MVA2005 IAPR Conference on Machine VIsion Applications, May 16-18, 2005 Tsukuba Science City, Japan
13-31

Performance Evaluation of Fractal Feature in Recognition of Postal Codes Using an RBF Neural Network and SVM Classifier

Saeed Mozaffari [1,2]
[1] Electrical and Computer Department, Qazvin Azad University, Iran
s_mozaffari@ aut.ac.ir

Karim Faez [2]
[2] Electrical Engineering Department, Amirkabir University of Technology, Tehran, Iran,
kfaez@ aut.ac.ir

Hamidreza Rashidy Kanan [2]
[2] Electrical Engineering Department, Amirkabir University of Technology, Tehran, Iran,
rashidykanan@ aut.ac.ir

Abstract

This paper presents a new method for isolated Farsi/Arabic characters and digits recognition. Fractal codes which are determined by a fractal encoding method are used as feature in this system. Fractal image compression is a relatively recent technique based on the representation of an image by a contractive transform for which the fixed point is close to the original image. Each fractal code consists of six parameters such as corresponding domain coordinates for each range block, brightness offset and an affine transformation. We made a comparison between support vector machine (SVM) which is based on statistical learning theory and radial basis function (RBF) neural network classifiers. Experimental results on our database which was gathered from various people with different ages and different educational background indicate that fractal codes are suitable features in the application of zip code recognition. This system achieves recognition rates of 92.71% and 91.33% for digits and characters respectively.

1 Introduction

The recognition of handwritten characters has been an active research domain in recent years. . Many researches have been concerned with the recognition of Latin, Chinese and Kanji characters. But little researches have been done on Farsi and Arabic. Previous works on recognition of isolated characters, words, and scripts of Farsi and Arabic languages have used structural features [1,2], moment features [3] and Wavelet Transform [4] as a feature. Neural Networks [4] and Hidden Markov Models (HMM) [5] are also used as a classifier in these systems. The feature used in this system is fractal codes which are obtained after some pre-processing tasks by a fractal encoding method. For a given binary image containing single character or digit, two pre-processing tasks are needed to make the system invariant to scale and frame size changes. By finding bounding rectangle box of each character or numeral and scaling it to a 64×64 pixel image, the system will be robust to location and scale changes. Fractal codes represent affine transformations which when iteratively applied to range-domain pairs in an arbitrary initial image, the result is close to the given image. Fractal concept has been used recently by some researches for face recognition [6,7]. Ebrahimpour [6] used the PSNR between feature vectors of the query image and feature vectors of all image in the database as a measure of distance and a minimum distance classifier. Researchers have begun to examine the use of Radial basis function (RBF) neural network for solving function approximation and pattern classification problems. An RBF neural network with HLA [8] learning algorithm has been used for classification in our system.

In this paper we made a comparison between RBF neural network and support vector machine (SVM) classifiers in the application of character and digit recognition using fractal feature.

The rest of this paper is organized as follows. In section 2 an overview of fractal image coding and fractal feature extraction are given. Section 3 and 4 describe the theory of RBF and SVM Classifiers. In section 5, we discuss the experimental results and make a comparison between recognition rate of RBF and SVM classifiers. Conclusion remarks are given in section 6.

2 Fractal Image Compression

With the advent of the information age, the need for mass information storage and retrieval grows. Different image compression methods have been focused for a long time to reduce this massive information, but fractal image compression is a relatively recent technique based on representation of an image by contractive transform, for which the fixed point is close to original image.

Suppose we are dealing with a 64×64 binary image in which each pixel can have one of 256 levels (ranging from black to white). Let $R_1, R_2, \ldots\ldots, R_{256}$ be 4×4 pixel non-overlapping sub-squares of the image (Range blocks), and let D be the collection of all 8×8 pixel overlapping sub-squares of the image (Domain blocks) as depicted in Fig.1. The collection D contains $57 \times 57 = 3249$ squares. For each

R_i block, search through all of D blocks to find a $D_i \in D$ which minimizes equation (1). There are 8 ways to map one square onto another. Each square can be rotated to 4 orientations or flipped and rotated into 4 other orientations as shown in Fig.2. Having 8 different affine transformations means comparing $8 \times 3249 = 25992$ domain squares with each of the 256 range squares.

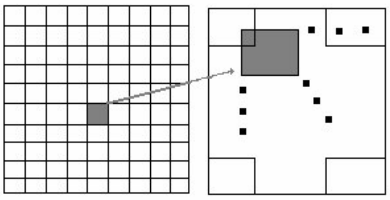

Figure 1. One of the block mapping in partitioned iterated function systems representation

1	2	2	1	3	1	1	3
3	4	4	3	4	2	2	4
4	3	3	4	2	4	4	2
2	1	1	2	1	3	3	1

Figure 2. Eight different affine transformations

$$Collage\ Error = \min \left\| R_i - W(D_{j_i}) \right\|^2 \quad (1)$$

As mentioned before, a D_i block has 4 times as many pixels as an R_i, so we must either sub-sample (choose 1 from each 2×2 sub-square of D_i) or average the 2×2 sub-squares corresponding to each pixel of R_i when we minimize equation (1) [9]. Minimizing equation (1) means two things. First it means finding a good choice for D_i. Second, it means finding a good contrast and brightness setting s_i and o_i for W_i in equation (2).

$$w_i \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} a_i & b_i & 0 \\ c_i & d_i & 0 \\ 0 & 0 & s_i \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} e_i \\ f_i \\ o_i \end{bmatrix} \quad (2)$$

A choice of D_i, along with a corresponding s_i and o_i, determines a map W_i. The type of image partitioning used for the range blocks can be so different. A wide variety of partitions have been investigated, the majority being composed of rectangular blocks. Different types of range block partitioning were described in [10]. In this research we used the simplest possible range partition consists of the fixed size square blocks, that is called fixed size square blocks (FSSB) partitioning.

The procedure for finding a fractal model for a given image is called encoding, compression, or searching for a fractal image representation [11]. After finding the best match, fractal elements which consist of 6 real numbers (a,b,c,d,e,f) are selected as follows. (a,b,c,d) are (x,y) coordinates of the D block and its corresponding R block respectively. (e) is the index of affine transformation that makes the best match. (it is a number between 1 and 8). (f) is the intensity which is a number between 0 and 256. In the case of character recognition, although it is not necessary to decode the fractal models that are obtained from previous section, but we have done it to verify the validation of coding algorithm. Decoding process starts with an arbitrary initial image. Then the decoding algorithm is iterated about 6 to 16 times. The results for different iterations and different R block sizes are depicted in Fig.3. After each iteration, the average of error and peak signal to noise ratio (PSNR) are calculated. (Table1).

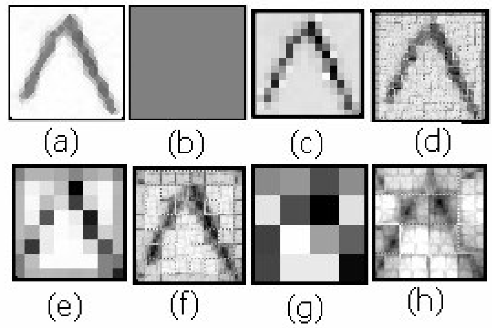

Figure 3. Decoding algorithm's results.
(a) original image. (b) arbitrary initial image
(c) decoded image after 1 iteration for N=4
(d) decoded image after 10 iteration for N=4
(e) decoded image after 1 iteration for N=8
(f) decoded image after 10 iteration for N=8
(g) decoded image after 1 iteration for N=16
(h) decoded image after 10 iteration for N=16

Table.1 Average of error and PSNR versus number of iteration for N=4

Number Of Iteration	Average Of Error	PSNR (dB)
0	2.45	40.32
1	1.95	42.30
3	1.38	45.53
5	1.06	47.55
10	0.89	49.14
15	0.869	49.34

2.1 Fractal Features

Since characters and digits have simple images, we used the fixed size square blocks (FSSB) partitioning so the first two numbers in the fractal codes (a,b) do not have information in feature extraction part. For an 64×64 image with N=16, by omitting (a,b), a feature vector with the length of 64 ,($\frac{64}{16} \times \frac{64}{16} \times 4 = 64$), is obtained for each character or number.

3 Support Vector Machines

One of the most important recent researches in classifier design is the introduction of support vector machines classifier. The basic idea of SVM utilized in pattern recognition is to construct a hyper-plane as decision plane, which separates the positive and negative patterns with the largest margin. The optimization of SVM consists to minimize the number of support vectors by maximizing the margin between the two classes. The decision function derived by the SVM classifier for a two-class problem, can be formulated, using a kernel function K(x, x_i) of a new example x and a training example x_i , as follows:

$$f(x) = \sum_{i \in SV} \alpha_i y_j K(x_i, x) + \alpha_0 \quad (3)$$

Where SV is the support vector set (a subset of training set) and $y_i = \pm 1$ the label of example x_i . The parameters $\alpha_i \geq 0$ are optimized during the training process. Table 2 shows some kernels that are used in SVMs. The kernel function used in our system is Gaussian function because experiment shows that it is better than other kernel functions in handwritten recognition [12].

Table 2. kernels are used in SVMs

Kernel Name	Kernel Function
Linear	$K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T \mathbf{x}_j$
Polynomial	$K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i^T \mathbf{x}_j + 1)^d$
Radial Basis Function (RBF)	$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\lVert \mathbf{x}_i - \mathbf{x}_j \rVert^2 / 2\sigma^2\right)$
Exponential RBF	$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\lVert \mathbf{x}_i - \mathbf{x}_j \rVert / 2\sigma^2\right)$
Perceptron	$K(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\lambda \mathbf{x}_i^T \mathbf{x}_j + \Theta)$

4 RBF Neural Network Structure

The most commonly used family of neural networks for handwritten characters recognition task is feed-forward network, which includes multilayer perceptron (MLP) and radial basis function (RBF) networks. Recently researchers have begun to examine the use of RBF for solving function approximation and pattern classification problems. One of the advantages of RBF neural networks, compared to multi-layer perceptron networks, is the possibility of choosing suitable parameters for the units of hidden layer without having to perform a non-linear optimization of the network parameters [13]. A schematic diagram of an RBF neural network is shown in Fig. 4. The construction of the RBFNN involves an input layer, a hidden layer and an output layer with feed-forward architecture. An RBF neural network with HLA [8] learning algorithm has been used for classification in our system.

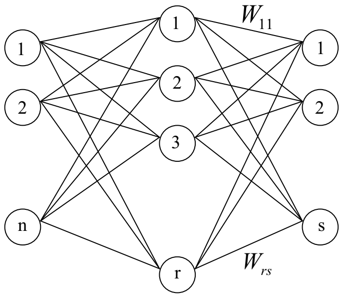

Figure.4 RBF neural network structure

Typically the activation function of the RBF units is chosen as a Gaussian function with mean vector c_i and variance vector c_i as follows:

$$R_i(x) = \exp\left(-\frac{\lVert x - c_i \rVert^2}{\sigma_i^2}\right) \quad (4)$$

Note that σ_i^2 represents the diagonal entries of covariance matrix of Gaussian function.

The Zernike Moment descriptors have such desirable properties: rotation invariance, robustness to noise, expression efficiency, fast computation and multi-level representation for describing the various shapes of pattern [5]. Zernike Moments are well known and widely used in the analysis of optical systems.

5 Experimental Results

In Farsi language, there are ten digits that are shown in Fig.5. Because of similarity between (۰, ۵) and (۲, ۳) especially in the handwritten texts, digits (۰) and (۲) are not used in postal codes in Iran. Thus, we have 8 different classes for digits.

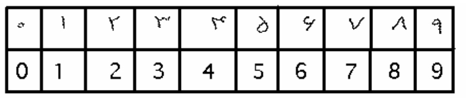

Figure 5. Digits in Farsi

Dots play an important role in Farsi characters. Fig.6 shows four different characters that only differ in the number of and the position of dots. To simplify, we neglect these dots and consider the characters in their main forms without the dots. We categorize the Farsi characters into 8 different classes which are shown in Table.3.

Figure 6. Four Farsi Characters with different dots and similar patterns

Table.3. Final character classes

1	2		3	4	5	6	7	8
ا	ف	ب	ج	د	س	ک	م	ه
ط	ق	پ	چ	ذ	ش	گ		
ظ	ل	ت	ح	ر	ص			
	ن	ث	خ	ز	ض			
			ع	ژ	ی			
			غ	و				

The basic SVM is a binary classifier. There are some strategies for for using SVM as muiti-class clasifier [12]. In this study we used exponential radial basis function kernel with $\sigma = 3$ and set $C = 1000$ with One to Others SVM method.A training and test set, for characters and numerals, were gathered from more than 200 people with different educational background. Our database contains 480 samples per digit (total of 3840), and 190 samples per character (total of 6080). We used 280 samples of each digit for training and the rest (200) for test. We also used 100 samples of each character for training and the rest (90) for the test. Table.4 shows results for different R blocks size (N).

Table 4. Experimental results for different R blocks size

N	Encoding Time (Character /Sec)	Feature Extraction Time(hours)	Average of Error	PSNR (dB)	Size of feature vector	Average of Recognition Rate
4	40.1	110	0.941	48.65	1024	92.1%
8	29.5	81	0.962	48.46	256	92.6%
16	13.9	38.6	1.059	47.63	64	92.2%

As it is shown in Table.4 with the increase in R blocks size, the PSNR and average of error decreases and increases respectively and also algorithm becomes faster. Column 3 in Table 4 shows feature extraction time for our database. Column 6 also shows the average of recognition rate for characters and digits in training and test sets. There is a trade-off between encoding time and average of recognition rate because when N decreases, size of feature vector will increase so the classifier learns more details and its generalization ability become weak. As feature extraction process is faster for N=16 and average of recognition rate is also fair so we encoded input images with this R blocks size.The classification results for characters and numerals are shown in Table.5 for N=16.

Table 5. Experimental results for N=16

	Characters		Digits		Training Time (hours)
	Train set	Test set	Train set	Test set	
SVM	100%	91.33%	100%	92.71%	35.8
RBF	96.2%	90.9%	98.4%	91.70%	22.5

Conclusions

In this paper we made a comparison of RBF neural network and SVM classifiers, whose topologic structure are the same. As shown in Table.5, SVM achieves very perfect recognition rate in training set and it has better generalization ability than RBF classifier but it takes more time to be trained. The feature used in the system is fractal codes which represent each character or digit as a vector with the length of 64. Too large R blocks size (N=32) results in loss of feature; while too small R blocks size (N=4) decrease recognition rate because the generalization ability decrease and the training and test speed decrease also. Computational complexity of fractal encoding is the disadvantage of fractal feature in the application of character recognition which can be removed by adaptive search to speed-up fractal image compression.

6 References

[1] D. M.Dehghan, K.Faez, M.Ahmadi and M.Shridhar “Off-Line unconstrained Farsi Handwritten Recognition Using Fuzzy Vector Quantization and hidden Markov Word Models”.*Proceeding of 15th International Conference on Pattern Recognition,* Vol.2, 2000, pp.351-354.

[2] A.I.S.I.Abuhiba, S.A.Mahmoud and R.G.Green “Recognition of handwritten cursive Arabic characters” ,*IEEE Trans on Pattern Analysis and Machine Intelligence.*Vol.16,No.6, June 1994,PP.664-672.

[3] M.Dehghan and K.Faez, “ Farsi Handwritten Character Recognition with moment invariants”, *Procedding of 13th International Conference on Digital Signal Processing,* Vol. 2,1997, pp.507-510.

[4] A. Mowlaei, K. Faez and A. T. Haghighat, “Feature Extraction with Wavelet Transform for Recognition of Isolated Handwritten Farsi/Arabic Characters and Numerals”, *Proceedings of 14th International Conference on Digital Signal Processing,* Vol. 2, July 2002, pp. 923-926.

[5] P.Temdee, D.Khawparisuth and K.Chamnongthai “Face Recognition by using Fractal Encoding and Backpropagation Neural Network”,*15th ISSPA ,Brisbane ,Australia*, August,1999,PP,159-161.

[6] H.Ebrahimpour,V.Chandran and S.Sridharan “Face Recognition Using Fractal Codes” , Proc. of *ISSPA-01, Kuala Lumpur, Malaysia,* 13-16 August, vol. 1, pp. 359-362, 2001.

[7] C. J. Burges, “A Tutorial on Support Vector Machines for Pattern Recognition”, *Data Mining and Knowledge discovery*, pp.121-167, 1998.

[8] J. Haddadnia, M. Ahmadi, K. Faez, “An Efficient Method for Recognition of Human Face Recognition Using Higher Order Pseudo Zernike Moment Invariant*”, The 5th IEEE Int. Conf. on Automatic Face and Gesture Recognition*” USA, 2002.

[9] Yuval Fisher “ Fractal Image Compression”, SIGGRAPH’92 Course Notes, Technion Israel Institute of Technology from The San Diego Super Computer Center, University of California, San Diego.

[10] Brent Wohlberg and Gerhard Jager,” A Review of Fractal Image Coding Literature”, IEEE Transactions on Image Processing, VOL 8, NO 12, December 1999

[11] . Ning Lu,”Fractal Imaging*”, Academic Press, June* 1997.

[12] ZHAO Bin,LIU Yong and XIA Shao-wei,” Support Vector Machine and its Application in Handwritten Numeral Recognition”. *Proceedings of 15th International Conference on Pattern Recognition ,* Vol.2, pp 720-723,2000.

[13] Lu Yingwei , Narashiman Sundararanjan and P.saratchandran, “ Performance Evaluation of a Sequential Minimal Radial basis function (RBF) Neural Network Learning Algorithm”. *IEEE Transactions on Neural NETWORKS,* Vol 9,NO.2, MARCH 1998.