Handwritten Signature Verification based on Neural 'Gas' Based Vector Quantization

Bai-ling Zhang[1], Min-yue Fu[1] and Hong Yan[2]
[1]Department of Electrical and Computer Engineering
The University of Newcastle, Callaghan, NSW 2308, Australia
[2]Department of Electrical Engineering
The University of Sydney, NSW, 2006, Australia

Abstract

This paper propose a vector quantization (VQ) technique to solve the problem of handwritten signature verification. A neural 'gas' model is trained to establish a reference set for each registered person with handwritten signature samples. Then a test sample is compared with all the prototypes in the reference set and the system outputs the label of the writer of the word. Several different feature extraction methods are compared and good results have been obtained by the VQ technique.

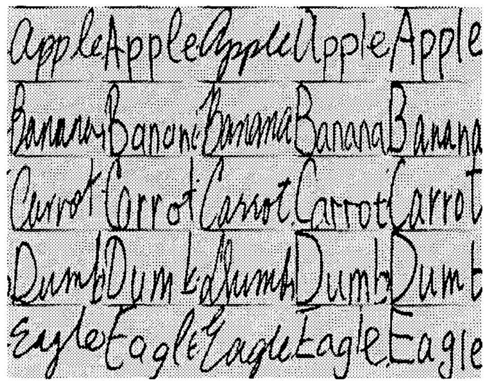

Figure 1. Some samples of handwritten words (half size of original images).

1. Introduction

Signature verification is an important research topic in the area of biometric verification, which aims to extract information of handwriting to establish the identity of the writer, irrespective to the handwritten content. Depending on the method of data acquisition, writer identification systems are mainly divided into online and offline categories. The former means the traces in a writing process are provided by digitizing tablets, instrumented pens, etc, in real time. In an offline system, handwriting images are collected by scanners or cameras when the handwritings have already been produced. Because it is difficult to extract individual features from static images or to detect imitations, off-line signature verification is usually more difficult than on-line verification.

In recent years, neural network techniques have been applied to signature verification with satisfactory results, which are potentially more tolerant and robust when dealing with the intricacies of real data. In this paper, we proposed a modular verification system based on the vector quantization (VQ) technique. A number of independent codebooks are established for individual words/person and classification is set up on a reconstruction scheme called population decoding. Preliminary experiments demonstrate good results.

2 Data preparation, preprocessing and feature extraction

The database comprised of 3770 signatures that were collected from 19 people. The data preparation was mainly divided into two stages. In the first stage, 19 writers were asked to write a set of 5 English words, with each word 26 times. The pen or pencil used by each writer is not prescribed but words are written within a predrawn 5 x 13 grid on A4 paper. These signatures were scanned into the computer using a 8-bit gray-scale, 150 dot-per-inch resolution. The individual images are extracted and labeled with both the writer identification number and the word class number. They are converted to binary images by thresholding. The number of samples from this first stage collection is 19 x 26 x 5 = 2470. In the second stage which was four months after the first collection, 10 of the writers were again asked to provide 26 x 5 words, yielding another 1300 samples.

As a preprocessing step, each signature image is first normalized. By normalization, all images are aligned to a fixed dimension of 240 by 120, i.e., each handwritting has 28,800 pixels. In Fig.1, we give some samples of signature image after normalization.

Feature extraction is an important step in establishing a handwriting verification system, which transform a data space into a feature space. We tried several simple techniques to extract global characteristics of input image. In the first method, we segmented an image into 24 x 12 subregions and the number of pixels in each subregion are counted. The resulted feature vectors are used as training samples. A smaller segmentation size 16 x 8 has also been tried. In the second technique, we use a simple gray scale transformation feature extraction. By this method, a neighborhood averaging filter 3 x 3 is applied to the normalized signature images twice. In the first operation, an image is divided into 80 x 40 regions. In each region, the average value is chosen to be its representative. Therefore, the image is down sampled to a size of 80 x 40 with gray scale values. In the same way, second operation yields a 14 x 27 gray scale image. The transformation process is illustrated in the second row of Fig.2

3 Neural gas model for vector quantization

Vector quantization techniques encode a data set V using a finite set of reference vectors. A reference vector w(m) is considered to best match a data vector x in the sense that appropriately defined distortion measure such as the squared error $||\mathbf{x}-\mathbf{w}(m)||^2$ is minimal.

In the neural gas model [1-2], the reference vectors w(m) are adapted by the relative distance between the neural units within the input space. Each time an input x is presented, we first make an ordering of the elements of a set of distortions $E_{\mathbf{x}} = \{||\mathbf{x}-\mathbf{w}(m)||, m = 1,\cdots,M\}$ and then determine the adjustment of reference vector w(m). Denote the ranking of distortion set as $(E_{\mathbf{x}}(m_0), E_{\mathbf{x}}(m_1),\cdots,E_{\mathbf{x}}(m_{M-1}))$, with $\mathbf{w}(m_0)$ being closest to x, $\mathbf{w}(m_k)$, k = 0, , M − 1 being the reference vector for which there are k vectors w(j) with

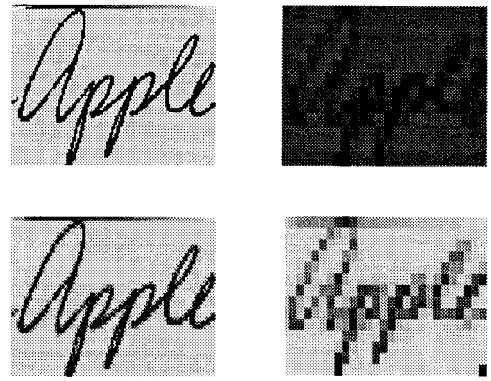

Figure 2. The upper left is an original normalized image; The upper right is the equalized image of the global feature extracted by 24 x 12 segmentation. The two images in the second row show the gray scale features extracted by the twice neighborhood averaging [3 x 3] down sampling.

$||\mathbf{x}-\mathbf{w}(j)|| < ||\mathbf{x}-\mathbf{w}(m_k)||$. Then, each neuron adjusts its weight via dynamical learning rate which depends on the ranking of its distortion. Denote the number k associated with each unit m by k_m. The following learning rule is the neural 'gas' algorithm [2].

$$\Delta\mathbf{w}_k(m) = \mu_k h_\lambda(k_m)(\mathbf{x}-\mathbf{w}_k(m)), m = 1,\cdots,M \quad (1)$$

where $h_\lambda(k_m)$ is 1 for $k_m = 0$ and decays to zero for increasing k_m with characteristic decay constant. In the simulaiton we choose the same one as in [2], $h_\lambda(k_m) = exp(-k_m/\lambda)$, λ is a characteristic decay constant.

4 Modular classification and verification system

In traditional VQ technique, decoding is taken by simple table look-up. In our scheme, reconstruction is based on the concept of population decoding [6]. In a neural gas model, mth reference vector w(m) provides a partial description of input x and a simple representation $\hat{\mathbf{x}}$ is the center of gravity of w(k), $k = 1,\cdot\cdot,M$, weighted by corresponding "virtual" activities. A "virtual" activity a_k can be specified as response function of the kth unit after the network has been organized, which can be taken as a Gaussian function, i.e.,

$$a_k = \exp(-\frac{||\mathbf{x}-\mathbf{w}(k)||^2}{2\rho_k^2}) \quad (2)$$

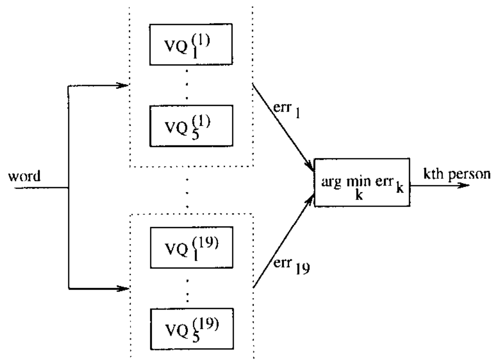

Figure 3. The classification/verification system based on vector quantization.

In the experiment; we simply choose $\rho_k = 1$, without considering the topological relationships among the reference vectors.

Our modular verification scheme can be shown in Fig.3. For a test signature image x, the content is known, so it is only necessary to compare it with 19 codebooks with same contents. In each codebook, the representation vector $\hat{\mathbf{x}}$ can be written as

$$\hat{\mathbf{x}}^{(l)} = \frac{\sum_k a_k^{(l)} \mathbf{w}^{(l)}(\mathrm{k})}{\sum_k a_k^{(l)}}, \; l = 1, \cdots, 19 \tag{3}$$

from which errors $err_l, l = 1, \cdots, 19$, are obtained:

$$err_l = \|\mathbf{x} - \hat{\mathbf{x}}^{(l)}\|^2, \quad l = 1, \cdots, 19 \tag{4}$$

where 1 indicates the index of codebook. We build a classifier by using a decision module which compare the reconstruction errors between the reconstructed vectors and presented pattern, i.e., we assign x to the class c^* where

$$c^* = \underset{c}{\mathrm{argmin}} \; \mathrm{err}, \tag{5}$$

In our first experiment, the first 20 signature images of each word provided by 19 people were used as training and the remaining images as testing. During training, feature vectors from the signature images of one class are presented to the corresponding gas model with appropriate M units. Learning is proceeded in several cycles for the samples in each class. The parameter μ in eqn (1) is initially set to 1 and then linearly decreases to 0.1. λ is dynamically changed from 20 to 0.1. For comparison purpose, different features have been tested, with results shown in Table 1. In the experiment, each VQ has 16 reference vectors.

Table 1. Verification rate (%) for different input features. In the table, feature I is based on 12 x 24 segmentation, feature II 8 x 16 segmentation, feature III 3 x 3 down sampling twice.

feature type	I	II	III
traing set	100%	100%	100%
testing set	93.7%	90%	91.85%

In handwritten signature verification, time-drifting effect is a difficult problem, i.e., handwriting varies over time. A simple remedy is to incorporate the time-drifting knowledge into training. In other words, training set should comprise signature samples collected from different periods. From this viewpoint, we proceeded another experiment with the first 15 samples from each collection stage as training data and the remaining samples in the two collections as testing data. The result is illustrated in Table 2, from which we can see that the recognition accuracy is still acceptable.

Table 2. Verification rate (%) for different input features from third experiment.

feature type	I	II	III
traing set	*99.8%*	*97.94%*	*98%*
testing set	*92.7%*	*89.8%*	*92.59%*

5 Conclusion

In this paper we provide some preliminary experiment results of using VQ technique in offline signature verification. Different feature extraction methods have been compared. For the difficult time-drifting effect of handwriting, we suggested a reasonable remedy by incorporating the time-variations into training.

References

[1] T.M.Martinetz, etc, Neural 'gas'network for vector quantization and its application to time-series prediction, *IEEE Trans.* Neural *Networks,* 4, 558-568, 1993.

[2] T.M.Martinetz, K.J.Schulten, A neural 'gas' network learns topologies, in *Artificial* Neural *Networks* T.Kohonen et al eds., Elsevier Science Publishers, 397-402, 1991

[3] H.P.Snippe, Parameter extraction from population codes: a critical assessment, Neural *Computation, 8,* 511-529, 1996.