# Signature Recognition and Verification with ANN


Cemil OZ
ozc@umr.edu
Sakarya University Computer Eng. Department, Sakarya, Turkey

Fikret Ercal
**ercal@umr.edu**
UMR Computer Science Department, Rolla, MO 65401

Zafer Demir
zdemir@sakarya.edu.tr
Sakaraya University electric electronic eng. Department sakarya , Turkey



## Abstract

**In this paper, we present an off-line signature recognition and verification system which is based on moment invariant method and ANN. Two separate neural networks are designed; one for signature recognition, and another for verification (i.e. for detecting forgery). Both networks use a four-step process. First step is to separate the signature from its background. Second step performs normalization and digitization of the original signature. Moment invariant vectors are obtained in the third step. And the last step implements signature recognition and verification.**



**Keywords**: Signature verification, signature recognition, ANN, moment invariants, computer vision


## 1. Introduction

Signature is a special case of handwriting which includes special characters and flourishes. Many signatures can be unreadable. They are a kind of artistic handwriting objects. However, a signature can be handled as an image, and hence, it can be recognized using computer vision and artificial neural network techniques.

Signature recognition and verification involves two separate but strongly related tasks: one of them is identification of the signature owner, and the other is the decision about whether the signature is genuine or forged. Also, depending on the need, signature recognition and verification problem is put into two major classes: (i) online signature recognition and verification systems (SRVS) and (ii) offline SRVS. Online SRVS requires some special peripheral units for measuring hand speed and pressure on the human hand when it creates the signature. On the other hand, almost all off-line SRVS systems relies on image processing and feature extraction techniques.

In the last two decades, in parallel with the advancement in the sensor technology, some successful online SRVS are developed [1, 2, 3]. There are also many studies in the area of offline SRVS category [4, 5, 6, 7, 8, 9, and 10]. These studies are generally based on ANN [4,10], analysis of the geometry and topology of the signature [11], and its statistical properties [9].

In this study, we present an off-line signature recognition and verification system which is based on a moment invariant method. Two neural networks are designed; one for signature recognition, and another for verification (i.e. for detecting forgery).

## 2. Image processing

The camera-captured or scanned real world images containing human signatures are processed using several image processing algorithms before the calculation of the moment invariants. These processes are given below.

### 2.1 Converting Color image to gray scale image

In present technology, almost all image capturing and scanning devices use color. Therefore, we also used a color scanning device to scan signature images. A color image consists of a coordinate matrix and three color matrices. Coordinate matrix contains x,y coordinate values of the image. The color matrices are labeled as red (R), green (G), and blue (B). Techniques presented in this study are based on grey scale images, and therefore, scanned or captured color images are initially coverted to grey scale using the following equation [12]:

$$\text{Gray color} = 0.299*\text{Red} + 0.5876*\text{Green} +0.114*\text{Blue} \quad (1)$$

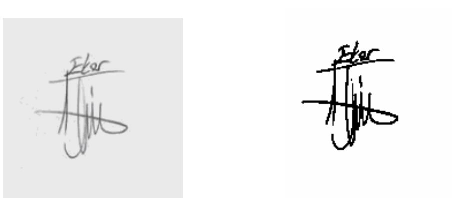
### 2.2 Noise Reduction

Noise reduction (also called “smoothing” or “noise filtering”) is one of the most important processes in image processing. Images are often corrupted due to positive and negative impulses stemming from decoding errors or noisy channels. An image may also be degraded because of the undesirable effects due to illumination and other objects in the environment. Median filter is widely used for smoothing and restoring images corrupted by noise. It is a non-linear process useful especially in reducing impulsive or salt-and-pepper type noise. In a median filter, a window slides over the image, and for each positioning of the window, the median intensity of the pixels inside it determines the intensity of the pixel located in the middle of the window. Different from linear filters such as the mean filter, median filter has attractive properties for suppressing impulse noise while preserving edges. Median Filter is used in this study due to its edge preserving feature [13, 14, 15,16].

### 2.3 Background elimination and border clearing

Many image processing applications require the differentiation of objects from the image background. *Thresholding* is the most trivial and easily applicable method for this purpose. It is widely used in image segmentation [17, 18]. Thresholding is choosing a threshold value T and assigning 0 to the pixels with values smaller than or equal to T and 1 to those with values greater than T. We used thresholding technique for differentiating the signature pixels from the background pixels. Clearly, in this application, we are interested in dark objects on a light background, and therefore, a threshold value T, called the brightness threshold, is appropriately chosen and applied to image pixels f(x, y) as in the following:

If $f(x,y) \geq T$ then
$f(x,y)$ = Background
else $f(x,y)$ = Object (2)



a) Captured signature b) signature image with background removed

**Figure 1.**

Signature image which is located by separating it from complex background image is converted into binary image whit background taking the pixel value of 1. Vertical and horizontal (histogram) projections are used for border clearing. For both direction, vertical and horizontal, we counted every row zeros and the resulting histogram is plotted sideways.

### 2.5 Signature normalization

Signature dimensions may vary due to the irregularities in the image scanning and capturing process. Furthermore, height and width of signatures vary from person to person and, sometimes, even the same person may use different size signatures. First, we need to eliminate the size differences and obtain a standard signature size for all signatures. After this normalization process, all signatures will have the same dimensions. In this study, we used a normalized size of 40x40 pixels for all signatures that will be processed further. During the normalization process, the aspect ratio between width and height of a signature is kept intact. Normalization process made use of the following equations:

$$x_i = \frac{x'_i - x_{\min}}{x_{\max} - x_{\min}} * M \qquad (3)$$

$$y_i = \frac{y'_i - y_{\min}}{y_{\max} - y_{\min}} * M \qquad (4)$$

In these equations:

$x_i, y_i$ : pixel coordinates for the normalized signature,
$x'_i, y'_i$ : pixel coordinates for the original signature,
M : one of the dimensions (width or height) for the normalized signature

The normalization process is demonstrated in the following figure.

a) Original signature image b) Normalized signature

**Figure 2. Signature normalization**

## 3. Moment invariant method

Moment invariants are properties of connected regions in binary images that are invariant to translation, rotation and scaling. They can be easily calculated from region properties and they are very useful in performing shape classification and part recognition. One of the techniques for generating invariants in terms of algebraic moment was originally proposed by Hu [19]. The algebraic moment of the characteristic function f(x,y) is defined to be:

$$m_{pq} = \int_x \int_y x^p y^q f(x,y) d_y d_x \qquad (5)$$

This can be approximated in discrete form by:

$$m_{pq} = \sum_x \sum_y x^p y^q f(x,y) \qquad (6)$$

A geometric figure can be uniquely determined by its algebraic moment. Therefore, instead of looking for invariants of moments, only invariants of low order moments are used in practical applications. Moment invariants are usually specified in terms of centralized moment. Here, the moment is measured with respect to the "center of mass", (x', y'). The central moment, μ, with respect to the centroid, and the normalized central moment, η, are calculated as:

$$m_{pq} = \sum_x \sum_y (x - x')^p (y - y')^q a_{xy} \qquad (7)$$

$$\eta_{pq} = \frac{\mu_{pq}}{(\mu_{00})^{\lambda}} \qquad (8)$$

where, $\lambda = \frac{(p+q)}{2} + 1, (p+q) > 2$ [20,21,22, 23].

The moment invariants used in our research are computed using the equations given in Table-1(b) for all signatures at various angles.

**Table 1 : (a) Formulas used for specific central moments b) List of the derived invariant moments**

| **Central Moments** | **Derived Invariant Moments** |
|---|---|
| $\mu_{00}=m_{00}$ | $I_1=\eta_{20} + \eta_{02}$ |
| $\mu_{10}=0$ | $I_2=(\eta_{20} - \eta_{02})^2+4\eta_{11}^2$ |
| $\mu_{01}=0$ | $I_3=(\eta_{30} - 3\eta_{12})^2+ (3\eta_{21}-\eta_{03})^2$ |
| $\mu_{20}= m_{20}-x' m_{00}$ | $I_4=(\eta_{30} + \eta_{12})^2+ (\eta_{21}-\eta_{03})^2$ |
| $\mu_{02}= m_{02}- y' m_{01}$ | $I_5=(\eta_{30} - 3\eta_{12}) (\eta_{30} + \eta_{12})^2( (\eta_{30} + \eta_{12})^2 -3(\eta_{21} + \eta_{03})^2) + (3\eta_{21} - \eta_{03}) (\eta_{21} + \eta_{03})(3 (\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2)$ |
| $\mu_{11}= m_{11}- y' m_{10}$ | $I_6=(\eta_{20} - \eta_{02}) ((\eta_{30} + \eta_{12})^2- (\eta_{21} + \eta_{03})^2 )+ 4\eta_{11} ( \eta_{30} + \eta_{21} )(\eta_{21}+ \eta_{03})$ |
| $\mu_{30}= m_{30}-3x' m_{20} + 2x' m_{10}$ | $I_7=(3\eta_{12} - \eta_{30}) (\eta_{30} + \eta_{12})( (3\eta_{30} \eta_{12}) - 3(\eta_{21} + \eta_{03})^2) + (3\eta_{21}- \eta_{03}) (\eta_{21} + \eta_{03})(( 3\eta_{30} \eta_{12})^2 -(\eta_{21} + \eta_{03})^2)$ |

## 4. Digitization of signatures

## Calculating moment invariants:

Feature vectors are generated using moment invariants. For this purpose, we use six different signature images which are sign different time. Then, we produced six different sets of feature vectors for every signature where each set consisted of seven moment invariant values listed in Table 1(b). A sample feature vector is shown in Figure 3.

5.18647933247972E-0001
1.72351384466328E-0001
1.32449449923652E-0001
2.13629276301316E-0001
1.45538602648840E-0002
8.86884674185101E-0002
5.61213979201577E-0003

**Global properties:** seven global features are used for better results. These features are signature height-to width ratio, maximum vertical projection, maximum horizontal projection, image area, vertical center of signature, vertical projection peaks and horizontal projection peaks.

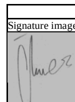
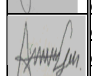
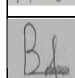
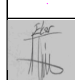
## 4. ANN design for signature recognition and verification

We designed a multilayer feed forward artificial neural network for recognition of off-line digitized signatures. The proposed ANN consists of 14 input variables, 18 hidden neurons, and 30 output variables and it is designed to recognize one signature at a time. Backpropagation algorithm is used for training.

### 4.1. Training for signature recognition

First, an input/output database is created manually for training and testing the ANN for six signature image which are belong to same person but signed different time. Each input vector consists of seven moment invariants obtained for a signature. As explained earlier in section 4, six different moment invariant vectors are produced for each signature. These six vectors are divided into two sets each containing three vectors. One of these sets (3 input vectors) is used in the training of ANN and the other set (remaining 3 input vectors) is used for testing. Additional, We also produce seven extra properties for a signature. As also explained earlier in section 4. The database contained a total of 30 different signature images which are used for both training and testing. Since 3 input vectors for each image is used for training purposes, there are a total of 90 (30*3) input vectors (data sets) in the training set. The remaining 90 data sets are used for testing. ANN contained 30 binary output values each corresponding to one signature being tested as shown in Table 2. Under normal (correct) operation of ANN, only one output is expected to take a value of "1" indicating the recognition of a signature represented by that particular output. The other output values must remain zero. In general, the number of outputs must be equal to the number signatures being considered. Table 2 shows a number of real input/output vectors used in the training set.which are obtained from a set of signatures.

**Table 2. A sample input set for signature recognition NN.**

| | Inputs(moments) | | | | Inputs(global features) | | | Outputs | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Signature images | $I_1$ | $I_2$ | ... | $I_7$ | $I_8$ | ... | $I_{14}$ | $O_1$ | $O_2$ | $O_3$ | $O_4$ | | $O_{30}$ |
|  | 0.24498 | 0.01320 | | 0.01508 | 0.9201 | | 7 | 1 | 0 | 0 | 0 | | 0 |
| | 0.21825 | 0.06348 | | 0.24186 | 0.9211 | | 7 | 1 | 0 | 0 | 0 | | 0 |
| | 0.76491 | 0.12616 | | 0.99508 | 0.9307 | | 7 | 1 | 0 | 0 | 0 | | 0 |
|  | 0.51865 | 0.17235 | | 0.21363 | 0.7413 | | 10 | 0 | 1 | 0 | 0 | | 0 |
| | 0.49462 | 0.02351 | | 0.08364 | 0.7371 | | 10 | 0 | 1 | 0 | 0 | | 0 |
| | 0.68945 | 0.09235 | | 0.20112 | 0.7382 | | 10 | 0 | 1 | 0 | 0 | | 0 |
|  | 0.56872 | 0.16345 | | 0.22116 | 0.8293 | | 7 | 0 | 0 | 1 | 0 | | 0 |
| | 0.56872 | 0.16345 | | 0.12125 | 0.8421 | | 7 | 0 | 0 | 1 | 0 | | 0 |
| | 0.39925 | 0.10635 | | 0.02115 | 0.8304 | | 7 | 0 | 0 | 1 | 0 | | 0 |
| . . . | . | . | | | | | | . . . | . . . | . . . | | | |
|  | 0.22156 | 0.00396 | | 0.00842 | 1 | | 11 | 0 | 0 | 0 | 0 | | 1 |
| | 0.37143 | 0.00296 | | 0.00242 | 1 | | 11 | 0 | 0 | 0 | 0 | | 1 |
| | 0.21433 | 0.01591 | | 0.00359 | 1 | | 11 | 0 | 0 | 0 | 0 | | 1 |

### 4.2. Signature verification

In this part of the study, our purpose is to authenticate a signature i.e. to verify that the signature is not counterfeit and it really belongs to the person who is claimed to be the owner of the signature. The ANN used for this purpose is also a multilayer feed forward network which consists of 14 input variables, 10 hidden neurons, and 2 output variables indicating whether the signature is fake or true.

Backpropagation algorithm is used for training. The training data set is obtained from three original (authenticated) signatures provided by the real owner and three fake signatures. As it was done for the preparation of the training data for the ANN used in recognition, three invariant vectors per signature is used in the training set. Therefore, a total of 18 moment invariant vectors are used in the training set. A sample set of three signatures belonging to the same person is shown in Figure 4.

**Figure 4. Tree signatures which belong to the true owner of the signature**

## 5. Implementation and test results

### 5.1. Signature recognition

The program used a windows interface as shown in Figure 5. This software allowed the signature images to be loaded one at a time and used in training and testing. First, the signature image is captured using a CCD camera or a scanner, then, through several image processing operations, it is converted to binary and normalized to a 40*40 image as explained earlier. Moment invariant and additional values are obtained from the normalized image which is then used as the input vector to the ANN. After the training of the ANN for signature recognition, the system is ready to recognize a given signature.

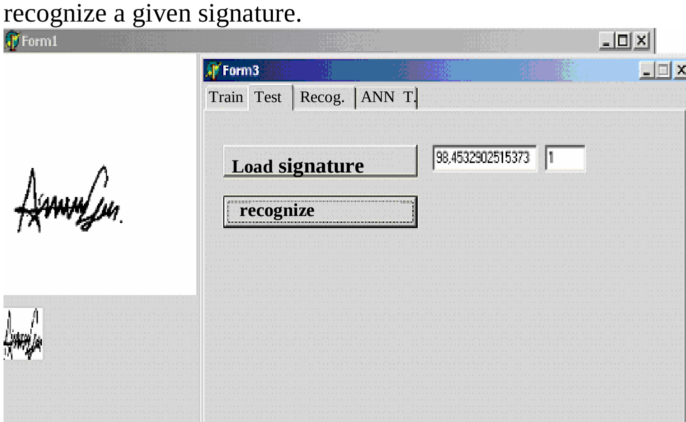



**Figure 5. Software interface for the signature recognition and verification system**

The signature recognition system is tested using 30 signatures chosen at random. The images were obtained using the following properties:

- Signatures were signed inside a special framed area.
- Images were taken with a simple CCD camera and they were shot from a fixed distance.

As explained in Section 4.1., 30 images in our database belonging to 30 different signatures are used for both training and testing. Since 3 (out of 6) input vectors for each image were used for training purposes, there are only 90 (30*3) input vectors (data sets) left to be used for the test set. Under normal (correct) operation of the ANN, only one output is expected to take a value of “1” indicating the recognition of a signature represented by that particular output. The other output values must remain zero. The output layer used a logic decoder which mapped neuron outputs between 0.5-1 to a binary value of 1. If the real value of an output is less than 0.5, it is represented by a “0” value. The ANN program recognized all of the 30 signatures correctly. This result translates into a 100% recognition rate. We also tested the system with 10 random signatures which are not contained in the original database. Only two of these signatures which are very similar to at least one of the 30 stored images resulted in “false positives” (output > 0.5) while the remaining 8 are recognized correctly as not belonging to the original set (the output value was <= 0.5). Since recognition step is always followed by the verification step, these kinds of false positives can be easily caught by our verification system. In other words, the verification step serves as a safeguard against “false positives” as well as “false negatives”.

### 5.2. Testing the verification system

Training for verification is explained in Section 4.2. Signatures used for testing the verification system are obtained the same way as in the recognition system. We tested the verification software using 10 signatures; 5 imitations (counterfeit signatures) and 5 true signatures. The program detected (classified) 4 true signatures and 5 counterfeits correctly. In other words, all counterfeit signatures are detected correctly. Only one signature is classified as a counterfeit while it was not (i.e. a “false negative”). Obviously, a “false negative” should be more acceptable in comparison to a “false positive”, because the person can always be given a second chance to prove that the signature is his/hers. On the other hand, a false positive in verification carries a lot of risk.

## 6. Conclusion

In this study, we presented an off-line signature recognition and verification system which is based on image processing, moment invariants, some global properties and ANNs. Both systems used a four-step process. In the first step, the signature is separated from its image background. Second step performs normalization and digitization of the original signature. Moment invariants and some global properties which are used as input features for the NN are obtained in the third step. Two separate ANNs are used; one for signature recognition and another for verification. Our recognition system exhibited a 100% success rate by identifying correctly all of the 30 signatures that it was trained for. However, it exhibited poor performance when it was presented with signatures that it was not trained for earlier. We did not consider this as a "high risk" case, because recognition step is always followed by the verification step and these kinds of false positives can be easily caught by the verification system. Indeed, the verification system did not miss any of the counterfeit signatures. However, its verification for true signatures lacked some accuracy. We think that this is also acceptable because a person can always be given a second chance to prove the ownership of a signature.

Generally, the failure to recognize/verify a signature was due to poor image quality and high similarity between two signatures. Recognition and verification ability of the system can be increased by using additional features in the input data set.

## References


1- Parizeu, M., Plamondon, R., " A comparative analysis of regional correlating dynamic time warping, and skeletal tree matching for signature verification", IEEE transactions on Pattern Analysis and Machine Intelligence 12, pp. 710-717, 1990.
2- Brault, J. Plamondon, R., "segmenting handwritten signatures at their perceptually important points." IEEE transactions on Pattern Analysis and Machine Intelligence 15, pp. 953-957, 1993.
3- Lee, L., Berger, T., Aviczer, e., "Reliable on-line human signature verification systems." IEEE transactions on Pattern Analysis and Machine Intelligence 18, pp. 643-647, 1996.
4- Xuhang, X., Graham, L., "Signature verification using a modified basian network", Pergamon Pattern Recognition 35, pp.983-995, 2002.
5- Yuan Y., T., Ernest C.M.L., "New method for feature extraction based on fractal behavior." Pergamon Pattern Recognition 35, pp.1071-1081, 2002.
6- Ismail M.A., Samia G., "Off-line Arabic signature recognition and verification", Pergoman Pattern Recognition 33, pp.1727-1740, 2000.
7- Qi, Y., Hunt, B.R., "signature verification using global and grid features." Pattern recognition 27, pp.1621-1629, 1994.
8- Yedekcioglu O.A., Akban M.B., Lim, Y.H., "off-line signature verification with thickened templates." COMCON5 proceedings of 5 th international conference on Advanced in communication and control, crete, Greece, pp. 131-142, 1995.
9- Han, K., Sethi I.K., "Handwritten signature retrieval and identification." Pattern Recognition Letters 17, pp. 83-90., 1996.
10- Baltzakis H., Papamorkos N., "A new signature verification technique based on a two-stage neurol network classifier.", PergomanEngineering Aplication of Intelligence 14, pp.95-103, 2001.
11- Droughord, J., Plamondon, R., Godbout, m., "A neural network approach to off-line signature verification using directional PDF.", Pattern Recognition 29, 415-424, 1996.
12- Luong, C. M., "Introduction to Computer Vision and Image Processing", web site: http://www.netnam.vn/unescocourse/computervision/comp_frm.htm
13- Lim, J.S., "Two-Dimensional and Image Processing", Prentice-Hall, 1990.
14- Yang, X., and Toh, P.S., " Adaptive Fuzzy Multilevel Median Filter", IEEE Transaction on Image Processing, Vol. 4, No. 5, pp.680-682, may 1995.
15- Hwang, H., and Haddad, R.A. "Adaptive Median Filters: new Algorithm and Results", Transactions on Image processing, Vol. 4, No. 4 pp.449-505, April 1995.
16- Rosenfeld, A., "Digital Picture Processing", Academic Press Inc., 1982.
17- Erdem, U.M., " 2D Object Recognition In Manufacturing Environment Using Implicit Polynomials and Algebraic Invariants", Master Thesis, Bogazici University, 1997.
18- Fu, K.S., Mui, J.K., "A survey On Image Segmentation", Pattern Recognition, Vol. 13, pp.3-16, Pergoman Press, 1981.
19- Hu, M., "Visual Pattern Recognition by Moment Invariants", IRE Trans. Inform. Theory 8, pp.179-187, 1962.
20- Koker, R., Oz, C., Ferikoglu, A., " Object Recognition Based on Moment Invariants Using Artificial Neural Networks", Proceedings of 3rd International Symposium an Intelligent Manufacturing Systems, August 30-31, Sakarya, 2001.
21- Awcock, G.J., and Thomas, R., "Applied image processing", McGraw Hill, Inc., 1996.
22- Reiss, T.H., "The Revised Fundamental Theorem of Moment Invariants", IEEE Transaction on Pattern Analysis and Machine Intelligence, 13(8):830-834, August 1991.
23- Ustun, A., "Cisim tanima Problemine Yapay Sinir Aglarinin uygulamasi", MSc Thesis, ITU, 1999