Neuro-pattern classification using Zernike moments and its reduced set of features

by P. Raveendran, Sigeru Omatu and S. H. Ong

The paper proposes a neural network technique to classify numerals using Zernike moments that are invariant to rotation only. In order to make them invariant to scale and shift, we introduce modified Zernike moments based on regular moments. Owing to the large number of Zernike moments used, it is computationally more efficient to select a subset of them that can discriminate as well as the original set. The subset is determined using stepwise discriminant analysis.

The performance of a subset is examined through its comparison to the original set. The results are shown of using such a scheme to classify scaled, rotated, and shifted binary images and images that have been perturbed with random noise. In addition to the neural network approach, the Fisher's classifier is also used, which is a parametric classifier. A comparative study of their performances shows that the neural network approach produces better classification accuracy than the Fisher's classifier. When a suitable subset of Zernike moments is used, the classifiers perform well, just like the original set. The performance of the classifiers is also examined. The computational time is greatly reduced when a suitable subset of Zernike moments is used.

1 Introduction

The features used in pattern classification should be invariant to shift, scale and rotation. To this end, regular moments [1, 2] and Zernike moments [3] have been used in pattern classification. When using these features in pattern classification, conventional classifiers performed well under noiseless conditions but were not robust to noise. To overcome this disadvantage, the neural network approach has been proven to perform better because of its generalisation ability and its robustness to noise [4–6]. We have used neural networks to generate regular moments and classify them to one of the classes. The generated regular moments are invariant to changes in shift and scale, but are not invariant to any change in rotation [4].

Khotanzard et al. [5] considered Zernike moments in the classification of images for both neural network and traditional classifiers. They normalised the image that has undergone some change in scale, by enlarging or reducing the image to a standard size. When the image is shifted, it is moved to a standard location so as to nullify the effect of positional changes of the image. A disadvantage of the scaling scheme, however, is that it is often required to scale the image by numbers that are non-integers. Likewise, the shifting of the image also introduces quantisation errors as the computed centroid may not be integer numbers. As it is not practical to use non-integer numbers, there are almost always quantisation errors which are highly dependent on the size of the image plane. This may lead to errors in classification.

Designing a pattern classifier to perform well requires not only a good set of input features which must be tailored separately for each problem domain, but also the number of such features to be used. Once the number of features is considered, then a pertinent question is whether a subset of the considered features can discriminate as well as the original set of features. By using only a subset of the considered features, the computational time can be greatly reduced. The question is how to select a subset of features that could classify as well as the original set of features. Khotanzard et al. [5] selected a
subset of Zernike moments by picking from their lowest to highest order. This method of selection did not produce good classification results as the discrimination power does not monotonically depend on the increasing order.

In this paper, to overcome the shortcomings of normalising the image, we propose a technique that does not use any form of preprocessing on the images, by introducing modified Zernike moments based on regular moments that are not only invariant to rotation, but also invariant to scale and shift. Furthermore, to select a suitable subset of Zernike moments from the original set, we use the stepwise discriminant method. Judging from experimental results, we are able to determine a subset of Zernike moments such that its performance is almost equal to the original set under noiseless and noisy conditions.

To compare the effectiveness of the classification using the proposed neural network, we consider the Fisher's classifier, which is a well known parametric classifier. We discuss the Zernike moments that are invariant to rotation, and in order to make them invariant to scale and shift, we introduce modified Zernike moments based on regular moments. Furthermore, to select a suitable subset of Zernike moments from the original set, we use the stepwise discriminant method. Judging from experimental results, we are able to determine a subset of Zernike moments such that its performance is almost equal to the original set under noiseless and noisy conditions.

2 Zernike moments

Zernike moments are widely used in the analysis of optical systems. Zernike polynomials are an orthogonal set of polynomials of the following form:

$$V_n(x, y) = V_n(r \cos \theta, r \sin \theta) = R_n(r) \cos \theta$$

where $V_n(x, y)$ denotes a complete set of complex-valued polynomials which are orthogonal in the interior of the unit circle $x^2 + y^2 = 1$; $n$ represents the degree of the polynomial; $l$ represents its angular dependence; $R_n(r)$ represents a real-valued set of orthogonal polynomials inside the unit circle; and $r$ and $\theta$ are polar co-ordinates. These polynomials are orthogonal and satisfy

$$\int_{x^2 + y^2 < 1} [V_n(x, y)]V_m(x, y) dx dy = \frac{\pi}{n + 1} \delta_{nm}$$

with

$$\delta_{nm} = \begin{cases} 1 & a = b \\ 0 & \text{otherwise} \end{cases}$$

The complex Zernike moments are defined as [8]

$$A_n = \frac{n + 1}{\pi} \int_0^{2\pi} \int_0^a f(r \cos \theta, r \sin \theta)R_n(r)e^{-i\theta} r dr d\theta$$

where $n = 0, 1, 2, \ldots, \infty$, and $l$ takes on positive and negative integer values subject to the condition $n - |l| = \text{even}$, $|l| = n$. $R_n(r)$ can be expanded in powers of $r$ using

$$R_n(r) = \sum_{k=1} B_{nk} r^k$$

where

$$B_{nk} = \frac{(-1)^{n-k}|n+k|!}{[(n-k)!|l+k|]!(k-l)!}$$

and $n$ denotes the order of the moment.

If an image is rotated by $\theta$, it can be shown that the Zernike moments of the rotated image $A_n^\theta$ and those of the unrotated image $A_n$ are related by

$$A_n^\theta = A_n e^{-i\theta}$$

Each Zernike moment acquires a phase shift on rotation and the magnitude of the Zernike moments of a rotated image remain identical to those before rotation. As $|A_n| = |A_n^\theta|$, we can concentrate on $l = 0$ for the Zernike Zernike moments. The Zernike moments are only rotation invariant and to make them scale and translation invariant the regular moments ($M_{m\ell}$) are used.

The two-dimensional $31 \times 31$ handwritten binary image is first mapped onto a square defined by $x \in [-1, 1]$ and $y \in [-1, 1]$. The regular moment for a digital binary image can now be represented as

$$M_m = \sum_{\alpha=-1}^{1} \sum_{\beta=-1}^{1} x^\alpha y^\beta f(x, y)$$

It has been shown [8] that the Zernike moments and regular moments are related by

$$A_n = \frac{n + 1}{\pi} \sum_{k=1 \atop k \text{even}} \sum_{l=0}^{n-k} \left(\frac{q}{m}\right) \left(\frac{m}{l}\right) B_{nk}$$

However, eqn. 8 is only rotation invariant, as shown in eqn. 6. To make the regular moments invariant to translation, we can define the central moment as

$$u_m = \sum_{\alpha=-1}^{1} \sum_{\beta=-1}^{1} (x-x)\alpha (y-y)\beta f(x, y)$$

with

$$x = \frac{M_{10}}{M_{00}}, \quad y = \frac{M_{01}}{M_{00}}$$

The central moments can be made invariant to scale by using new moments $\eta_m$, defined [1] as

$$\eta_m = \frac{u_m}{u_0 \left(\frac{q}{2} + 1\right)}$$

In order to make Zernike moments invariant to shift, scale and rotation, we introduce a modified Zernike moment, given by

$$A_n = \frac{n + 1}{\pi} \sum_{k=1 \atop k \text{even}} \sum_{l=0}^{n-k} \left(\frac{q}{m}\right) \left(\frac{m}{l}\right) B_{nk}$$

where $q = (k - 1)/2$. $A_n$ is the same as $A_n$ when $x = y = 0$ and $\mu_0 = 1$, which means the original image has been normalised. To
simplify the notation, we call the modified Zernike moment $A_k$ a Zernike moment, and it is also denoted as $A_k$.

Table 1 lists the rotation, scale, and shift invariant Zernike moments and their corresponding numbers from order to order 12. The orders 0 and 1 are not considered because they give results equal to $1/\pi$ and 0, respectively, for all images [8].

### 3 Fisher's discriminant function

In multiple discriminant analysis, the objective is to find an axis with the property of maximising the ratio of between-groups to within-groups variability of projections onto this axis. With $K$ groups and $q$ predictor variables, there are $\min(q, K - 1)$ possible discriminant axes [10] in total. When a population can be partitioned into $K$ distinct groups $G_1, G_2, G_3, \ldots, G_K$, and a set of observations $x = [x_1, x_2, x_3, \ldots, x_q]^T$ is given, the training sample is used to develop a discriminant criterion to classify each observation into one of the groups, in such a manner that the misclassification error rate is minimised. Here, $T$ denotes transposition of a vector. The derived discriminant criterion from this data set is then applied to classify a second set of unknown data.

The Fisher's classification procedure based on sample discriminants is said to assign a pattern vector $x$ to group $G_\alpha$ if

$$\sum_{j=1}^{q} [\tilde{e}(x - \bar{x}_\alpha)]^2 \leq \sum_{j=1}^{q} [\tilde{e}(x - \bar{x}_\beta)]^2$$

for all $i \neq j, r \leq s$ (12)

where $s = \min(q, K - 1)$ is the number of discriminants, $\bar{x}_\alpha$ is the sample mean vector and $\tilde{e}_\alpha$ is the vector of coefficients that maximises the ratio

$$\frac{\tilde{e}_\alpha^T \tilde{e}}{\tilde{e}_\alpha^T \tilde{W}_\alpha}$$

(13)

Here $\tilde{W}_\alpha$ is the sample covariance matrix between groups matrix and $\tilde{W}$ is the sample covariance matrix within groups matrix as shown elsewhere [10].

### 4 Stepwise selection of features

Owing to the large number of features considered, it is computationally more efficient to select a subset of features that would discriminate as well as the full set. There are many number of techniques available for variable selection in statistical discriminant analysis; the all-subsets selection, stepwise selection and selection by canonical variate analysis [11]. Stepwise selection of variables using the $F$ statistics is a popular method. This procedure is based on the assumptions of multivariate normality and covariance homogeneity for the given variables. The Box–Cox transformation [12]

$$X^{(\lambda)} = \begin{cases} x^\lambda & \lambda \neq 0 \\ \ln x & \lambda = 0 \end{cases}$$

(14)

for $x = |A|_a$ has been applied to each variable in each group to make the data 'more normal'. The transformed data are plotted for each variable and are found to 'look normal'. The hypothesis of the equality of the covariance matrices has been tested using Bartlett's test and is found to be acceptable at a significance level of 0.1.

Suppose there are $K$ groups and $q$ predictor variables (Zernike features). The Zernike features with the highest $F$ value exceeding a critical value $F_1$ are selected first. Assume that $[x_1, x_2, \ldots, x_q]$ are denoted by the Zernike features. Suppose $x_1, x_2, \ldots, x_n$ have been already selected. Partial $F$ statistics to test for differences in the $K$ groups are then computed for each $x_{n+1}, \ldots, x_q$ that are not selected. The variable with the largest $F$ value exceeding $F_1$ is chosen from $x_{n+1}$ to $x_q$. If this variable is $x_{n+1}$, the variables selected are now $x_1, x_2, \ldots, x_n, x_{n+1}$. Partial $F$ values for each $x_{n+1}, \ldots, x_q$ are calculated and, for variables with a partial $F$ value less than a critical value $F_2$, the variable with the lowest $F$ value is dropped. The procedure stops when no variables can be selected or deleted [11].

### 5 Experimental study

In this Section, we describe the performance of the neural net and Fisher's method to classify the ten numerals. Two experiments are carried out in this study. In the first experiment, only the Zernike features of 200 noiseless binary images are used as training examples, and the performance of the neural net and Fisher's classifier is examined by testing the remaining sets of data which include noiseless and noisy images. In the second experiment, the number of input features is varied between 5 and 47, and the performance of the classifiers is examined.

**Fig. 1** Four types of each numeral from 0 to 9

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>0</td>
</tr>
</tbody>
</table>

**Table 1** Zernike moments and corresponding number of features

<table>
<thead>
<tr>
<th>order</th>
<th>moments</th>
<th>number of moments</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>$A_{21}, A_{22}$</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>$A_{31}, A_{32}, A_{33}$</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>$A_{11}, A_{22}, A_{33}$</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>$A_{41}, A_{42}, A_{43}$</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>$A_{33}, A_{43}, A_{53}$</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>$A_{44}, A_{54}, A_{64}$</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>$A_{55}, A_{65}, A_{75}$</td>
<td>5</td>
</tr>
<tr>
<td>9</td>
<td>$A_{66}, A_{76}, A_{86}$</td>
<td>5</td>
</tr>
<tr>
<td>10</td>
<td>$A_{77}, A_{87}, A_{97}$</td>
<td>6</td>
</tr>
<tr>
<td>11</td>
<td>$A_{88}, A_{98}, A_{108}$</td>
<td>6</td>
</tr>
<tr>
<td>12</td>
<td>$A_{99}, A_{109}, A_{119}$</td>
<td>7</td>
</tr>
</tbody>
</table>

*INTELLIGENT SYSTEMS ENGINEERING WINTER 1994*
Fig. 2 Scaled images of numerals from 0 to 9; scale 0.69, 0.77, 1.0, 1.07 and 1.15 (top to bottom)

Fig. 3 Rotated images of numerals from 0 top 9; 30°, 60°, 150°, 225° and 300° (top to bottom)

Fig. 4 56 scaled, rotated and translated images of numeral 9 in the data set; note slight variation in shape in first column

Fig. 5 Numerals from 0 to 9 with different levels of noise; SNR is noiseless, 45 dB, 35 dB, 30 dB and 20 dB (top to bottom)

5.1 Data sets

Four slightly different types of noiseless binary image of numerals 0 to 9 of 31 x 31 are generated. Each noiseless binary image of a numeral is then scaled, rotated and translated to produce 56 noiseless binary images for each class of numerals. Fig. 1 shows the four slightly different types of numerals from 0 to 9. Fig. 2 is the scaled binary image of numerals 0 to 9. Fig. 3 shows rotated images of numerals 0 to 9. A complete set of numeral 9 is shown in Fig. 4. In addition to the above noiseless sets, three other sets of noisy images of SNRs 45, 35, 30 and 20 dB are generated. This is done by perturbing each binary image of a numeral with a random noise. Fig. 5 depicts numerals 0 to 9 with different SNRs.

The noisy binary image is generated by randomly selecting some of the 961 pixels of the noiseless binary image and reversing their values from 0 to 1, or vice versa. This random pixel selection is done according to a uniform probability distribution between 1 and 961. The SNR is computed using 20 log[(961 - N)/N], where N is the number of pixels difference between a noisy image and a noiseless image.

5.2 Description of experiments

As two different types of classifiers are used, it is important to establish how much they differ, when the original set of 47 Zernike features are used. We train 200 noiseless binary images to determine the performance of the neural network and Fisher's classifier using the original set of Zernike features.

The back-propagation learning algorithm [9] is used in training the neural network. The number of neurons used in the hidden layer is varied between 10 and 100. 47 inputs are used when the original set of Zernike features are considered. In the experiments where a subset of the original set of features is used, the number of inputs is varied between 5 and 47. The training samples are presented in a random order. The training features are normalised to have zero mean and unit variance. This is necessary to ensure that a subgroup of the features does not dominate the weight adjustment process during training. The mth feature is normalised

\[ l_m = \frac{l_m - \bar{l_m}}{\sigma_m} \]

where \( \bar{l_m} \) and \( \sigma_m \) are the sample mean and standard deviation of the mth training features of all classes. The number of iterations is 500, and a learning rate of 0.4 and a momentum value of 0.3 are used in the training of the neural set.

<p>| Table 2 Performance of classifiers when 200 samples of noiseless images are trained |
|------------------------------------------|-----------------|-----------------|-----------------|-----------------|</p>
<table>
<thead>
<tr>
<th>number of hidden units</th>
<th>noiseless</th>
<th>SNR 45 dB</th>
<th>SNR 35 dB</th>
<th>SNR 30 dB</th>
<th>SNR 20 dB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neural network</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>100°</td>
<td>96.6</td>
<td>92.8</td>
<td>87.6</td>
<td>75.6</td>
</tr>
<tr>
<td>15</td>
<td>100°</td>
<td>95.6</td>
<td>92.6</td>
<td>86.0</td>
<td>76.4</td>
</tr>
<tr>
<td>20</td>
<td>99.7</td>
<td>96.8</td>
<td>92.4</td>
<td>87.8</td>
<td>75.8</td>
</tr>
<tr>
<td>25</td>
<td>99.7</td>
<td>96.2</td>
<td>92.2</td>
<td>86.8</td>
<td>76.4</td>
</tr>
<tr>
<td>30</td>
<td>100°</td>
<td>97.5</td>
<td>93.2°</td>
<td>89.0°</td>
<td>78.5°</td>
</tr>
<tr>
<td>40</td>
<td>100°</td>
<td>96.8</td>
<td>92.8</td>
<td>87.8</td>
<td>77.2</td>
</tr>
<tr>
<td>50</td>
<td>100°</td>
<td>96.6</td>
<td>92.6</td>
<td>87.8</td>
<td>77.4</td>
</tr>
<tr>
<td>60</td>
<td>100°</td>
<td>96.6</td>
<td>92.2</td>
<td>87.8</td>
<td>77.8</td>
</tr>
<tr>
<td>70</td>
<td>100°</td>
<td>96.6</td>
<td>92.8</td>
<td>86.0</td>
<td>77.2</td>
</tr>
<tr>
<td>80</td>
<td>100°</td>
<td>96.0</td>
<td>92.8</td>
<td>86.2</td>
<td>77.0</td>
</tr>
<tr>
<td>90</td>
<td>100°</td>
<td>95.6</td>
<td>93.0</td>
<td>87.6</td>
<td>77.2</td>
</tr>
<tr>
<td>100</td>
<td>100°</td>
<td>95.4</td>
<td>92.8</td>
<td>87.4</td>
<td>77.4</td>
</tr>
<tr>
<td>Fisher’s classifier</td>
<td>100°</td>
<td>93.0</td>
<td>87.8</td>
<td>81.9</td>
<td>68.0</td>
</tr>
</tbody>
</table>

* denotes highest value.
In the first experiment, the features of binary images of Fig. 1 and their scaled version are used as training data for both the neural network and Fisher's classifier. The performance of the classifiers is tested with the remaining data set. Table 2 shows the performance of the classifiers when 200 samples of features of noiseless binary images are trained. The performance of the neural network is around 96% for SNR of 45 dB, but drops to about 77% for 20 dB. As shown in the results, the performance of the neural network is generally better than the Fisher's classifier when it is tested by noisy sets of data.

In the second experiment, the performance of the classifiers is examined using a subset of the Zernike features. The stepwise selection method based on $F$ statistics is used in selecting the features. Using eqn. 14, each Zernike moment of the same class is transformed by varying $\lambda$ such that an approximate normal curve is obtained for each Zernike moment of each class. After transforming the Zernike moment, the stepwise selection method ranks the Zernike moments based on the $F$ statistics. In the selection of five features, the stepwise method selected $A_{11}, A_{22}, A_{42}, A_{12,4}$, and $A_{14}$. When we increased the number of features to ten, then the selected features include the five features already selected plus the new ones, which include $A_{31}, A_{14}, A_{81}, A_{12,0},$ and $A_{11,3}$. Similarly, the selection of 15 features includes the ten features already selected plus five new features; $A_{32}, A_{44}, A_{51}, A_{11,2}$, and $A_{10,2}$. The process of selecting 20 and more features follows the same procedure as mentioned in Section 4.

The training set is the same as in the first experiment, and the number of hidden units used is 10, 15, 20, 25 and 30. A summary of the obtained results is plotted in Fig. 6. For each case, only the best classification accuracy is plotted, obtained by varying the number of hidden layer nodes, and the corresponding number of hidden nodes is denoted above or below each point. Figs. 6a–e shows the performance of the classifiers when a varying number of
input features and hidden units is trained. When 15 features are used, the computational time are reduced by as much as 63% compared with the original set of Zernike moments.

6 Conclusions
In this paper, we have considered a neuro-pattern classification using a reduced feature set to classify numerals. The feature set consists of Zernike moments. Zernike moments are only rotation invariant, and in order to make them scale and translation invariant, the regular moment-based technique was used. The data set consists of noiseless and noisy images. The SNRs were 45, 35, 30, and 20 dB. The discrimination power of the proposed Zernike moments and the feature selection method were tested by a series of experiments. Two experiments were carried out in this study. In the first experiment, we used all of the Zernike features as inputs to the neural network and Fisher's classifier, and in the second experiments the stepwise selection method was used in selecting the features.

It can be observed from the experimental results that a subset of features used as inputs to the classifiers performed just as well as the original set of features. The computational time was greatly reduced when a subset of the original set of features was used.

The problems are to determine how many neurons of the neural network are required and what order of Zernike moments should be used for the original set.

7 References

© IEE: 1994

The paper was first received 7 January and in revised form 21 September 1994.

P. Raveendran and S. Ong are with the Department of Electrical Engineering, University of Malaya, 59100 Kuala Lumpur, Malaysia; Sigeru Omatsu is with the University of Tokushima, Japan.