

Efficient classification by neural networks using encoded patterns

C.N.S. Ganesh Murthy and Y.V. Venkatesh

Indexing terms: Neural networks, Pattern classification

The corners and the middle points, which are extracted as features from the line approximation of a given pattern, are overlaid on a radial grid to form the input array for training a backpropagation network for classification. The proposed method is shown to be simple and robust by extensive testing of its performance on patterns both with and without noise.

Introduction: Automatic recognition of handwritten characters and natural patterns has been one of the most difficult problems in artificial intelligence. Artificial neural networks (ANNs) constitute an important class of computational models for handling this class of problems. Many neural techniques have been proposed to solve this problem, such as those in [1 - 4]. We propose an efficient way of preprocessing and encoding patterns to achieve better classification by the backpropagation network (BN).

Proposed technique: The first step in the proposed technique is to extract relevant features which unambiguously represent the pattern. The next step is to create a feature vector, using these features, that retains the spatial relationship among the features. The

feature components considered in this Letter are the corners and middle points of the line-approximated (preprocessed) input pattern. The features extracted are then encoded as explained below. For example, consider the pattern of Fig. 1a.

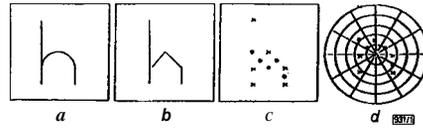


Fig. 1 Example of preprocessing for proposed technique

- a Original pattern (P1)
- b Result of line approximation (P2)
- c P3 where end points and middle points are marked
- d P3 overlaid on a radial grid

We process the pattern using the following steps:

- (i) Approximate the given pattern P1 of Fig. 1a by straight lines to obtain the pattern P2 (Fig. 1b).
- (ii) Extract only the end points of the line segments of P2, and if any two end points are connected in the original pattern, mark the middle point between those end points to obtain pattern P3 (Fig. 1c).
- (iii) Place the pattern P3 on the radial grid to obtain the feature vector *feat*.
- (iv) Blur *feat* by a 3 x 3 mask to obtain the input array *inp* to the BN. Blurring is employed to achieve better distortion tolerance in the recognition.

Fig. 1c shows the end points of the line segments marked 'x' (with value 1) and the middle points marked 'o' (with value 2), whereas Fig. 1d gives the radial grid on to which the pattern of Fig. 1c is superposed. The following is the pseudocode for obtaining input array *inp* from the pattern P3:

```

Find the centroid of the pattern P2.
Initialise inp = 0 for all indices.
Do
For all i, j
  if ( $P3[i][j] \neq 0$ ), it is either an end point of a line segment or a centre point
    find the angle  $\theta_i$ 
    find the normalised distance,  $\rho_i$ , from the centroid
    quantise the angle and the radial distance
     $feat[\theta_i][\rho_i] += P2[i][j]$  ( $0 \leq \theta_i < 12$ ) and ( $0 \leq \rho_i < 5$ ).
For all  $\theta, \rho_j$  *blurring*
  sum = 0.0
  for  $|l| \leq 1$ 
    for  $|m| \leq 1$ 
       $sum += feat[\theta_i + l][\rho_j + m] * mask[l][m]$ 
   $inp[\theta][\rho_j] = sum$ 

```

Using this algorithm, the input vectors *inp* have been obtained for 8000 patterns (500 variations of 16 types). Each of the 16 exemplar patterns (of size 57 x 57) is subjected to rotation and random independent magnifications in the x and y directions. The value of magnification ranges from 1 to 1.7, and the random rotation from -15 to +15°. These 8000 patterns were used to train a BN with the following specifications: input neurons = 60, hidden neurons = 25, output neurons = 4, learning rate < 0.009, and momentum constant < 0.0009.

Table 1: Details of training of BN

Iteration number	Error	Number of misses	Iteration number	Error	Number of misses
0	0.257020	7500	1000	0.001374	24
250	0.004822	109	1300	10.0011741	21
500	10.0024621	46	2000	10.0010201	21

Since the angular measure has been quantised to 12 levels, and the radial distance to five levels, the number of input neurons is 60. Further, since the number of classes (of patterns) is 16, the number of output neurons is chosen to be four. The results of training the network are given in Table 1. Some samples of the exemplars used to train the network are given in Fig. 2.

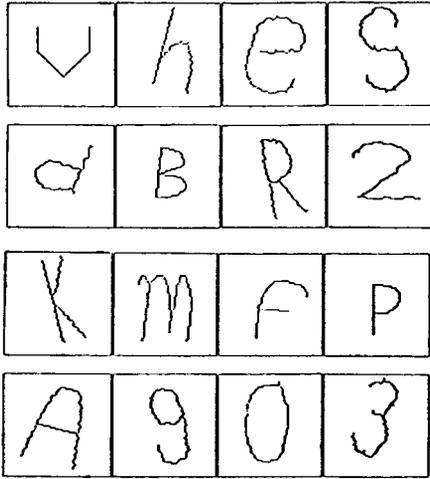


Fig. 2 Samples of test patterns from which input vector to neural network is calculated after preprocessing

Table 1 indicates that, after 2000 iterations of training, the network misclassifies only 21 of the 8000 patterns; a success rate of more than 99%, which is an indication of the efficiency of the feature extraction procedure in the proposed scheme. An additional set of 8000 patterns (with and without noise) is created with a different seed value for the random-number generator to test the proposed technique. The testing is carried out several times and the details are given in Table 2.

Performance of coding scheme with noisy patterns: Noise, introduced by flipping (with a specified probability) the pixels, results in breaks and isolated dots in the pattern. The line approximation removes the isolated dots by rejecting short line segments. However, the breaks caused give rise to many more line segments, and new end and middle points of the line segments, thereby causing much deterioration in the performance of the proposed technique. By further preprocessing the patterns using simple morphological techniques such as dilation followed by thinning, the breaks in the patterns can be minimised and the performance of the proposed technique vastly improved. Table 2 gives the performance results of the proposed technique compared to noiseless patterns, noisy patterns and preprocessed noisy patterns using morphological techniques.

Table 2 Percentages of correct classification by proposed technique for various cases

Types of patterns	Test data 1	Test data 2	Test data 3	Test data 4	Test data 5
noiseless	97.6	97.1	97.1	97.67	97.1
10% noise	68.0	67.5	67.3	68.3	68.2
20% noise	41.3	41.5	41.0	40.8	41.8
10% noise + morphology	92.7	92.6	92.8	93.0	92.7
20% noise + morphology	89.7	90.3	89.3	90.1	89.9

Conclusions: A new, efficient method for encoding patterns has been proposed to provide a feature vector as an input to an artificial neural network (ANN) for classification. Feature vector extraction involves approximation of the given pattern by straight lines, followed by the extraction of the corner points along with the centre points of the line segments. These features are then overlaid on a radial grid of cells to form the input array to the BN. The results obtained show the efficacy and robustness of the proposed technique, complete details of which will appear in [5].

References

- 1 FUKUSHIMA, K., and WAKE, N.: 'Hand written alphanumeric character recognition by the neocognitron', *IEEE Trans. Neural Netw.*, 1991, 2, pp. 355-365
- 2 YOU, S.D., and FORD, G.E.: 'Network model for invariant object recognition', *Pattern Recognit. Lett.*, 1994, 15, (8), pp. 761-768
- 3 PERANTONIS, S.J., and LISBOZ, P.J.G.: 'Translation, rotation, and scale invariant pattern recognition by high-order neural networks and moment classifiers', *IEEE Trans. Neural Netw.*, 1992, 3, (2), pp. 241-251
- 4 LeCUN, Y., BOSER, B., DENKER, J.S., HENDERSON, D., HOWARD, R.E., HUBBARD, W., and JACKEL, L.D.: 'Hand written digit recognition with a back-propagation network', in 'Advances in neural information processing systems 2' (Morgan Kaufmann, San Mateo, CA), pp. 396-404
- 5 GANESH MURTHY, C.N.S., and VENKATESH, Y.V.: 'A new method for encoding patterns for classification by neural networks'. submitted to *IEEE Trans. Neural Netw.*