPR Workshop on Machine Vision Applications Dec. 13-15, 1994, Kawasaki

Mesh Pattern Recognition Using Correlation Matching Method

Chin-Chuan Han **Kuo-Chin Fan** *

Institute of Computer Science and Information Engineering
National Central University, Chung-Li, Taiwan, R.O.C.

ABSTRACT

In this paper, a novel method is proposed to recognize the type of mesh pattern synthesized in a binary text image. Since the pixels of mesh pattern are regularly arranged, it is intuitive to identify the type of mesh pattern by analyzing the regularity of black pixels in the considered image. To achieve the recognition goal, the coefficient of correlation between the input image and the mesh pattern stored in the database is calculated to determine the type of mesh pattern presented in the input image. Experimental results are conducted to verify the validity of our proposed method.

1 INTRODUCTION

Text characters synthesized with mesh patterns appear in many disk top publications frequently. For instance, the title together with certain mesh pattern emphasizes the importance of an article to attract the interest of readers. Computer system designers utilize the mesh patterns in flowchart design to distinguish one module from other modules. However, if the document is synthesized with mesh patterns, the performance of available optical character recognition system will deteriorate drastically. Hence, removing the mesh pattern from the synthesized text characters before the character recognition process is prerequisite to resolve the recognition problem. To achieve this, the identity of mesh pattern in the synthesized images is firstly identified. And then a specified filter is designed to filter out the mesh pattern. In this paper, we will propose a novel method to identify the identity of mesh pattern in the synthesized images.

A mesh pattern can be considered as a region with regular textures. Texture is an important characteristic feature in the image analysis. During the past years, three major approaches, *statistical, spectral,* and *structural*[1 – 5], have been developed to extract the texture information. In the statistical approach, the image texture is considered as finding the useful features such as the fineness, coarseness, contrast, directionality, roughness, and regularity of images. In practice, the co-occurrence matrix approaches are widely used to extract the texture information of images. However, the texture information is characterized by a set of multi-dimensional features in these methods. Spectral techniques[6, 7] are based on the Fourier spectrum and are primarily used to detect global periodicity in an image by identifying the high-energy or narrow peaks in the spectrum. As to structural techniques[8, 9], they deal with the arrangement of image primitives, such as the description of texture based on regularity space.

Recently, the texture spectrum method[4, 5] has been proposed as a statistical approach to analyze the texture signals. The *texture unit* is designed to characterize the local texture for a given pixel and its neighborhood. The texture information can be characterized as the texture spectrum which is the occurrence frequency function of all texture units within the image. However, all of them are the gray-scale images application.

Carefully analyze the binary mesh pattern images, we perceive that the black pixels(value 1) of mesh pattern are arranged regularly and the mesh pattern can be considered as a region with regular texture. A novel method is developed to recognize the mesh pattern type presents in the considered text image by analyzing the regularity of black pixels.

2 THE APPROACH

The methodology of our proposed method is composed of five main modules: (1) noise removal; (2) generation of pixel arrangement histogram; (3) data filtering and normalization of statistical data; (4) measurement between two statistical data sets; (5) correlation matching.

Before describing the five modules, an encoded function is firstly defined to facilitate the later process.

Consider a pixel P and its eight neighboring pixels $P_0, P_1, \ldots, P_7$ as shown in Figure 1(a), the arrangement of these eight pixels can be encoded by

$$\begin{aligned}\varphi_P &= P_7 \times 2^7 + P_6 \times 2^6 + P_5 \times 2^5 + P_4 \times 2^4 \\ &\quad + P_3 \times 2^3 + P_2 \times 2^2 + P_1 \times 2^1 + P_0, \qquad (1)\end{aligned}$$

and a byte value φ_P $(0 \sim 255)$ as shown in Figure 1(b) is accordingly obtained. Note that for conciseness sake an encoded value φ, for example $\varphi = 63$, is represented in hexadecimal form such as $3F_{16}$ in the rest of the paper.

Module 1 : Noise removal

Noise removal module is a preprocess procedure to remove the unwanted noises, improve the quality of input

P_7	P_6	P_5
P_4	P	P_3
P_2	P_1	P_0

(a)

P_7	P_6	P_5	P_4	P_3	P_2	P_1	P_0

(b)

Figure 1: (a) A pixel P and its eight neighboring pixels, (b) the encoded byte of these eight neighbors.

image and enhance the performance of later recognition task.

In the processed image, if a pixel P in a 3×3 mask satisfies

1. $P = 1$;
2. $\varphi_P = 07_{16}$ or $\varphi_P = \mathsf{E0}_{16}$ or $\varphi_P = 29_{16}$ or $\varphi_P = 94_{16}$,

the pixel P will be treated as a black noise and removed. Besides, if the pixel P satisfies the following criteria :

1. $P = 0$;
2. $\varphi_P = \mathsf{1F}_{16}$ or $\varphi_P = \mathsf{F8}_{16}$ or $\varphi_P = \mathsf{6B}_{16}$ or $\varphi_P = \mathsf{D6}_{16}$,

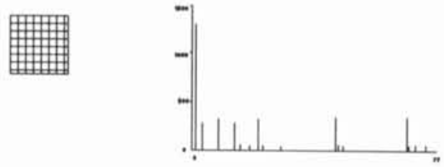
the pixel P will be considered as a white noise and will thereby be set as a black pixel. In the noise removal module, the black and white noises presented in an image will be processed accordingly following the stated principles.

Module 2 : Generation of pixel arrangement histogram

The main purpose of this module is to reflect the geometric arrangement of a pixel by encoding its corresponding neighborhoods. All pixels in the mesh pattern image are encoded by Equation (1). Then count the number of pixels for a particular arrangement code φ, denoted $\Gamma[\varphi]$, to generate the arrangement histogram for each possible arrangement.

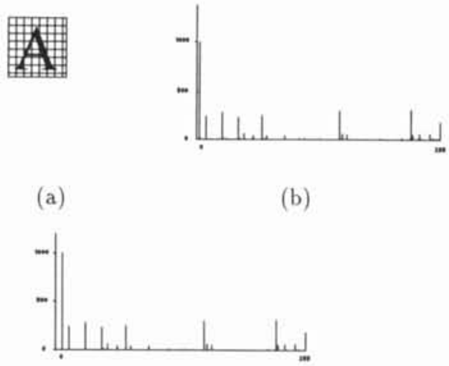
To best illustrate the generation of arrangement histogram, let us consider the image consisting of horizontal and vertical lines as shown in Figure 2(a). After encoding and counting the number of pixels for all arrangements, an arrangement histogram as shown in Figure 2(b) is thus obtained. Since the pixels of mesh pattern is regularly arranged, some arrangements of pixels, such as $42_{16}, 18_{16}, 94_{16}$, etc., as shown in Figure 2(c) occur frequently.

An example of synthetic binary image as shown in Figure 3(a) is generated by adding a text character 'A' into the mesh pattern (e.g., Figure 2(a)). The arrangement histogram as shown in Figure 3(b) for this synthetic image can be obtained accordingly by making use of the above rules. Making comparisons between the



(a) (b)

Figure 2: (a) Mesh pattern image with horizontal and vertical lines, (b) the arrangement histogram of (a).



(a) (b)

(c)

Figure 3: (a) An image with character 'A' and 'grid' mesh pattern, (b) the arrangement histogram of (a) before filtering, (c) the arrangement histogram of (a) after filtering.

two arrangement histograms of Figures 2(b) and 3(b), the pixels whose arrangement code is changed due to the appearance of boundary pixels are very rare. That is; the arrangement histogram would not be changed too much even though the text characters are added into the mesh pattern images.

Furthermore, Figure 4 demonstrates the examples of three different mesh patterns without any text character with their corresponding histograms being plotted at the right-hand side. These histograms are stored in the mesh pattern database for later correlation matching module. From Figure 4, an important verdict is observed: *Different mesh patterns will generate different arrangement histograms.* This verdict is the foundation for use in the recognition process. Based on these training data stored in the database, the type of mesh pattern embedded in an input text image can be recognized.

Module 3 : Data filtering and normalization of statistical data

After adding the text characters, the number of pixels whose arrangement code is changed due to the appearance of boundary pixels is relatively rare comparing to those of the pixels without change. A simple filter rule is designed as

$$\Gamma[\varphi] = \left\{ \begin{array}{ll} \Gamma[\varphi] & \text{if } \Gamma[\varphi] \geq \overline{\Gamma}; \\ 0 & \text{otherwise;} \end{array} \right\} \text{for } \varphi = 0, 1, \ldots, 255 \quad (2)$$

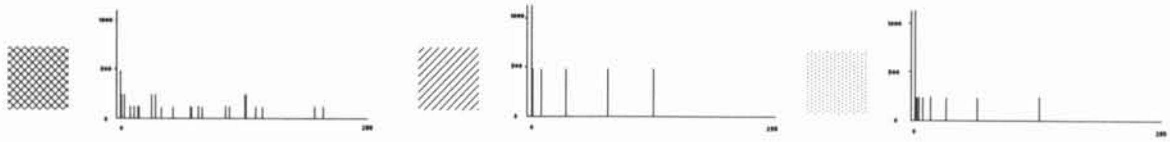


Figure 4: Three arrangement histograms of three different mesh patterns.

where :

$\Gamma[\varphi]$ is the number of pixels occurring in an image with arrangement code φ, and the value $\overline{\Gamma}$ is the average number which equals $\frac{1}{256}\sum_{\varphi=0}^{255}\Gamma[\varphi]$.

If the number for a specific arrangement code is smaller than the average number, the number for this code is set to zero. Because this arrangement code may be considered as the code generated via the boundary pixels of added text characters. Otherwise, if the number for an arrangement code is larger than the average number, the number for this arrangement code remains unchanged.

According to the expression stated in Equation (2), a new arrangement histogram as depicted in Figure 3(c) is thus generated from Figure 3(b).

Since the sizes of input text image and mesh pattern image stored in the database may be different, the normalization procedure should be performed to normalize the size of arrangement code in the arrangement histogram. The rules for normalization can be formulated as

$$x_{\varphi} = \frac{\Gamma[\varphi]}{\sum_{\varphi=0}^{255}\Gamma[\varphi]}, \qquad \text{for } \varphi = 0, 1, \ldots, 255. \qquad (3)$$

Module 4 : The measurement of two statistical data sets

In this module, an evaluation function, *correlation coefficient*[10], is devised to measure the similarity of input image and mesh pattern image stored in the database. The *coefficient of correlation* between the tested random variable Y and the variable Z in the database is denoted by $\rho\{Y, Z\}$ and defined as

$$\rho\{Y, Z\} = \frac{\sum_{\varphi=1}^{254}(y_{\varphi} - \overline{Y})(z_{\varphi} - \overline{Z})}{\sqrt{\sum_{\varphi=1}^{254}(y_{\varphi} - \overline{Y})^2 \sum_{\varphi=1}^{254}(z_{\varphi} - \overline{Z})^2}}, \qquad (4)$$

where:

$\overline{Y} = \frac{1}{254}\sum_{\varphi=1}^{254} y_{\varphi}$ and $\overline{Z} = \frac{1}{254}\sum_{\varphi=1}^{254} z_{\varphi}$ are the sample mean of variable Y and Z, respectively.

Four elements $y_{00_{16}}$, $y_{FF_{16}}$, $z_{00_{16}}$, and $z_{FF_{16}}$ in random variables Y and Z should be excluded from consideration because the size of these four elements is mostly affected by the body of added characters. Hence, the indexes of Equation (4) start from 1 and end at 254.

Module 5 : Correlation matching

In this module, we adopt the correlation coefficient as the matching criterion to perform the recognition job. The correlation matching principle is : *The higher the similarity between two statistical sets, the larger the correlation coefficient will be generated.* According to the matching principle, the identity of mesh pattern presented in an input text image can thereby be determined by choosing the one with the maximum correlation coefficient. For the given example, the correlation coefficient ρ_{max} of the two images as shown in Figure 2(a) and Figure 3(a) is 0.992 which is the maximum correlation coefficient value comparing with the correlation coefficients for the input image in Figure 3(a) and three mesh pattern images in Figure 4 which are -0.059, -0.029 and -0.037, respectively. Therefore, we can conclude that the input image is a text image mixed with the mesh pattern of horizontal and vertical lines.

3 EXPERIMENTAL RESULTS

In this section, some experimental results are illustrated to show the validity of our proposed method. Eight mesh patterns as shown in Figure 5 are inputted from the scanner to construct the database. Each mesh pattern is identified with an identity, such as type **A**, **B**, ..., **F**. The arrangement histograms for each mesh pattern in the database are also displayed at the right-hand side of each mesh pattern. In Figure 5, we only plot the arrangement codes from 1 to 254 in each histogram.

Two real images as shown in Figure 6 are inputted from a platform scanner. They are two examples of text characters with **F** type mesh pattern, and text characters with **B** type mesh pattern. The arrangement histograms for these two images are shown at the right-hand side of Figure 6. The correlation coefficients calculated for each of these two images vs the images in the database are tabulated in Table 1. In Figure 6(a), the type of mesh pattern with the maximum coefficient value equaling 0.95 is type **F**. Similarly, the maximum correlation coefficient of the image in Figure 6(b) is 0.98. According to the correlation matching rule, the mesh pattern presented in the input image of Figure 6(b) is identified as type **B**.

4 CONCLUSIONS

A binary mesh pattern recognition method for recognizing the type of mesh pattern presented in a text image is presented in this paper. The basic idea behind our approach is the analysis of the regularity embedded in the mesh pattern from which the correlation coefficient is developed. The identity of the mesh pattern presented in an input image can then be obtained by performing the correlation matching. The one with the maximum correlation coefficient is identified as the match. Experimenting with a wide variety of text images verify the validity of our proposed method.

Table 1: The correlation coefficients for the four images in Figure 3 vs each mesh pattern image in the database.

	(A)	(B)	(C)	(D)	(E)	(F)
Fig. 6(a)	0.07	0.76	0.60	0.40	0.72	0.95
Fig. 6(b)	-0.09	0.98	0.20	0.21	0.70	0.90

References

[1] R. Bajcsy and L. Lieberman, "Texture Gradient as a Depth Cube," *Comput. Graph. Image Proc.*, vol. 5, no. 1, pp. 52-67, 1976.

[2] G. R. Cross and A. K. Jain, "Markov Random Filed Texture Models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-5, no. 1, pp. 25-39, 1983.

[3] R. M. Haralick, "Statistical and Structural Approaches to Texture," *Proc. 4th Int. Joint. Conf. Pattern Recog.*, pp. 45-60, 1979.

[4] D. C. He, L. Wang, and J. Gilbert, "Texture features extraction," *Pattern Recognition Lett.*, vol. 6, pp. 269 - 273, 1987.

[5] D. C. He and L. Wang, "Texture features based on texture spectrum," *Pattern Recognition*, vol. 24, no. 5, pp. 391 - 399, 1991.

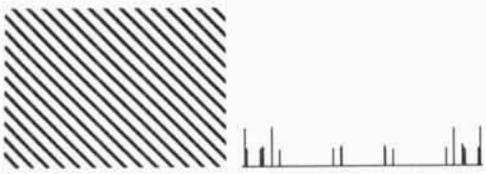
[6] R. Bajcsy, "Computer Description of Texture Surfaces," *Proc. 1973 Int. Conf. Artificial Intell.*, Stanford, Calif., pp. 572-579.

[7] D. Chen and L. Wang, "Texture Feature Based on Texture Spectrum," *Pattern Recog.*, vol. 24, no. 5, pp. 391-399, 1991.

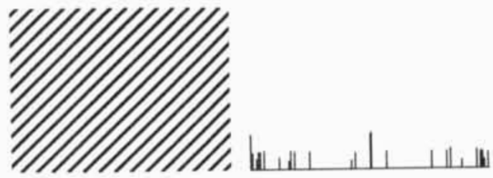
[8] S. Y. Lu and K. S. Fu, "A Syntactic Approach to Texture Analysis," *Comput. Graph. Image Proc.*, vol. 7, no. 3, pp. 303-330, 1978.

[9] F. Tomita, Y. Shirai and S. Tsuji, "Description of Texture by a Structural Analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-4, no. 2, pp. 183-191, 1982.

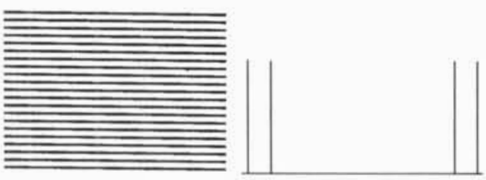
[10] R. V. Hogg and A. T. Craig, *Introduction to Mathematical Statistics*, Machillan, New York, 1986.



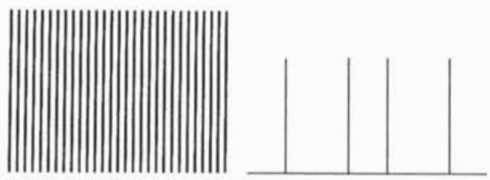
(a) Type A



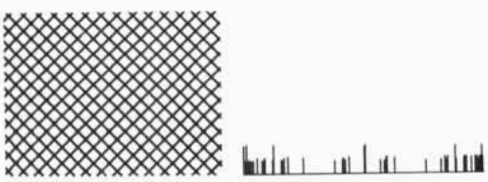
(b) Type B



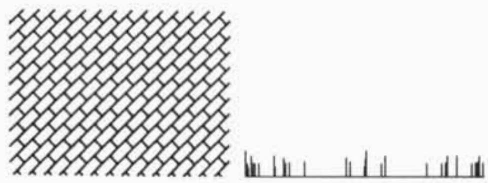
(c) Type C



(d) Type D



(e) Type E



(f) Type F

Figure 5: The mesh patterns and their corresponding arrangement histograms stored in database.

(a)

(b)

Figure 6: Three real images and their arrangement histograms.