The independent components of characters are `strokes'

Conference Paper · October 1999

DOI: 10.1109/ICDAR.1999.791812 · Source: IEEE Xplore

Kalpathi Ramakrishnan
Indian Institute of Science

The independent components of characters are ‘strokes’

S H Srinivasan
Institute for Robotics & Intelligent Systems
Bangalore - 560 001
shs@cair.res.in

K R Ramakrishnan
Department of Electrical Engineering
Indian Institute of Science
Bangalore - 560 001, INDIA
krr@ee.iisc.ernet.in

S Bhagavathy
Karnataka Regional Engineering College
Suratkal

Abstract

What are the natural features of hand-written characters and how to arrive at them automatically? We apply independent components analysis on hand-written characters. Independent components analysis extracts the underlying statistically independent signals from a mixure of them. We expect strokes to be the independent components of handwritten characters. Our findings show that stroke-like features emerge as a result of the analysis confirming the above intuition. This finding is significant since it gives an automatic procedures for extracting stroke-like features from multilingual character data sets. We use these features for handwritten digit recognition using a very simple classifier. The classifier is chosen to be simple so that the quality of input feature set can be evaluated. The recognition results indicate that the features arrived at by independent component analysis are useful.

1 Introduction

Handwritten character recognition poses a major challenge to researchers and practitioners in the field of automated handwritten document analysis and recognition. This is particularly so because different people have different ways of writing each character. There may be variations in size, thickness and style among samples of the same character. Handwritten character recognition has been approached using a variety of techniques. Some of these are: binary weighted scheme [7], frequency weighted scheme [8], multi-layer perceptron network [9], moment-based pattern classifiers [10], and hidden markov models [11] Methods using multiple classifiers [12] have been developed to improve robustness. In classical pattern recognition parlance, these are different *classifiers*. The *features* used for classification are different. In general the features used are carefully selected and optimized for particular classifiers. It should also be noted that the features used are also optimized for particular orthographic character sets.

This approach is useful when one is considering few orthographic character sets. In the context of Indian languages, this approach is of limited use since there are a large number of languages each with its own orthography. The characters also have a component structure in the sense that the graphic representation of characters is made of “strokes” and the strokes can stand for a short-hand of some other character. Thus a stroke-based analysis of the input is essential. A stroke-like feature set optimal for each language has to be arrived at. This is a tedius process. A more useful approach would learn the optimal features from examples of characters.

In this paper, we present a method of recognizing handwritten characters which makes use of independent component analysis. Independent component analysis is a signal processing approach in which a set of multidimensional measurement vectors are represented in a basis where the components are as statistically independent as possible. The redundancy reduction that results from such a process is similar to that achieved by the human visual system, and hence can be applied to practical problems concerning pattern recognition.

We have tested the usefulness of resulting features for character recognition. The results shown in this paper use the simplest possible classifier: nearest neighbor classifier. This is to show the optimality of the features learned. With more sophisticated classifiers and some amount of tuning, the classification accuracies are bound to improve.

2 Independent Component Analysis (ICA)

Let us denote a vector, which represents a phenomenon associated with n independent random variables, as $\boldsymbol{s} = [s_1\ s_2\ s_3\ s_4\ \cdots\ s_n]^T$. This is called the source vector, as the components are independent sources of information. Let $\boldsymbol{x}$ be an observed m dimensional random vector obtained when we subject $\boldsymbol{s}$ to a linear process,

$$\boldsymbol{x} = \boldsymbol{A}\boldsymbol{s} \tag{1}$$

Here matrix $\boldsymbol{A}$ is a $m \times n$ matrix referred to as the mixing matrix. The converse relation can be written as,

$$\boldsymbol{s} = \boldsymbol{W}\boldsymbol{x} \tag{2}$$

$\boldsymbol{W}$ is referred to as the separating matrix. We are interested in extracting the sources from the observed vector. The estimation of $\boldsymbol{s}$ and/or $\boldsymbol{A}$ and $\boldsymbol{W}$, based on the observed data, is called independent component analysis or ICA [1][3][5-6]. Independent component analysis basically involves two steps. These are,

Data Preprocessing: In this we center the data by removing its mean. We also remove the correlations between the components of the data. This process is called whitening.

Extraction of independent components: This is done using one among the various algorithms available. We have chosen a fast fixed-point algorithm by Hyvarinen and Oja [1], because it is simple, fast and can be implemented by readily available software.

3 Data preprocessing

Before we apply the algorithm to compute the independent components, we subject the given data to preprocessing. Let $\boldsymbol{E}$ denote the matrix of eigenvectors of the covariance matrix $cov\{\boldsymbol{x}\}$, and $\boldsymbol{D} = diag\{\xi_1, \cdots, \xi_m\}$, a diagonal matrix of corresponding eigenvalues. The whitened data vector corresponding to the observed vector $\boldsymbol{x}$ is given by,

$$\boldsymbol{v} = \boldsymbol{D}^{-1/2}\boldsymbol{E}^T\boldsymbol{x}$$

The matrix, $\boldsymbol{V} = \boldsymbol{D}^{-1/2}\boldsymbol{E}^T$ is called the whitening matrix. Using relation (1), we have,

$$\boldsymbol{v} = \boldsymbol{V}\boldsymbol{x} = \boldsymbol{V}\boldsymbol{A}\boldsymbol{s} = \boldsymbol{B}\boldsymbol{s}$$

Matrix $\boldsymbol{B}$ is called the whitened mixing matrix.

4 Extraction of the independent components

The image data is first converted to a vector using the row major representation. Each image is now a vector and the collection of input images now becomes a matrix. To extract the independent components of the data matrix $\boldsymbol{x}$, we use the a fixed-point algorithm [1]. This main idea of this algorithm is the optimization of a contrast function, such as the kurtosis [4]. Kurtosis of a random variable y is defined as

$$kurt(y) = E\{y^4\} - 3(E\{y^2\})^2$$

Consider the optimization of the kurtosis of projection of a zero mean whitened random variable $\boldsymbol{v}$ onto vector $\boldsymbol{w}$

$$f(\boldsymbol{w}) = E\{(\boldsymbol{w}^T\boldsymbol{v})^4\} - 3||\boldsymbol{w}||^4$$

with constraint

$$h(\boldsymbol{w}) = ||\boldsymbol{w}||^2 - 1 = 0$$

Necessary conditions for an optimum are given by the method of Lagrange multipliers[2]

$$4E\{(\boldsymbol{w}^T\boldsymbol{v})^3\boldsymbol{v}\} - 12||\boldsymbol{w}||^2\boldsymbol{w} + 2\lambda'\boldsymbol{w} = 0$$

Setting $\lambda = -\lambda'/2$ and noting that $||\boldsymbol{w}||^2 = 1$,

$$\lambda\boldsymbol{w} = E\{(\boldsymbol{w}^T\boldsymbol{v})^3\boldsymbol{v}\} - 3\boldsymbol{w} \tag{3}$$

Equation (3) shows that at an optimum $\boldsymbol{w}$ the right hand side of the equation is parallel with $\boldsymbol{w}$, or that the direction of $\boldsymbol{w}$ remains fixed under the iteration

$$\boldsymbol{w}(k+1) = E\{(\boldsymbol{w}(k)^T\boldsymbol{v})^3\boldsymbol{v}\} - 3\boldsymbol{w}(k)$$

It has been shown [1] that the computation in the algorithm converges to the desired optima if we start at any random initial point.

5 ICA of characters

Independent component analysis was performed on a set of 60 images of handwritten Devanagari characters. Each image is resized to 32 × 32 pixels and then arranged into one row. Thus each row of the input matrix is one image. This data matrix is subjected to ICA using the fixed-point algorithm that was previously discussed. Figure 1 shows the data from which the independent components have to be extracted. Figure 2 shows the 20 most significant independent components of the set of images. By "most significant", we mean the independent components corresponding to the 20 highest eigenvalues in the covariance matrix of the data matrix. Figure 3 is obtained after Figure 2 has been subjected to a thinning process. This has been done sharpen the features of the independent components. Now, we are able to observe clearly that the independent components resemble "strokes" constituting the individual characters. We can interpret the independent components of a set of characters as a set of strokes, like those made by a pen. Each character is then a linear combination of these strokes with each stroke having a corresponding weight. It is found that the images can be reconstructed with negligible distortion using a much smaller number of independent components than the number of images used as input data. This means that these more significant independent components correspond to those strokes which are more pronounced in the characters. These are the strokes which play the major role in the formation of the individual characters. There is no significant distortion as a result of ignoring the less significant independent components corresponding to the less pronounced strokes.

5.1 Recognition methodology

Based on the conclusion that the independent components of a set of characters are strokes constituting the individual characters, we can arrive at a useful method of handwritten character recognition. Initially, a set of images of the class of handwritten characters to be recognized are taken. These constitute the training set of data. Each image is resized to, say, 32 × 32 pixels and then arranged into one row. The training data is realized by arranging these rows into a matrix – one after the other. This training data, comprising the set of character images, is then subjected to ICA and a number of significant independent components are extracted. These independent components, as explained above, are the significant strokes which form the individual characters by combining linearly. Now, each row of the mixing matrix $\boldsymbol{A}$ of relation (1) contains the weight of each stroke in forming the character corresponding to that row in the training matrix.

The image to be recognized, called the test image, is then taken, resized to the same size as that of the training images, and arranged into one row. Let this test vector be referred to as $\boldsymbol{y}$. We then determine the weight of each independent component or stroke in forming this test image. Let $\boldsymbol{s} = [s_1 \quad s_2 \quad \cdots \quad s_m]^T$ be the matrix containing the independent components and $\omega = [\omega_1 \quad \omega_2 \quad \cdots \quad \omega_m]$ be the weight vector. The weight vector ω is found by solving the equation

$$\boldsymbol{y} = \omega \boldsymbol{s} \tag{4}$$

which can be expanded as $\boldsymbol{y} = \omega_1 \boldsymbol{s}_1 + \omega_2 \boldsymbol{s}_2 + \cdots + \omega_m \boldsymbol{s}_m$. We then determine the row of the mixing matrix $\boldsymbol{A}$ to which the weight vector ω lies closest to.

6 Experiments with handwritten digit recognition

The initial experiment was performed on a training set consisting of 20 different handwritten samples for each digit, giving a total of 200 images. These images were resized to 32 × 32 pixels and arranged as rows of the training matrix in sequence, i.e. the samples of the digit 1 form the first twenty rows, the samples of digit 2 form the next twenty, etc. These training data then subjected to ICA. The 20 most significant independent components are extracted and the corresponding truncated mixing matrix is obtained. This entries in this matrix corresponding to each digit are averaged resulting in 10 templates, one for each digit. For a given test input, we calculate the mixture coefficients and find the closest template.

Recognition tests were performed by varying the number of training samples per digit (20 or 40) and number of indepdendent components retained (20 or 40), with and without thinning the input data.

7 Results and Discussion

The percentage of recognition of each digit was documented for each case mentioned in the previous section. The results are tabulated in Table 1. These percentages were recorded by conducting recognition tests on a large number of test samples of each digit. The training and test data have been obtained from a database of handwritten matter distributed by the Center of Excellence for Document Analysis and Recognition (CEDAR) at the State University of New York at Buffalo. The percentages of non-zero eigenvalues of the covariance matrix retained for computing the independent components have been mentioned for each case in parentheses. With a training set consisting of only 40 samples per digit, an average recognition rate of 87.2% was recorded, which is comparable to the success rates of other available methods [12]. The effect of thinning is depdends on the character. Larger data sets improve recognition accuracy and larger number of independent compondents generally improves recognition performance.

Thus the merits of ICA for handwritten character analysis are two fold:

1. The features learned do reflect the underlying orthographic structure.

2. Recognition using a simple classifier compares favorably with other approaches.

Extensions for robust recognition of multiple Indian language character sets are being worked out.

8 References

[1] HYVARINEN, A., and OJA, E. A fast fixed-point algorithm for independent component analysis. Tech. Rep. A35, Helsinki University of Technology, Laboratory of Computer and Information Science, 1996.

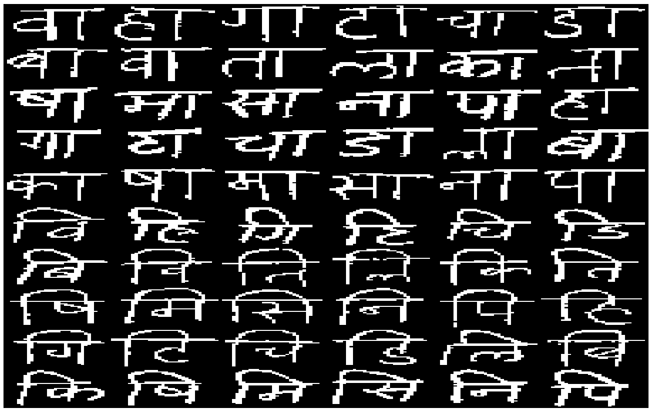

Figure 1. The images of 60 Devanagari characters on which ICA is performed.

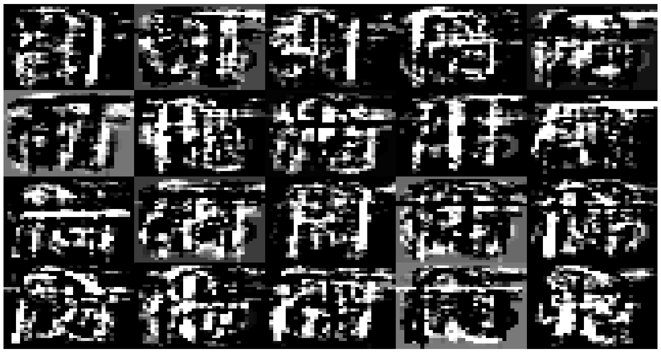

Figure 2. The 20 most significant independent components of the set of font images shown in previous figure.

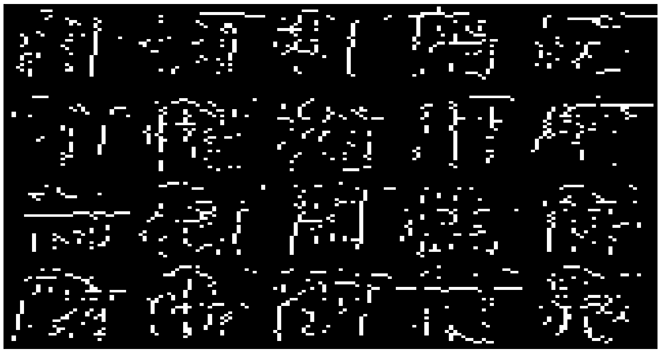

Figure 3. The previous figure after thinning. Stroke-like patterns are clearly visible

[2] LUENBERGER, D. G. Optimization by Vector Space Methods. Series in decision and control. John Wiley and Sons, New York, 1969.

[3] HYVARINEN, A., OJA, E., HOYER, P. and HURRI, J. Image feature extraction by sparse coding and independent component analysis. Helsinki University of Technology, Laboratory of Computer and Information Science.

[4] HYVARINEN, A. Independent component analysis by minimization of mutual information. Helsinki University of Technology, Laboratory of Computer and Information Science, 1997.

[5] BELL, A. J., SEJNOWSKI, T. J. The independent components of natural images are edge filters. Vision Research, 1997.

[6] HURRI, J. Independent component analysis of image data. Masters Thesis. Helsinki University of Technology, Department of Computer Science and Engineering, 1997.

[7] FAIRHURST, M. C. and STONHAM, T. J. A classification system for alphanumeric characters based on learning network techniques. Digital Processes 2, 1976.

[8] FAIRHURST, M. C., and MATTASO MALA, M. A. G. Performance comparison in hierarchical architectures for memory network pattern classifiers. Pattern Recognition Letters 4(2), 1986.

[9] RUMELHART, D. E., HINTON, G. E., and WILLIAMS, R. J. Learning internal representations by error propagation. Parallel Distributed Computing, Vol. 1., 1986.

[10] REISS, T. H. Recognizing Planer Objects using Invariant Image Features. Springer, Berlin, 1993.

[11] KIM, W. S. and PARK, R. H. Offline recognition of handwritten Korean and alphanumeric characters using Hidden Markov Models. Pattern Recogniiton 29(5), 1996.

[12] RAHMAN, A. F. R., and FAIRHURST, M. C. An evaluation of multiple -expert configurations for the recognition of handwritten numerals. Pattern Recognition, Vol. 31, No. 9, 1998.

Table 1. The results are reported as percentages of correctly recognized digits. The values in parentheses are percentages of non-zero eigenvalues retained for that case.

Digit	20 samples / digit				40 samples / digit		Overall best %
	Normal		Thinned		Normal		
	20 ICs (62.5%)	40 ICs (76.7%)	20 ICs (40.4%)	40 ICs (56.7%)	40 ICs (72.5%)	20 ICs (48.3%)	
0	92.6	95.8	96.8	98.9	97.9	100	100
1	93.5	94.6	98.9	98.9	93.5	100	100
2	85.0	86.2	77.5	72.5	86.2	67.5	86.2
3	84.6	86.8	87.9	91.2	90.1	82.4	91.2
4	94.3	98.9	51.1	60.2	90.9	68.2	98.9
5	75.6	72.2	38.9	51.1	74.4	56.7	75.6
6	90.5	98.8	73.8	75.0	91.7	72.6	98.8
7	68.8	69.9	68.8	75.3	95.7	84.9	95.7
8	69.7	68.5	70.8	73.0	80.1	74.2	80.1
9	61.1	65.6	37.8	57.8	70.0	56.7	70.0
Av	81.5	83.6	70.4	75.7	87.2	76.7	