



# Classification of handwritten characters by their symmetry features

DOI:  
[10.1109/ACT.2009.85](https://doi.org/10.1109/ACT.2009.85)

[Link to publication record in Manchester Research Explorer](#)

## Citation for published version (APA):

Holland, S., & Neville, R. (2009). Classification of handwritten characters by their symmetry features. In *ACT 2009 - International Conference on Advances in Computing, Control and Telecommunication Technologies|ACT - Int. Conf. Adv. Comput., Control Telecommun. Technol.* (pp. 316-318). IEEE. <https://doi.org/10.1109/ACT.2009.85>

## Published in:

ACT 2009 - International Conference on Advances in Computing, Control and Telecommunication Technologies|ACT - Int. Conf. Adv. Comput., Control Telecommun. Technol.

## Citing this paper

Please note that where the full-text provided on Manchester Research Explorer is the Author Accepted Manuscript or Proof version this may differ from the final Published version. If citing, it is advised that you check and use the publisher's definitive version.

## General rights

Copyright and moral rights for the publications made accessible in the Research Explorer are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

## Takedown policy

If you believe that this document breaches copyright please refer to the University of Manchester's Takedown Procedures [<http://man.ac.uk/04Y6Bo>] or contact [openresearch@manchester.ac.uk](mailto:openresearch@manchester.ac.uk) providing relevant details, so we can investigate your claim.



# Classification of Handwritten Characters by their Symmetry Features

Sam Holland and Richard Neville

School of Computer Science  
The University of Manchester  
Manchester, United Kingdom

sam.holland@postgrad.manchester.ac.uk, r.neville@manchester.ac.uk

**Abstract**—We propose a technique to classify characters by two different forms of their symmetry features. The Generalized Symmetry Transform is applied to digits from the USPS data set. These features are then used to train Probabilistic Neural Networks and their performances are compared.

**Keywords**—neural, networks, probabilistic, symmetry, classification, handwritten, characters

## I. INTRODUCTION

Symmetry is a property of many of the objects in the world around us, often a by-product of simplicity in design and manufacturing. Symmetry is not widely used to assist neural networks to perform classification. The performance of Optical Character Recognition (OCR) systems have not yet reached that of humans [2,3]. We believe that symmetry features used by Neural Networks will improve recognition.

The aim of this work is to test the hypotheses that symmetry features of patterns can be used for classification and that classification of a pattern's symmetry features can improve generalization.

To this end, we trained with a large set of handwritten characters collected from real world samples. We partitioned the set into two groups: a training set and a test set. The performance on the test set provides an insight into the net's ability to generalize – i.e. correctly classify new, unseen patterns which vary from those in the training set.

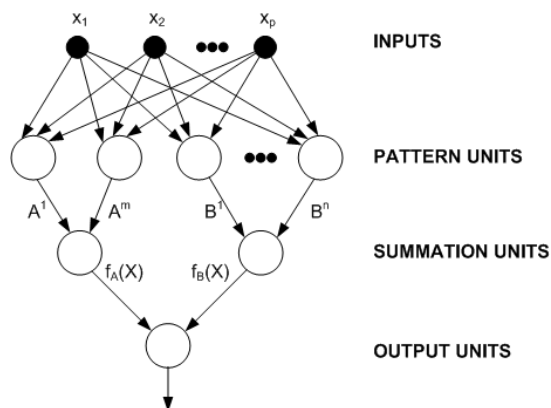


Figure 1. Structural overview of a Probabilistic Neural Network.

## II. PROBABILISTIC NEURAL NETWORKS

Probabilistic Neural Networks (PNNs) have been shown to outperform the traditional Multi-Layer Perceptron (MLP) (described in [4], used in [5]) classifier in Optical Character Recognition (OCR) tasks [9]. Their main advantage is that their training is trivial and near instantaneous, requiring only a single pass [1,7]. It is based on approximating probabilities through the use of Parzen windows. They can support very complex decision surfaces which approach the Bayes optimal [8]. The structure is depicted in Figure 1.

### A. Pattern Units

The pattern units are similar to a semi-linear neuron of an MLP except that it uses an exponential activation-output function:

$$g(Z_i) = e^{\frac{Z_i - 1}{\sigma^2}} \quad (1)$$

A smoothing parameter  $\sigma$ , identical for all the units in the layer, controls the exponential scale factor. The activation,  $Z_i$  is defined as the sum of the weighted ( $W$ ) inputs ( $X$ ):

$$Z_i = X \cdot W_i \quad (2)$$

For each training pattern there is one pattern unit. The training pattern input is used as the weight for that pattern unit. Whilst this layer will be large for any non-trivial classification tasks, training is completed within a single pass. Recognition is slower, but training is quicker than an MLP used for the same task.

After training, the output of these units represents how similar the new pattern is to the training patterns. The unit with the highest output represents the closest matching training pattern.

### B. Summation Units

The summation units simply sum the output of pattern units belonging to the same class:

$$f_A(X) = \sum_{i \in A} Z_i \quad (3)$$

### C. Output Units

The output units have two inputs and a binary output. The first input is from the summation unit of patterns belonging to that class. A summation unit of non-class members is inverted (multiplied by negative one). Hence, the activation for the output unit is:

$$a = f_A(X) - f_B(X) \quad (4)$$

The activation-output function is a hard limiter:

$$y = \begin{cases} 1 & \text{if } a \geq 0 \\ -1 & \text{if } a < 0 \end{cases} \quad (5)$$

### D. Smoothing Parameter, $\sigma$

The smoothing parameter,  $\sigma$  affects the network's Probability Distribution Function (PDF) estimate. As  $\sigma \rightarrow 0$ , the network acts as a nearest-neighbour classifier. As  $\sigma \rightarrow \infty$ , the network creates a non-linear decision boundary approaching the Bayes optimal. In practice, it is not difficult to find the optimal value for  $\sigma$  and the value can be modified after training of the weights is complete.

## III. THE GENERALIZED SYMMETRY TRANSFORM

The symmetry measure used in earlier work calculated the mean average location of symmetry in the horizontal and the vertical axes. In this work, we used the Generalized Symmetry Transform (GST) [6] because it uses symmetry information from all possible orientations. The GST also produces a detailed map of the symmetry magnitudes, as a continuous measure, rather than produce an average position.

The symmetry magnitude is calculated from an approximation of the pixel greyscale intensity gradients. We can approximate the gradient using the Sobel edge operator convolution masks:

$$\nabla p_k = \left( \frac{\delta}{\delta_x} p_k, \frac{\delta}{\delta_y} p_k \right) \quad (6)$$

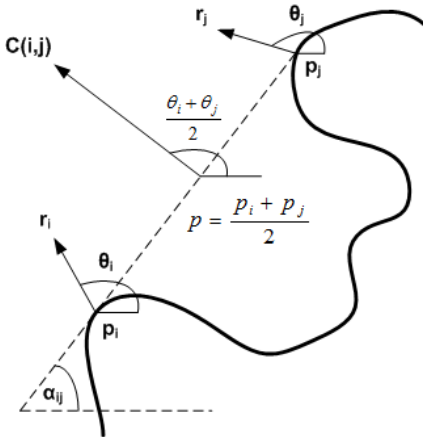


Figure 2. Two points,  $i$  and  $j$  contributing to the point mid-way between them.

A vector,  $v_k = (r_k, \theta_k)$  is associated with each pixel  $k$  in the input image consisting of the magnitude of the intensity gradient,  $r_k$ ; and the angle between the normal of the gradient and the horizontal axis,  $\theta_k$  (Figure 2).

$$r_k = \log(1 + \|\nabla p_k\|) \quad (7)$$

$$\theta_k = \arctan \left( \frac{\frac{\delta}{\delta_x} p_k}{\frac{\delta}{\delta_y} p_k} \right) \quad (8)$$

Each pair of pixels contributes to the symmetry magnitude of a pixel mid-way between them. Hence, the set  $\Gamma(p)$  consists of all the pixel-pairs that contribute to the magnitude at point  $p$ :

$$\Gamma(p) = \left\{ (i, j) \mid \frac{p_i + p_j}{2} = p \right\} \quad (9)$$

A Gaussian-based distance weight function favours contributions from pixel-pairs that are closer together than those further apart:

$$D_\mu(i, j) = \frac{1}{\sqrt{2\pi\mu}} e^{-\left(\frac{\|p_i + p_j\|}{2\mu}\right)^2} \quad (10)$$

The shape of the Gaussian is controlled by  $\mu$ . A phase weight function favours opposing gradient orientations from a pair of pixels than those which share an orientation:

$$P(i, j) = [1 - \cos(\theta_i + \theta_j - 2\alpha_{ij})]x[1 - \cos(\theta_i + \theta_j)] \quad (11)$$

Where  $\alpha_{ij}$  is the angle between a line connecting points  $i$  and  $j$ , and the horizontal axis. The total contribution of two pixel points a function of the distance and phase weight functions:

$$C(i, j) = D_\mu(i, j) P(i, j) r_i r_j \quad (12)$$

The symmetry magnitude at a given point  $p$  is the sum of the contributions over a set of pixel pairs,  $\Gamma(p)$ :

$$M_\mu(p) = \sum_{(i, j) \in \Gamma(p)} C(i, j) \quad (13)$$

The contribution has a direction, which is calculated from the two contributing gradient orientations:

$$\varphi(i, j) = \frac{(\theta_i + \theta_j)}{2} \quad (14)$$

The direction of the symmetry at a specific point  $p$  is  $\phi(p) = \varphi(i, j)$  such that  $C(i, j)$  is the maximum in the set  $\Gamma(p)$ .

#### IV. EXPERIMENT

The experiment used the United States Postal Service (USPS) data set. The set contains 9292 patterns each representing a handwritten digit (the numerals zero to nine inclusive). A 16x16 greyscale image describes each digit. We trained the networks with 7291 patterns. The networks' performance was measured by presenting 2001 previously unseen patterns.

One network trained with the original patterns without any additional processing. A second network trained on the symmetry magnitude map,  $M_\mu$  and a third trained on the symmetry orientation map,  $\phi$ .

We normalised the magnitude maps into the range -1 to 1. The GST represents no symmetry with a zero value, but these do not propagate well through weighted-sum connectionist systems.

We varied the Probabilistic Neural Networks' smoothing parameter and measured the effect on the classification performance to find the optimal performance.

#### V. RESULTS

The optimal performances were: 95.17% at  $\sigma = 0.5$  using the original data as input; 87.2% at  $\sigma = 0.1$  for the symmetry magnitude map; and 72.2% at  $\sigma = 0.7$  for the symmetry orientation map. The relationship between the smoothing operator and the recognition rate for all three networks is shown in Figure 3 and listed in Table 1.

Smoothing Parameter	Recognition Rate		
	Original	Magnitude Map	Orientation Map
0.1	53.26	<b>87.24</b>	28.7
0.2	92.72	86.85	56.1
0.3	95.02	82.06	79.02
0.4	95.12	73.24	79.17
0.5	<b>95.17</b>	64.52	79.17
0.6	95.17	57.65	79.12
0.7	95.12	50.92	<b>79.22</b>
0.8	95.02	45.89	79.22
0.9	94.97	41.9	79.02

Table 1. Recognition rates for the three networks for smoothing values in the range 0.1 to 0.9 inclusive.

#### VI. DISCUSSION

The network using the raw USPS data set outperforms the networks using the symmetry magnitude and orientation maps. The smoothing operator value indicates that all sets of data are better suited to nearest-neighbour classification, suggesting no clear separation of classes within the decision space. This in

turn suggests that the symmetry methods are no more useful for the generalization of the pattern classes for this particular domain.

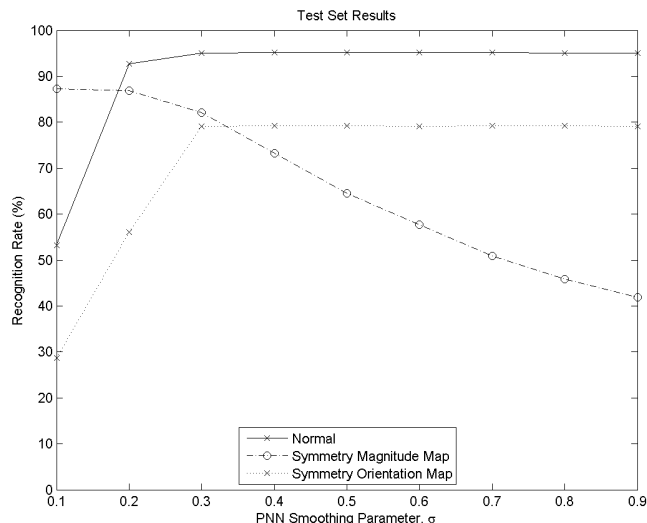


Figure 3. Graph showing the relationship between the PNN Smoothing Parameter and the performance of the three networks in terms of their recognition rate.

Only one form of symmetry was used in this experiment: reflectional (or mirror) symmetry. It is too early to discount symmetry's potential significance. Research into the use of rotational (or combinations of) symmetry should be conducted. Normalisation techniques applied to the magnitude map or orientation map prior to classification may improve performance.

#### REFERENCES

- [1] F. Ancona, A.M. Colla, S. Rovetta and R. Zunino, "Implementing probabilistic neural networks," *Neural Computing and Applications*, vol 5, pp. 152-159, 1997.
- [2] J.L. Blue, G.T. Candela, P.J. Grother, R. Chellappa and C.L. Wilson, "Evaluation of pattern classifiers for fingerprint and ORC applications," *Pattern Recognition*, vol 27 pp. 689-698, 1994.
- [3] C-L Liu and H. Fujisawa, "Classification and learning methods for character recognition: advances and remaining problems," *Studies in Computational Intelligence (SCI)*, vol 90, pp. 139-161, 2008.
- [4] G.E. Hinton, D.E. Rumelhart and R.J. Williams, "Learning internal representations by error propagation," in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, vol 1: Foundations, 1986.
- [5] Y. LeCun, I. Guyon, L.D. Jackel, D. Henderson, B. Boser, R.E. Howard, J.S. Denker, W. Hubbard and H.P. Graf, "Handwritten digit recognition: applications of neural network chips and automatic learning," *IEEE Communications Magazine*, vol 11 pp.41-64, 1989.
- [6] D. Reisfeld, H. Wolfson and Y. Yeshurun, "Context-free attentional operators: the generalized symmetry transform," *International Journal of Computer Vision*, vol 14(2) pp 119-130, 1995.
- [7] D.F. Specht, "Probabilistic Neural networks for classification, mapping or associative memories," *Proceedings, IEEE International Conference on Neural Networks*, vol 1 pp 525-532, 1988
- [8] D.F. Specht, "Probabilistic neural networks," *Neural Networks*, vol 3 pp. 109-118, 1990.
- [9] D.F. Specht, "Probabilistic neural networks and general regression neural networks," in *Fuzzy Logic and Neural Network Handbook*, pp. 3.1-3.44, Ed, C.H. Chen, McGraw-Hill Inc, New York, 1996

