Directional Continuous Wavelet Transform Applied to Handwritten Numerals Recognition Using Neural Networks

Diego J. Romero, Leticia M. Seijas, Ana M. Ruedin

Departamento de Computación, Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires
Buenos Aires, Argentina
dromero@dc.uba.ar, lseijas@dc.uba.ar, anita@dc.uba.ar

ABSTRACT

The recognition of handwritten numerals has many important applications, such as automatic lecture of zip codes in post offices, and automatic lecture of numbers in checknotes. In this paper we present a preprocessing method for handwritten numerals recognition, based on a directional two dimensional continuous wavelet transform. The wavelet chosen is the Mexican hat. It is given a principal orientation by stretching one of its axes, and adding a rotation angle. The resulting transform has 4 parameters: scale, angle (orientation), and position (x,y) in the image. By fixing some of its parameters we obtain wavelet descriptors that form a feature vector for each digit image. We use these for the recognition of the handwritten numerals in the Concordia University data base. We input the preprocessed samples into a multi-layer feed forward neural network, trained with backpropagation. Our results are promising.

Keywords: Neural Networks, Continuous Wavelet Transform, Pattern Recognition.

1. INTRODUCTION

Optical character recognition is one of the most traditional topics in the context of Pattern Recognition and includes as a key issue the recognition of handwritten characters and digits. One of the main difficulties lies in the fact that the intraclass variance is high, due to the different forms associated to the same pattern, because of the particular writing style of each individual. No mathematical model is presently available being capable to give account of such pattern variations [1]. Many models have been proposed to deal with this problem, but none of them has succeeded in obtaining levels of response comparable to human ones. The use of neural networks has provided good results in handwritten character and numeral recognition. Most of the existing literature on this matter applies classical methods for pattern recognition, such as feed-forward networks (multilayer perceptrons) trained with the backpropagation algorithm. This architecture has been acknowledged as a powerful tool for solving the problem of pattern classification, given its capacity to discriminate and to learn and represent implicit knowledge. The performance of a character recognition system strongly depends on how the features that represent each pattern are defined. Kirsch masks [2] have been used as directional feature extractors by several authors [3] [4], as they allow local detection of line segments. On the other hand, the suitability has been explored of a change of representation base by means of principal component analysis [5][6] enabling, without loss of information, to quantify the resolution at which input is represented (with respect to the variance of the projections over the components).

Wavelet transforms have proved to be a useful tool for many image–processing applications. They have given good results for edge detection [7] and texture identification [8]. As a preprocessing step for digit recognition, a one-dimensional discrete orthogonal dyadic wavelet has been applied onto the previously extracted contour of the digit, which is represented with 2 vectors x and y [1]. A one-dimendional discrete multiwavelet transform has also been applied to the previously extracted contour in [9].

The Discrete Wavelet Transform (DWT) provides a decomposition of an image into details having different resolutions and orientations; it is a bijection from the image space onto the space of its coefficients [10], [11]. It has been mainly used for image compression [12]. It is not, however, translation invariant.

On the other hand, the Continous Wavelet Transform (CWT), which is translation invariant, provides a redundant representation of an image. It is mainly used for image analysis. The 2 dimensional CWT has been extended to construct directional wavelet transforms [13], by giving one principal orientation to the wavelet, via stretching one of its axes, and adding a rotational angle

as a parameter. The resulting transform has 4 parameters: scale, angle (orientation), and position (x,y) in the image.

This two–dimensional directional CWT has been applied for pattern recognition in images [14]. In [15] it was used for pose estimation of targets in Synthetic Aperture Radar (SAR) image clips containing regions where the target was previously detected, and experiments over the MSTAR database confirmed the superior robustness of this approach when compared to principle component analysis (PCA). In our preliminary work [16] we have applied it with satisfactory results. In this work we apply the directional CWT as a preprocessing step for recognition of hand written numerals.

Our experiments were performed on the handwritten numeral database from the Centre for Pattern Recognition and Machine Intelligence at Concordia University (CENPARMI), Canada. This database contains 6000 unconstrained handwritten numerals originally collected from dead letter envelopes by the U.S. Postal Service at different locations in the United States. The numerals in the database were digitized in bilevel on a 64 x 224 grid of 0.153 mm square elements, given a resolution of approximately 166 ppi. The digits taken from the database presents many different writing styles as well as different sizes and stroke widths. Some of the numerals are very difficult to recognize even with human eyes. Since the data set was prepared by thorough preprocessing, each digit is scaled to fit in a 16 x 16 bounding box such that the aspect ratio of the image is preserved. Then we apply a directional 2D continuous wavelet transform on each image. We implement the recognition system using a feed-forward neural network trained with the stochastic backpropagation algorithm with adaptive learning parameter. The training and test sets contain 4000 and 2000 numerals from the database (400 / 200 by digit) respectively. Figure 1 shows samples from both sets.

This work is organized as follows: in section 2 the bidimensional CWT is explained, and we give details of our implementation. In section 3 we give the network architecture used in our tests, we give results in section 4 and concluding remarks in section 5.

2. THE TWO–DIMENSIONAL CONTINUOUS WAVELET TRANSFORM

The wavelet transform has given good results in different image processing applications. Its excellent spatial localization and good frequency localization properties makes it an efficient tool for image analysis. The most currently used DWT is

(a) Training set

(b) Test set

Figure 1: Handwritten digits from CENPARMI database, normalized in size.

calculated via filtering the image with both low-pass and highpass filters, followed by subsampling by 2, i.e. omitting one value out of 2. This is carried out on the rows and columns. The subsampling operation, also called decimation, causes the DWT not to be traslation invariant. By this we mean that if we calculate the DWT of an image, shift the same image and calculate the DWT of the shifted image, the values of the 2 DWTs will differ. Since we aim at using the wavelet transform as a preprocessing step for recognition of digits, we want to have the same values for a digit as well as for a shifted copy of the same digit.

This is why we turn to a continous wavelet transform, which is translation invariant.

The directional two-dimensional CWT is the inner product of an image s with a scaled, rotated and translated version of an anisotropic wavelet function ψ.

Let s be a real–valued square–integrable function of 2 variables, i.e. $s \in L^2(\Re^2)$. $S(b, a, \theta)$, the directional CWT of s with respect to a wavelet function $\psi : \Re^2 \rightarrow \Re$, is defined ([13, 14]) in the following way:

$$S(b, a, \theta) = a^{-1} \int_{\Re^2} \psi(a^{-1} r_{-\theta}(b - x)) s(x) \, dx,$$
(1)

where $b = (b_x, b_y) \in \Re^2$ is translation vector, $a \in \Re$ is a scale ($a > 0$), θ is an angle, $0 \leq \theta \leq 2\pi$, and $r_\theta(x)$ is a rotation of angle θ, acting upon a vector $x = (x_1, x_2) \in \Re^2$ as follows:

$$r_\theta(x) = (x_1 \cos \theta - x_2 \sin \theta, x_1 \sin \theta + x_2 \cos \theta).$$
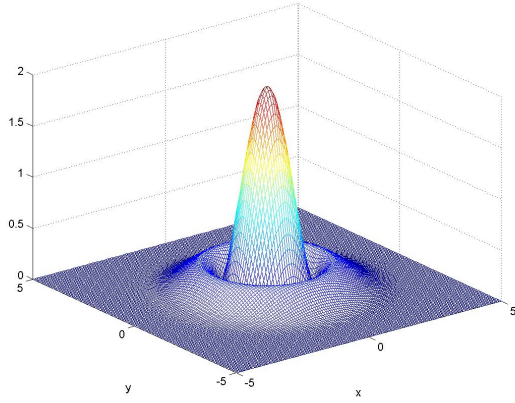(2)

In this approach there is no multiresolution property, as with the DWT. In order to be able to reconstruct the image s from its wavelet transform S, function ψ must be admissible, this is
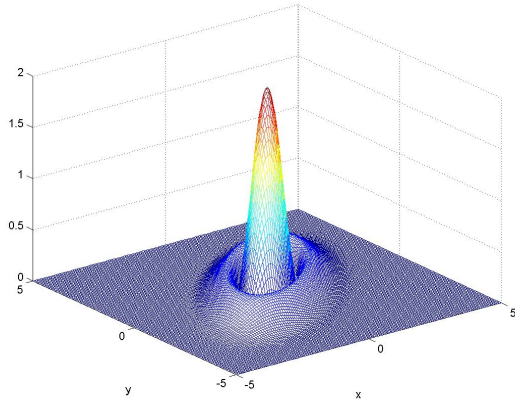
equivalent to the zero mean condition:

$$\int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \psi(x)dx = 0. \qquad (3)$$

For our wavelet, we have chosen the Mexican Hat, defined as

$$\psi_{mh}(x,y) = (2 - (x^2 + \frac{y^2}{\epsilon}))\ e^{-\frac{(x^2+\frac{y^2}{\epsilon})}{2}}. \qquad (4)$$

(a) Isotropic wavelet

(b) Anisotropic wavelet

Figure 2: (a) Mexican Hat ($a = 1, \theta = 0°, \epsilon = 1$), (b) Directional Mexican Hat ($a = 0.8$, $\theta = 135°$, $\epsilon = 5$).

Note that when $\epsilon = 1$, we have the usual Mexican Hat wavelet, which is isotropic. For $\epsilon \neq 1$, we have an anisotropic Mexican Hat wavelet. By giving it a special orientation, we have the directional Mexican Hat wavelet $\psi(r_{-\theta}(x))$. In figures 2 and 3 we have the 3d plot and level curves of $\psi(a^{-1} r_{-\theta}(b-x))$ for different values of $a, b, and \theta$. The 2–dimensional directional CWT provides a redundant representation of an image in a space of scale, position and orientation. To reduce the complexity of this representation, we work with

the so-called "Position Representation", in which the angle and the scale have been fixed:

$$S_{a\theta}(b_x, b_y) = S(b_x, b_y, a, \theta) \qquad a,\ \theta\ fixed. \qquad (5)$$

(There are other possible representations, obtained by fixing other parameters, such as the scale–angle representation).

Through observation that the most common slant is of 135° (we consider the angle formed with the negative x axis, clockwise), we have fixed angle $\theta = 135$. Experiments revealed the convenience of setting the scale to $a = 0.8$.
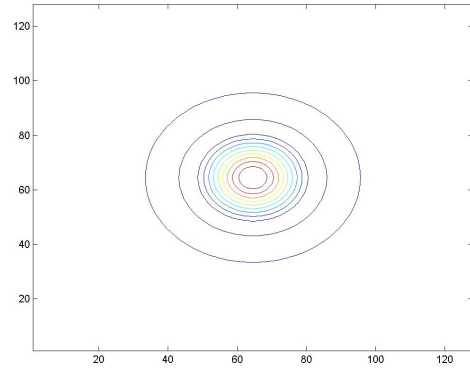
(a) Isotropic wavelet
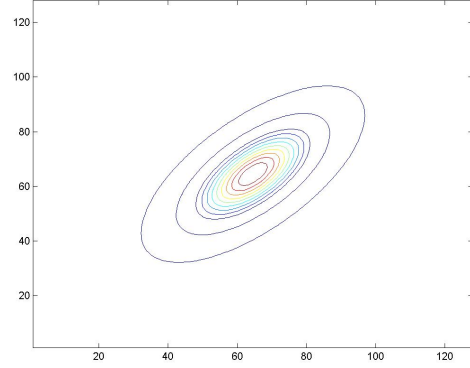
(b) Anisotropic wavelet

Figure 3: Level curves for (a) Mexican Hat ($a = 1, \theta = 0°, \epsilon = 1$), (b) Directional Mexican Hat ($a = 0.8, \theta = 135°, \epsilon = 5$).

Our sample digits are binary 16x16 images. For b_x, 16 regularly spaced values were chosen in interval $[-32, 32]$. The same was done for b_y. This gives a transform that is a real 16x16 image. To obtain a binary image, the transform was thresholded. In images 5 to 8 we show examples of the preprocessing step to a few digit samples.

3. RECOGNITION SYSTEM

Multilayer feed-forward networks have been used in optical character recognition systems for many

years. These networks can be treated as feature extractors and classifiers. Figure 4 shows an example of architecture for this kind of network. Each node in a layer has full connections from the nodes in the previous layer and the proceeding layer. There are several layers of neurons: the input layer, hidden layers and the output layer. During the training phase, connection weights are learned. The output at a node is a function of the weighted sum of the connected nodes at the previous layer.

Figure 4: Example of feedforward multilayer network architecture.
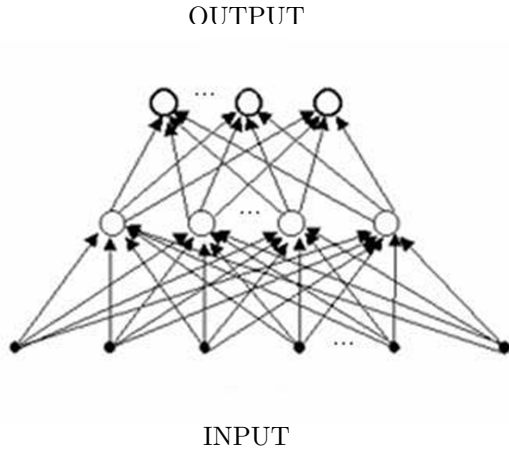
OUTPUT

INPUT

We use a two-layer feed-forward neural network in our experiments. The number of nodes in each layer is given by 160 x 10. The input layer depends on the input feature size (preprocessed images of 16x16), so the number of neurons is 256. Each output node is associated with a different class or digit. Each numeral presented at the input layer feeds into the network until the computation of the network output is performed.
For each iteration or time step t, we define:

- w_{ij} weight that connects ith unit from mth layer with jth unit from $m-1$th layer

- $h_i = \sum_{j \in J} w_{ij} V_j$ net input to ith unit, J includes all neurons from preceding layer

- $V_i = g(h_i)$ output from ith unit; g is the activation function of the unit. If V_i is in the input layer, then its value equals the ith component of the input pattern.

- ς_i desired (target) output of ith unit.

- O_i actual output of ith unit in the output layer

- $E(t) = \frac{1}{2} \sum_{i \in C} (\varsigma_i - O_i)^2$ where C includes all neurons from the output layer; it defines error at iteration t.
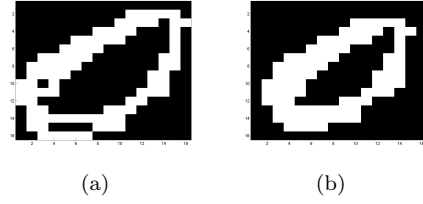
We define the cost function by

(a)   (b)

Figure 5: (a) An original sample digit 0. (b) Same digit after CWT– preprocessing.
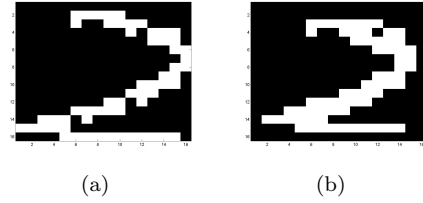

(a)   (b)

Figure 6: (a) An original sample digit 2. (b) Same digit after CWT– preprocessing.

- $E(w) = \frac{1}{N} \sum_{\mu \in P, i \in C} (\varsigma_i^\mu - O_i^\mu)^2$ where C includes all the neurons in the output layer and P includes all training patterns.

The network was trained with the stochastic back-propagation algorithm with momentum and adaptive learning parameter [17] [18]. The algorithm gives a prescription for changing the weights w to learn a training set of input-output pairs. The basis is gradient descent; it allows minimizing the cost function, which measures the system's performance error as a differentiable function of the weights. The stochastic approach allows wider exploration of the cost surface: a pattern chosen in random order is presented at the input layer and then all weights are updated before the next pattern is considered. This decreases the cost function (for small enough learning parameter) at each step, and lets successive steps adapt to the local gradient. The back-propagation update rule for input pattern at the iteration t has the form

- $w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij}(t)$

- $\Delta w_{ij}(t) = -\eta \frac{\partial E(t)}{\partial w_{ij}(t)} + \alpha \Delta w_{ij}(t-1)$

where η is the learning parameter, and α is the momentum parameter (it allows larger effective learning rate without divergent oscillations occurring). Values 0.01 and 0.9 as initial learning rate and momentum parameter respectively were used in our experiments. We train the neural network up to 3500 ephocs.
The logistic function defined by

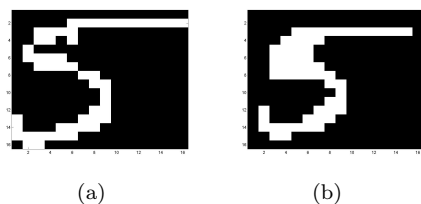$$g(h) = \frac{1}{1 + e^{-h}} \qquad (6)$$

(a)   (b)

Figure 7: (a) An original sample digit 5. (b) Same digit after CWT– preprocessing.
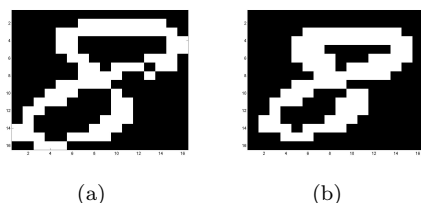

(a)   (b)

Figure 8: (a) An original sample digit 8. (b) Same digit after CWT– preprocessing.

was used as activation function associated with neurons from hidden and output layers.

4. RESULTS

In this section we show the results obtained after the preprocessed digits were classified with the neural network described in section 3.

These results are listed in tables 1 and 2. For both tables, in the first column we have the digit, in the second column we give the fraction of incorrectly classified samples, in the third column the fraction of correctly classified samples, and in the last column we give the recognition percentage for that digit.

In the last row we list the totals over the whole set.

The percentage of correctly classified patterns was 99.17% and 90.20% for the training set and the test set, respectively. This result is promising, as it improves over the percentages obtained with the same nework architecture with no preprocessing stage (87.1% of test patterns recognized).

The performance obtained in this work is comparable to to other results reported in the literature for the same data set [9] [1].

6. CONCLUSIONS

We have presented a preprocessing stage based on the directional CWT in 2 dimensions, prior to the training of a feedforward multilayer neural network for handwritten numeral classification.

With our choice for the parameters of the directional CWT, we obtained an efficient wavelet descriptor for the handwritten numerals. The directional CWT is translation invariant. Because

Digit	Miss Classified	Correctly Classified	Recogn. %
0	5/400	395/400	98.75
1	1/400	399/400	99.75
2	4/400	396/400	99.00
3	4/400	396/400	99.00
4	4/400	396/400	99.00
5	1/400	399/400	99.75
6	4/400	396/400	99.00
7	6/400	394/400	98.50
8	2/400	398/400	99.50
9	2/400	398/400	99.50
Total	33/4000	3967/4000	99.17

Table 1: Results obtained over the training set.

Digit	Miss Classified	Correctly Classified	Recogn. %
0	20/200	180/200	90.00
1	6/200	194/200	97.00
2	18/200	182/200	91.00
3	32/200	168/200	84.00
4	7/200	193/200	96.50
5	30/200	170/200	85.00
6	17/200	183/200	91.50
7	15/200	185/200	92.50
8	36/200	164/200	82.00
9	15/200	185/200	92.50
Total	196/2000	1804/2000	90.20

Table 2: Results obtained over the test set.

we chose the Mexican hat wavelet, the transformed and thresholded patterns had a smoother contour. By setting the angle of the directional CWT to the most common slant, we obtained digits with a wider stroke. By taking $a = 0.8$, the size of the digit was reduced, adding a black frame around the bounding box. All these properties added up to the posterior identification of the digit with a neural network.

Our method was tested on the database of the Concordia University, Canada. Our results are comparable to other proposed techniques [9] [1], which are also based on a wavelet–transform preprocessing step, and also train a feedforward neural network for pattern classification. These mentioned works employ a wavelet or multiwavelet transform in one dimension, and require the identification of the contour of the digit, which is not necessary in our case.

For future work we plan to exploit the invariance properties of the directional CWT more fully, in order to improve our classifier.

7. REFERENCES

[1] P. Wunsch, A.F. Laine: Wavelet Descriptors for multiresolution recognition of hand-printed characters, Pattern Recognition, Vol. 28, No. 8, 1995, pp. 1237-1249.

[2] W. Pratt: *Digital Image Processing*, New York, Wiley, 1978.

[3] D. Gorgevik, D. Cakmakov: An Efficient Three-Stage Classifier for Handwritten Digit Recognition, Proceedings of the 17th International Conference on Pattern Recognition (ICPR'04), Vol.4, 2004, pp. 507-510.

[4] L. M. Seijas y E. C. Segura: Un clasificador neuronal que explica sus respuestas: aplicación al reconocimiento de dígitos manuscritos, Proceedings IX Congreso Argentino de Ciencias de la Computación (CACIC 2003), La Plata, Argentina, 2003.

[5] E. López-Rubio, J. Muñoz-Pérez, J. Gómez-Ruiz: A principal components analysis self-organizing map. Neural Networks, Vol. 17, No. 2, 2004, pp.261-270.

[6] B. Zhang, Fu, M., Yan, H.: A nonlinear neural network model of mixture of local principal component analysis: application to handwritten digits recognition, Pattern Recognition, Vol. 34, No. 2, 2001, pp. 203-214.

[7] Ana Ruedin: A Nonseparable multiwavelet for edge detection, Wavelet Appl. Signal Image Proc. X, Proc. SPIE, Vol. 5207, 2003, pp. 700-709.

[8] S. Liapis, G. Tziritas, Color and Texture Image Retrieval Using Chromaticity Histograms and Wavelet Frames, IEEE Transactions on Multimedia, Vol.6, No.5, 2004, pp. 676-686.

[9] G. Y. Chen, T. D. Bui and A. Krzyzak: Contour-Based Handwritten Numeral Recognition using Multiwavelets and Neural Networks, Pattern Recognition, Vol.36, No.7, 2003, pp.1597-1604.

[10] I. Daubechies: *Ten lectures on wavelets, Society for Industrial and Applied Mathematics*, 1992.

[11] S. Mallat: *A Wavelet Tour of Signal Processing*, Academic Press, 1999.

[12] A. Skodras, C. Christopoulos, T. Ebrahimi: JPEG2000: The upcoming still image compression standard, Elsevier Pattern Recognition Letters, Vol. 22, 2001, pp. 1337-1345.

[13] J.P. Antoine, P. Vandergheynst, K. Bouyoucef, R. Murenzi: Target detection and recognition using two-dimensional isotropic and anisotropic wavelets, Automatic Object Recognition V, SPIE Proc., 2485, 1995, pp. 20-31.

[14] J.P. Antoine, R. Murenzi: Two dimensional directional wavelets and the scale-angle representation, Signal Process. 53, 1996, pp. 259-281.

[15] L. Kaplan, R. Murenzi: Pose estimation of SAR imagery using the two dimensional continuous wavelet transform, Pattern Recognition Letters 24, 2003, pp. 22692280.

[16] J. Romero, L.Seijas, A. Ruedin: Reconocimiento de Dgitos Manuscritos Usando La Transformada Wavelet Continua en 2 Dimensiones y Redes Neuronales, XII Congreso Argentino de Ciencias de la Computacin CACIC 2006.

[17] J. Hertz, A. Krogh, R. Palmer: *Introduction to the Theory of Neural Computation*, Santa Fe Institute Editorial Board, 1990.

[18] S. Haykin: *Neural Networks A Comprehensive Foundation*, Prentice Hall, 1999.