Chinese Character Font Recognition using Fourier Spectrum Features and Back-propagation Neural Network✞

Hsien-Hung Tseng@ (曾憲宏) and Wen-Hsiang Tsai* (蔡文祥)

Department of Computer and Information Science
National Chiao Tung University
Hsinchu, Taiwan 300, Republic of China
E-mail:whtsai@cis.nctu.edu.tw
Tel:886-35-712121 ext. 56621, Fax:886-35-721489

Abstract

A simple and feasible approach to Chinese character font recognition using Fourier spectrum features and the back-propagation neural network is proposed. An input image with one Chinese character bitmap pattern is transformed into the frequency domain using the Fourier transform. The Fourier amplitude spectra of the transformed image is then divided into non-overlapping square blocks with non-equal sizes, and a spectrum feature vector is derived from the mean amplitudes of the square blocks. A back-propagation neural network classifier is used to recognize the font types of Chinese characters after learning. Experiments have been conducted in which training samples and test samples were selected randomly from 5,401 most frequently used Chinese characters. An average recognition rate of 86% for font recognition of a single character and average recognition rates of 96%-100% for font recognition of a string with an identical font were obtained for the six most popular fonts, which shows the feasibility and efficiency of the proposed approach for Chinese character font recognition. The method can be used in character recognition and document analysis applications.

* To whom all correspondence should be send.
@Also with the Department of Information Management, Ming Chuan College, Taipei, Taiwan 111, Republic of China.
✞This work was supported by National Science Council, Republic of China under Grant NSC83-0408-E009-010.

1. Introduction

Huge amounts of printed materials, such as newspapers, journals, magazines, and office documents are published every day. Various document components, such as headlines, text lines, and graphics, are used to distinguish articles and indicate article contents. To more effectively access the information contained within such documents, an automated document entry system is needed to handle different layout styles[1]. Various techniques, such as document image segmentation, character segmentation, and character recognition have been investigated for use in an automated document entry system, but font recognition is seldom considered.

Artificial neural network techniques have recently been applied to many different fields and have demonstrated capabilities in solving complex problems. The error back-propagation method proposed by Rumelhart[2,3] is a representative learning method for hierarchical networks. Hirose et al.[4] proposed a back-propagation algorithm that varies the number of hidden units which can be used to escape local minima and make it unnecessary to decide the number of hidden units. In this study, we use the back-propagation neural network model as a classifier to recognize the Chinese character font because of the high capability of learning and the high speed of recalling of the model.

The Chinese character font recognition problem is more diffucult than that of English because the size of the Chinese character set is larger and the strokes of the Chinese characters are more complicated. Morris[5] used spectral signatures[6] and a statistical analysis method[7] to recognize the English digital typefaces. The samples used in [5] consist of strings selected at random from English text. Each string is rendered in a particular typeface and truncated to an image of 512x64 pixels. Then a 512x64 discrete Fourier transform of the string is computed and the amplitudes at each of the resulting points recorded. In the classification stage using ideal filters, the spatial frequency plane is first partitioned into 38 rectangular regions, and the mean Fourier amplitude of each region is computed and collected as a feature vector. By assuming that the class conditional densities are normal, a piecewise quadratic classifier was used as an approximation to the Bayes minimum error classifier. Jou et al.[8] proposed a universal data compression algorithm to model chinese character patterns and a scheme based on the universal model for recognizing the Chinese character font. They used 10,000 Chinese characters with the same font as an input source sequence to obtain the universal model. After different models were built, a long string of input characters were simultaneously encoded using different models obtained from different character fonts. The character font type is finally decided to be the model that yields the largest compression ratio. In this study, we use features extracted from certain bands of the Fourier amplitude spectrum of each character image as the input of the classifier. An average recognition rate of 86% for font recognition of a single character and average recognition rates of 96%-100% for font recognition of a string with an identical font using an average technique can be obtained for the six most popular fonts. Experimantal results show the feasibility and efficiency of the proposed approach for Chinese character font recognition. It can be used in character recognition and document analysis applications.

The remainder of this paper is organized as follows. A brief review of the Fourier tranform and the back-propagation neural network is given in Section 2. In Section 3, the proposed Chinese character font recognition method is presented. Several experimental results showing the feasibility of the proposed approach are described in Section 4. Finally, some conclusions are given in Section 5.

2. Review of Fourier Transform and Back-propagation Neural Network

2.1 Fourier Transform

A 2D image i may be reguarded as a real-valued function on the plane, where $i(x,y)$ denotes the image intensity at (x,y). The Fourier transform $I(\omega_x,\omega_y)$ of $i(x,y)$ is given by

$$I(\omega_x,\omega_y)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} i(x,y)e^{-j2\pi(x\omega_x+y\omega_y)}dxdy. \quad (1)$$

It can be written in polar coordinates as

$$I(\omega_x,\omega_y) = A(\omega_x,\omega_y)e^{j\phi(\omega_x,\omega_y)}, \quad (2)$$

where $A(\omega_x,\omega_y)=\left|I(\omega_x,\omega_y)\right|$ is the Fourier amplitude at (ω_x,ω_y) and $\phi(\omega_x,\omega_y)$ the Fourier phase at (ω_x,ω_y).

The variables ω_x and ω_y denote the spatial frequencies and roughly represent the rate at which the image is changing in its intensity in the x and y directions, respectively. High spatial frequencies correspond to rapid rates of changes with respect to position, i.e., correspond to small features in the image. On the other hand, low spatial frequencies correspond to slow rates of changes, i.e., correspond to large features. The amplitude denotes the amount of variation at the given spatial frequency. Large amplitudes at a given spatial frequency or block of frequencies accompany prominent features repeating at those frequencies. Thus, the amplitude encodes the amount of variation in features. The phase, on the other hand, encodes the gross position of features. The differences of the Chinese character fonts come from the sizes of the characters, the thicknesses of the strokes, and the slopes of the characters, etc. For these reasons, it is natural to expect that the amplitude distribution of the frequency plane might have font-specific information. The Chinese character fonts do not depend on the positions of the strokes. Thus, we may ignore the effect of phases.

2.2 Back-Propagation Neural Network

Back propagation is often used as the learning algorithm in layered-structure neural networks because of its efficiency. The algorithm is a learning rule which suggests a way of modifying weights to represent a function from input to output. The network architecture incldes an input layer, one or more hidden layer(s), and an output layer. The units on a layer have full connections to the units on the adjacent layers, but no connection to the units on the same layer, as shown in Figure 1. A way of computing the function from the input to the output is to calculate the difference (or the error) between the expected and actual output value and to adjust the weights in order to minimize the error. The algorithm minimizes the error by performing a gradient descent on the error surface in the weight space whose height at any point in the weight space is equal to the error measure. Thus, the amount of the weight change for each input pattern in a learning cycle is proportional to the error; a learning cycle is completed after the network sees all of the input and output pairs. The error is gradually reduced to a certain point as the number of learning cycles is increased. In the case of the three-layered back-propagation neural network model shown in Figure 1, the algorithm assigns the input vector as an output vector of the input layer; no transformation occurs in the input layer. The output vector of the input layer propagates forward to the hidden units to compute the output values of the hidden units, H, with the help of a sigmoid output function defined as follows :

$$H_k = 1 \Big/ \left\{ 1 + \exp\left[-\sum_{i=0}^{n} W_{ki} X_i - \theta_k \right] \right\} \tag{3}$$

where X_i is the output value of the ith unit in the input layer, θ_k is a threshold of the kth unit in the hidden layer, W_{ki} is the connection weight between the ith unit in the input layer, and the kth unit in the hidden layer. The output vector of the hidden layer then propagates forward to the output unit to calculate the output value of the final output layer, O, by the same output function using U_{jk} and H_k instead defined as follows:

$$O_j = 1 \Big/ \left\{ 1 + \exp\left[-\sum_{k=0}^{n} U_{jk} H_k - \theta_j \right] \right\} \tag{4}$$

where θ_j is a threshold of the jth unit in the output layer, U_{jk} is the connection weight between the kth unit in the hidden layer, and the jth unit in the output layer. The output value from the output layer is compared with the desired one to compute a sum of squared errors according to

$$E = \frac{1}{2} \sum_{j=0}^{n} \left(D_j - O_j \right)^2 , \tag{5}$$

where D_j is the desired and O_j is the actual output values of the final output layer computed by (4). The connection weight between the jth unit in the output layer and the kth unit in the hidden layer is modified according to

$$\Delta U_{jk} = \eta \delta_j H_k , \tag{6}$$

where

$$\delta_j = \left(D_j - O_j \right) O_j \left(1 - O_j \right) . \tag{7}$$

The connection weight between the kth unit in the hidden layer and the ith unit in the input layer is modified according to

$$\Delta W_{ki} = \eta \delta_k X_i , \tag{8}$$

where

$$\delta_k = H_k \left(1 - H_k \right) \sum_{j=0}^{n} \delta_j U_{jk} . \tag{9}$$

The value η in (6) and (8) is a scalar quantity which determintes the rate of the gradient descent. In training a network, a small value of η leads to slower learning; however, a large value of η may lead to a local minimum . A moderate value smaller than 1.0 is often found appropriate for η. The learning process is repeated for a number of times until the change of weights become negligible.

3. Proposed Approach to Chinese Character Font Recognition

A Chinese character may be seen as a 2-dimensional pattern with a number of strokes. Each stroke consists of a cluster of black points, and the area outside the strokes is filled with white points. Each character pattern stands for an individual meaning, unlike English words which are spelled in a string of letters from the 26-letter Roman alphabet. We select six types of most popular Chinese fonts for use in this study. Some examples are shown in Figure 2. A scanned document image like the one shown in Figure 3 is first thresholded to a bitmap image. Isolated character patterns with 64x64 in size are obtained after a segmentation and normalization process is performed. The result of such a process of Figure 3 is shown in Figure 4. We use the resulting character samples for learning and testing later. Two methods for font recognition are proposed in this study and are described in the following.

3.1 An integrated method

In general, there are two steps in pattern recognition. The first is the feature extraction step and the other is the classification step. We use the Fourier transform for feature extraction and the back-propogation neural network for classification. A character sample is first transformed into the spatial frequency plane by a 64x64 discrete Fourier transform. The amplitude at each of the resulting points is recorded. Some results were shown in Figure 5, Figure 6, and Figure 7. We can see that the most different amplitudes in those figures are located on the low frequency areas. Then, the spatial frequency plane is divided into 50 non-overlapping square blocks with varying sizes as shown in Figure 8. The mean Fourier amplitude of each square block is computed. Finally we take these mean values to form a feature vector for use in the classification step.

As to the structure of the back-propagation neural network model, we use 50 units in the input layer for the 50-dimensional feature vector, 25 units in the hidden layer, and 6 units in the output layer for the six font types.

3.2 An average technique

In general, font recognition of a string with an identical font is more practical than font recognition of a character only. In this study we also use the technique for single character font recognition described previously to recognize string font recognition by an average principle. The proposed method is to compute, for a given string of characters all with a single font, the following measure, called the Fourier amplitude of the string at (ω_x, ω_y),

$$\overline{A}(\omega_x, \omega_y) = \sum_{i=1}^{n} A_i(\omega_x, \omega_y) \Big/ n, \qquad (10)$$

where n is the length of the string, and $A_i(\omega_x, \omega_y)$ is the Fourier amplitude of the ith character in the string. This method enhances the common property of different characters with a single font by the way of summation, but decreases the effect coming from different characters by the way of averaging. We can also use the technique to recognize the font type of a Chinese text block with some characters which is rendered in an identical font by the way of selecting n characters in the block at random.

4. Experimental Results

The proposed methods have been tested using a personal computer. Figure 9 shows some training characters which are selected randomly from the 5,401 most-frequently used characters. Figure 10 shows a tested document with six blocks. There are 50 characters with a single font in each block. Each block has a distinct font type.

For font recognition of a single character using the integrated method, we made three experiments which are different in the numbers of used training characters. 33 characters were used for training in the first experiment, 100 characters in the second, and 300 characters in the last. The confusion tables of the above three experiments are shown in Table 1, Table 2, and Table 3, respectively. The leftmost columns of Figure 11, Figure 12, and Figure 13 show respectively the classification results of the above three experiments, in which the tested characters are selected sequentially from the tested document with the six blocks shown in Figure 10. The overall correct classification rates of the three experiments are shown in Table 4. We can see that a higher recognition rate comes from the use of a larger number of training characters.

For font recognition of a string with an identical font using the average technique, we use the same classifiers used in font recognition of a single character and we made three experiments which are different in the lengths of tested strings. The tested strings are selected sequentially from the tested document shown in Figure 10. Strings consisting of two characters were tested in the first experiment, five characters in the second, and ten characters in the last. The confusion tables of the first experiment using the three classifiers(i.e., classifiers trained with 33,100, and 300 characters) which were obtained in the experiments for font recognition of single characters are shown in Table 5, Table 6, and Table 7, respectively. The confusion tables of the second experiment using the three classifiers are shown in Table 8, Table 9, and Table 10, respectively. The confusion tables of the last experiment using the three classifiers are shown in Table 11, Table 12, and Table 13, respectively. The rightmost three columns of Figure 11, Figure 12, and Figure 13 show respectively the classification results of the above three experiments. The overall correct classification rates of the three experiments are shown in Table 14. We can see that a higher recognition rate comes from the use of a larger number of training characters and a larger length of the tested string with the same font.

5. Conclusion

A feasible approach to the Chinese font recognition has been proposed. The approach is based on the use of Fourier spectrum features and the back-propagation neural network. Some important results were obtained in this study. First, high recognition rates have been obtained in our experiments. Second, the number of training characters is much smaller than 10,000 which is needed in [8]. Third, we can use the method not only for font recognition of a single character but also for font recognition of a string with identical fonts. The last, the approach is simple and effective. It can be improved by increasing the number of training characters, or by dividing the spatial frequency plane more properly. The font recognition result can be used in Chinese character recognition and document analysis applications.

References

1. T. Akiyama and N. Hagita, "Automated entry system for printed documents," *Pattern Recognition*, vol. 23, pp. 1141-1154, 1990.
2. D. E. Rumelhart, et al., *Parallel Distributed Processing*, vol. 1. London: The MIT Press, 1986.
3. D. E. Rumelhart, G. E. Hinton and R. J. Williams, "Learning representation by back -propagating errors," *Nature*, vol. 323, pp. 533-536, 1986.
4. Y. Hirose, K. Yamashita and S. Hijiya, "Back -propagation algorithm which varies the number of hidden units," *Neural Networks*, vol. 4, pp. 61-66, 1991.
5. R. A. Morris, "Classification of digital typefaces using spectral signatures," *Pattern Recognition*, vol. 25, pp. 869-876, 1992.
6. T. M. Caelli, W. F. Bischof and Z. Q. Liu, "Filterbased models for pattern classification," *Pattern Recognition*, vol. 21, pp. 639-650, 1988.
7. K. Fukanaga, *Introduction to Statistical Pattern Recognition*, 2nd Edn. NY: Academic Press, 1990.
8. I. C. Jou, et al., "Modeling for Chinese character patterns," *Proc. CVGIP*, NanTou, Taiwan, R.O.C., pp. 131-134, 1994.

Table 1. Confusion table of the first experiment.

	font 1	font 2	font 3	font 4	font 5	font 6
font 1	43	0	5	1	1	0
font 2	0	36	0	0	11	3
font 3	5	0	35	6	0	4
font 4	8	0	0	42	0	0
font 5	0	8	2	0	37	3
font 6	6	1	3	0	1	45

Table 2. Confusion table of the second experiment.

	font 1	font 2	font 3	font 4	font 5	font 6
font 1	45	0	3	1	1	0
font 2	0	45	0	0	1	4
font 3	6	0	38	5	0	1
font 4	7	0	2	39	0	2
font 5	0	1	2	0	43	4
font 6	2	2	4	1	0	41

Table 3. Confusion table of the third experiment.

	font 1	font 2	font 3	font 4	font 5	font 6
font 1	40	0	9	0	1	0
font 2	0	47	0	0	1	2
font 3	3	0	42	1	0	4
font 4	1	0	6	42	0	1
font 5	0	5	0	0	42	3
font 6	0	2	2	0	2	44

Table 4. The recognition rates for different numbers of training characters in six Chinese fonts.

experiment	The number of training characters	Learning Cycles	recog. rates for Training characters	recog. rates for Tested characters
1	33	95	99%	79%
2	100	254	96%	84%
3	300	711	96%	86%

Table 5. Confusion table of the first experiment using the average technique by the classsifier which was obtained in the first experiment using the integrated method (the lengths of test strings are two).

	font 1	font 2	font 3	font 4	font 5	font 6
font 1	24	0	1	0	0	0
font 2	0	20	0	0	3	2
font 3	4	0	19	2	0	0
font 4	4	0	0	21	0	0
font 5	0	2	0	0	23	0
font 6	0	0	2	0	0	23

Table 6. Confusion table of the first experiment using the average technique by the classsifier which was obtained in the second experiment using the integrated method (the lengths of test strings are two).

	font 1	font 2	font 3	font 4	font 5	font 6
font 1	25	0	0	0	0	0
font 2	0	23	0	0	0	2
font 3	3	0	20	2	0	0
font 4	2	0	1	22	0	0
font 5	0	0	0	0	25	0
font 6	0	0	2	0	0	23

Table 7. Confusion table of the first experiment using the average technique by the classsifier which was obtained in the last experiment using the integrated method (the lengths of test strings are two).

	font 1	font 2	font 3	font 4	font 5	font 6
font 1	23	0	2	0	0	0
font 2	0	24	0	0	0	1
font 3	0	0	25	0	0	0
font 4	0	0	1	24	0	0
font 5	0	1	0	0	24	0
font 6	0	1	0	0	0	24

Table 8. Confusion table of the second experiment using the average technique by the classsifier which was obtained in the first experiment using the integrated method (the lengths of test strings are five).

	font 1	font 2	font 3	font 4	font 5	font 6
font 1	10	0	0	0	0	0
font 2	0	10	0	0	0	0
font 3	1	0	9	0	0	0
font 4	0	0	0	10	0	0
font 5	0	0	0	0	10	0
font 6	0	0	0	0	0	10

Table 9. Confusion table of the second experiment using the average technique by the classsifier which was obtained in the second experiment using the integrated method (the lengths of test strings are five).

	font 1	font 2	font 3	font 4	font 5	font 6
font 1	10	0	0	0	0	0
font 2	0	9	0	0	0	1
font 3	1	0	9	0	0	0
font 4	0	0	0	10	0	0
font 5	0	0	0	0	10	0
font 6	0	0	0	0	0	10

Table 10. Confusion table of the second experiment using the average technique by the classsifier which was obtained in the last experiment using the integrated method (the lengths of test strings are five).

	font 1	font 2	font 3	font 4	font 5	font 6
font 1	10	0	0	0	0	0
font 2	0	10	0	0	0	0
font 3	0	0	10	0	0	0
font 4	0	0	0	10	0	0
font 5	0	0	0	0	10	0
font 6	0	0	0	0	0	10

Table 11. Confusion table of the last experiment using the average technique by the classsifier which was obtained in the first experiment using the integrated method (the lengths of test strings are ten).

	font 1	font 2	font 3	font 4	font 5	font 6
font 1	5	0	0	0	0	0
font 2	0	5	0	0	0	0
font 3	0	0	5	0	0	0
font 4	0	0	0	5	0	0
font 5	0	0	0	0	5	0
font 6	0	0	0	0	0	5

Table 12. Confusion table of the last experiment using the average technique by the classsifier which was obtained in the second experiment using the integrated method (the lengths of test strings are ten).

	font 1	font 2	font 3	font 4	font 5	font 6
font 1	5	0	0	0	0	0
font 2	0	5	0	0	0	0
font 3	0	0	5	0	0	0
font 4	0	0	0	5	0	0
font 5	0	0	0	0	5	0
font 6	0	0	0	0	0	5

Table 13. Confusion table of the last experiment using the average technique by the classsifier which was obtained in the last experiment using the integrated method (the lengths of test strings are ten).

	font 1	font 2	font 3	font 4	font 5	font 6
font 1	5	0	0	0	0	0
font 2	0	5	0	0	0	0
font 3	0	0	5	0	0	0
font 4	0	0	0	5	0	0
font 5	0	0	0	0	5	0
font 6	0	0	0	0	0	5

Table 14. The recognition rates for different numbers of training characters in six Chinese fonts.

The number of training characters	The length of tested strings		
	2	5	10
33	87%	98%	100%
100	92%	97%	100%
300	96%	100%	100%

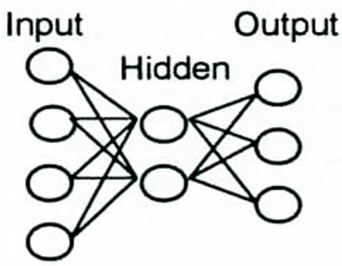

Figure 1. A three-layered neural network model.

交通大學 (font type 1)
交通大學 (font type 2)
交通大學 (font type 3)
交通大學 (font type 4)
交通大學 (font type 5)
交通大學 (font type 6)

Figure 2. Example of Chinese font types.

Figure 3. Part of a scanned character image.

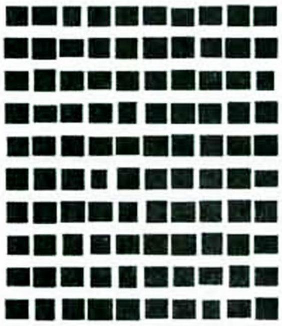

Figure 4. Segmentation results of the image shown in Figure 3.

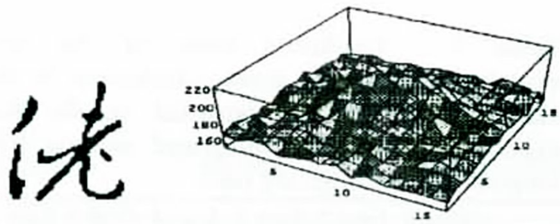

Figure 5. An isolated character pattern of font type 1 and its Fourier transform plane.

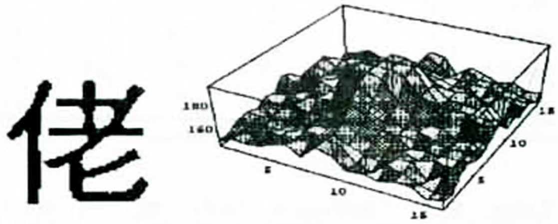

Figure 6. An isolated character pattern of font type 2 and its Fourier transform plane.

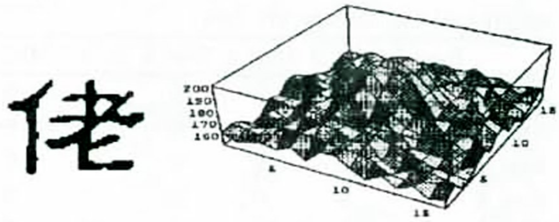

Figure 7. An isolated character pattern of font type 6 and its Fourier transform plane.

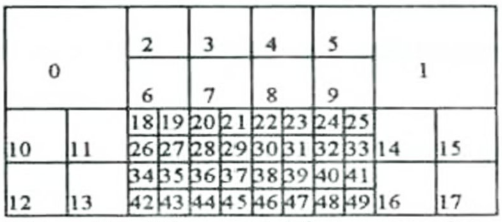

Figure 8. The spatial frequency plane (upper half).


Figure 9. Some training characters.

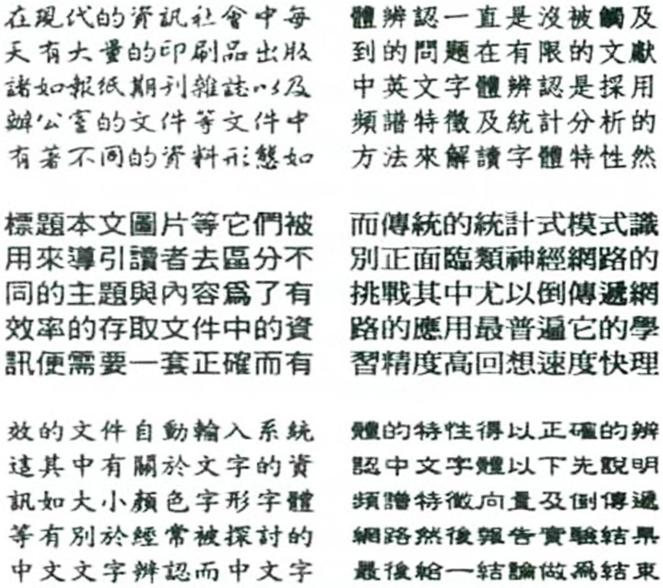

Figure 10. The tested characters in the experiments.

1111111111	11111	11	1
3111113111	11111	11	1
4111531111	11311	11	1
1111111111	11111	11	1
1113113111	11111	11	1
2225222552	25252	22	2
2522225255	22225	22	2
2222252262	22226	22	2
2522255226	22222	22	2
2222622222	22622	22	2
3333634133	33313	33	3
3633433634	33333	33	3
1333363134	11334	33	3
3333331343	33333	33	3
1333433333	13433	13	3
4444441441	44414	44	4
4444414414	44444	44	4
4414444444	41444	44	4
4444144144	44114	44	4
1444444444	44444	44	4
5555563535	55555	55	5
5522255255	52255	55	5
2555655552	55555	55	5
5555552556	55555	55	5
5555255555	55555	55	5
6666636666	66366	66	6
6666636666	66366	66	6
6666656366	66666	66	6
6666662666	66666	66	6
6666666666	66666	66	6

Figure 11. The classification results obtained in the first experiment.

1111111111	11111	11	1
1111113111	11111	11	1
4111511111	11111	11	1
1111111111	11111	11	1
1113113111	11111	11	1
2222222652	22222	22	2
2222222222	22222	22	2
2222222262	22226	22	2
2222222222	22222	22	2
2262622222	22622	62	2
1333633133	33313	33	3
3333433334	33333	33	3
1333333134	13334	33	3
3333331343	33333	33	3
1333433333	13433	13	3
4444464441	44444	44	4
4434444414	43444	44	4
4414444644	41444	44	4
4444144114	44414	44	4
1444444434	44444	44	4
5555563535	55555	55	5
5555555555	55555	55	5
5555655555	55555	55	5
5555555656	55555	55	5
5555255555	55555	55	5
6666636664	66366	66	6
6666636666	66366	66	6
6666662366	66666	66	6
6616662666	66666	66	6
6661666636	66666	66	6

Figure 12. The classification results obtained in the second experiment.

3113111113	11111	11	1
1111113111	11111	11	1
1311531111	11311	11	1
1111111113	11111	11	1
1113111113	13111	11	1
2222222252	22222	22	2
2222222222	22222	22	2
2222222262	22222	22	2
2222222222	22222	22	2
2222622222	22622	22	2
3333633133	33333	33	3
3633336333	33333	33	3
3333333333	33333	33	3
3333331333	33333	33	3
1333433363	33333	33	3
4444444444	44444	44	4
4436344444	43444	44	4
4444444344	44444	44	4
3443444414	44444	44	4
4444444434	44444	44	4
5555565555	55555	55	5
5522555255	52555	55	5
5555655552	55555	55	5
5555555556	55555	55	5
5555255555	55555	55	5
6666636666	66666	66	6
6666636666	66666	66	6
6666662666	66666	66	6
6665662666	66626	66	6
6666656666	66666	66	6

Figure 13. The classification results obtained in the last experiment.