ABSTRACT

In this paper we have recognized the nearly 1200 Latin handwritten characters collected from people using artificial neural network. We used backpropagation algorithm for supervised learning. In pre-processing and feature extraction step, normalization and edge detection has been performed. Cellular neural network (CNN) is used for edge detection. CNN are a parallel computing paradigm similar to neural networks, with the difference that communication is allowed between neighbouring units only. In this system we achieved 84.5% recognition accuracy. To reach this percentage it is observed with graphics how input datas, network parameters and training period affect the result. Then the character recognition performance of the network according to changeable parameters is analysed. And factors that increase performance of system are determined.

Keywords: Character Recognition, Artificial Neural Networks, Backpropagation, Cellular Neural Networks.

ÖZET

Bu çalışmada farklı kişilerden toplanan yaklaşık 1200 el yazısı karakterini tanıyan bir yapay sinir ağları sistemi geliştirilmiştir. Öğreticili öğrenme genel yapılamaları kullanılmıştır. Ön-işleme ve nitelik çıkarma gibi adımlarda, normalizasyon ve kenar belirleme gerçekleştirmiştir. Bu aşamada kenar belirleme için Hücresel Yapay Sınır Ağları (Cellular neural
network) teknolojisi kullanılmıştır. CNN, yapay sinir ağılarından farklı olarak iletişim sadece komşu elemanlarla sağlanabildiği paralel bir işleme teknigidir. Gelişilen sistemde %84.5 tanıma oranı sağlanmıştır. Bu sonuç ulaşmak için giriş verileri, ağ parametreleri ve eğitim süresi gibi kriterlerin sisteme olan etkisi elde edilen grafiklerle incelenmiştir. Sonrasında değişen parametrere göre ağın karakterleri tanıma performansı analiz edilmiştir. Ve performansı yükselen faktörler belirlenmiştir.

Anahtar Kelimeler: Karakter Tanıma, Yapay Sinir Ağıları, Geriye Yayılım Algoritması, Hücresel Yapay Sinir Ağları

1. INTRODUCTION

Optical character recognition, usually abbreviated to OCR, is a type of computer software designed to translate images of handwritten or typewritten text (usually captured by a scanner) into machine-editable text, or to translate pictures of characters into a standard encoding scheme representing them.

The optical character recognition is one of the earliest applications of Artificial Neural Networks. Classical methods in pattern recognition do not as such suffice for the recognition of visual characters [1]. A neural system is able to work with noisy, unknown and indefinite data. In this work a handwritten recognizing system has developed.

1.1 Artificial Neural Networks (ANN)

An Artificial Neural Network is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. It is composed of a large number of highly interconnected processing elements (neurones) working in unison to solve specific problems.

2 FEATURE EXTRACTIONS

Feature extraction involves simplifying the amount of resources required to describe a large set of data accurately. For feature extraction edge detection with Cellular Neural Network (CNN) is used in this study. Figure 1 shows the block diagram of our system.

Before feature extraction, characters from the scanned image were normalized from various sizes into 30 x 30 pixels as shown in Figure 2. Normalization has the capability of extracting all of the invariant features from
an image using only a small amount of information about the image. It is also called contrast stretching [2].

![Image of a 30 x 30 'B' Character](attachment:image.png)

**Figure 2: A 30 x 30 'B' Character**

### 2.1 Edge Detection

Edge detection is a fundamental tool used in most image processing applications to obtain information from the frames as a precursor step to feature extraction and object segmentation [3]. In this study, Cellular Neural Network (CNN) which is introduced [4-5] technology has been used for edge detection. CNN is defined by an MXN rectangular array of cells C(i,j) located at site (i,j), i=1,2,...,M and j=1,2,...,N. Each cell C(i,j) is defined mathematically by:

\[
X_{ij} = \frac{dx_{ij}}{dt} = -x_{ij} + \sum_{C(j:j \neq (i,j))} A(i,j,k,l) y_{ij} + \sum_{C(j:j \neq (i,j))} B(i,j,k,l) u_{ij} + z_{ij} \quad (1)
\]

where \(x_{ij}\), \(y_{ij}\), \(u_{ij}\) and \(z_{ij}\) are called state, output, input and threshold of cell C(i,j), respectively. \(A(i,j,k,l)\) and \(B(i,j,k,l)\) are called the feedback and the input synaptic operators. Here state and threshold are taken zero as initial. Figure 3 shows edge detector uses a pair of 3x3 convolution masks A and B.

![Convolution Masks A and B](attachment:convolution_masks.png)

**Figure 3: a) A Template b) B Template**
Convolution is illustrated with A and B templates using Euler approach. Mathematically we can write the convolution as equation 5. After convolution all images are represented by their edges in Table 1.

\[ O(i, j) = \sum_{k=1}^{m} \sum_{l=1}^{n} I(i + k - 1, j + l - 1)K(k, l) \]  

(2)

3 SYSTEM OF NEURAL NETWORK

3.1 Architecture of Artificial Neural Network

In this study our network consists of an input layer, a single hidden layer and an output layer. Latin characters are 30 x 30 = 900 is pixels. Therefore 900 nodes which symbolize each character exist in the input layer, 35 nodes in the hidden layer. Because of Latin alphabet consists of 26 characters, output layer has 26 neurons (Figure 4). Each output layer neuron set represents a character. Only first neuron target value is ‘1’ and others are ‘0’ for character ‘A’. 

Figure 4: Neural Network Architecture
4 MATERIALS AND METHODS

4.1 Backpropagation Algorithm

Backpropagation learning algorithm [6] is used for training in this neural system. The p-th sample input vector of pattern \( X_p = (X_{p1}, X_{p2}, \ldots, X_{pN0}) \) and corresponding output target \( T_p = (T_{p1}, T_{p2}, \ldots, T_{pNM}) \) is presented. The input values to the first layer are passed.

\[
Y_{0i} = X_{pi}
\]  

(3)

For every neuron \( i \) from input to output layer, the output from the neuron is found:

\[
Y_{ji} = f\left(\sum_{k=1}^{N_{j-1}} Y_{(j-1)k} W_{jik}\right)
\]  

(4)

Here \( f(x) \) is sigmoid activation function.

\[
f(x) = \frac{1}{1 + \exp(-x)}
\]  

(5)

For the output layer, the error value is:

\[
\delta_{Mj} = Y_{Mj} (1 - Y_{Mj}) (T_{pj} - Y_{Mj})
\]  

(6)

For hidden layers, the error value is:

\[
\delta_{ji} = Y_{ji} (1 - Y_{ji}) \sum_{k=1}^{N_{j+1}} \delta_{(j+1)k} W_{(j+1)ik}
\]  

(7)

Weight change is:

\[
\Delta W_{jik} = \beta \delta_{ji} Y_{(j-1)k}
\]  

(8)

\( \beta \) is a constant learning rate.

4.2 Running of Algorithm

In this study backpropagation algorithm runs with a large data set about 1200 input characters. In training a large data set is applied to network. To train the network first pattern is applied and trained. The network weights are updated. Next we apply the second letter and do the same, then the third and so on [7]. Once all 26 letters are done, we return to the first one again and repeat the process until the error becomes small. We stop training once the network can recognize all the letters successfully, so the error fall to a lower value first. We evaluate the total error of the network by adding up all the
errors for each individual neuron and then for each pattern in turn to give a total error as shown in Figure 5.

![Neural Network Architecture](image)

4.3 Stopping Training

Stopping criterion for the network specifies the value of the mean squared error at which the network can stop. The system has been trained until MSE (Mean Square Error) falls to 0.0001. MSE is defined below. \( P \) represents pattern number, \( N_M \) is total neuron number of a pattern, and \( j \) is the index of neurons.

\[
MSE = \frac{1}{2D} \sum_{P=1}^{P} \sum_{j=1}^{N_M} (T_j - O_j)^2
\]

5 PERFORMANCE ANALYSIS

According to test results, we determined some parameters affect the system. Performance is observed by changing parameter values each time. So we reach the network parameters which increase performance by the help of graphics. These parameters will be investigated respectively: initial weights, number of process units in layers, learning rate, selection of characters, representing characters to network.

5.1 Initial Weights

To start training process, the initial weights are chosen randomly. There is no technique determines how to choose the weights as initial. In this work, it is seen that, if the weight range is selected as big values, network rambles between local solutions. Else, converging to zero of MSE wastes much time. Therefore, weights are selected in \([-0.5, +0.5]\) range. Error is high at the beginning with these values. But in the next iterations MSE decreases fastly. In Figure 6 the decrease of the error is shown when connection weights are selected in \([-0.5, +0.5]\) and \([-0.1, +0.1]\) range.
Performance Analysis Of Neural Network Handwritten Character Recognition System Using Cnn Edge Detection

5.2 Number of Process Units in Layers

There is also no technique to determine the number of neurons in the layers. This number affects the network performance directly [8]. Although the error is small, for smaller error possibility we continue to train network with different optimal number of neurons. Firstly, 30 neurons were used in hidden layer. Iteration time was short but rate of error decrease was too small. Hence this architecture is not preferred. Afterwards hidden layer neurons were raised to 40. In this training, rate of error decrease was great but in next iterations MSE began to stay in a same undesired value through iterations. And training of network became more difficult. As a solution 35 neurons are used in hidden layer for optimality. So Mean Square Error decreases regularly after iterations as seen in Figure 7. This decrease stops when MSE is acceptable.

Figure 6: MSE – Iteration Graphic in Different Initial Weights

Figure 7: MSE – Iteration Graphic in Different Hidden Layer Neuron Numbers
5.3 Learning Rate

Learning rate is a numerical variable specifying the proportion of the error derivative by which the weights will be adjusted during training. Basically, it is a variable that affects how the network learns from its errors. If learning rate is big, oscillation occurs. Otherwise iteration step takes much time. Figure 8 shows MSE decrease after iterations at learning rate 0.2, 0.4 and 0.8. When learning rate is 0.8, network needs more iterations than 120,000. This causes time waste, so is not preferred. If 0.4 is selected as learning rate, oscillation does not occur but converging to zero of MSE is not so simple. In this study, we choose it 0.2. We see in Figure 8 that error converges to zero after enough iteration without so much time waste.
5.4 Sample Number in Training Set

Image acquisition is significant for handwritten character recognition system. It is important to choose characters to represent problem space well. Our training set consists of various characters either noisy or smooth. The network adjusts connection weights according to training set and it can only comment about images that are represented in training step. All acquired images are randomly separated to two groups of training set and testing set. Test set elements are never represented to network in training stage. The test set characters that can not be recognized by system are determined. And they are added to training set, and then training begins again with the new set. Thus rising of system performance is observed. Recognition accuracy increases when character number in training set rises as seen in Figure 9.

5.5 Representing Characters to Network

There is two ways to represent training set to neural network that representing the characters in order or randomly. In orderly representing, first character class in training set is represented to network respectively. Afterwards the characters belong to second class are represented. This operation is repeated until samples of last character class (‘Z’) are finished. The training ends when all characters are learned. On the other hand in random representing type characters are not represented to network class by class in order. For instance, after an ‘E’ sample is trained, a ‘Y’ character sample can be collocated next. Recognition accuracy increases as seen in Figure 10 when we give characters to neural system randomly. Although network learns a
character class well because of representing the samples of this class orderly, there is a threat that network may forget about the last class when the next class is represented. Because network will not study again with a sample belongs to a previous class.

6 SIMULATION RESULTS

In this study we acquired 1200 various handwritten characters. Approximately after 115,000 iterations error reaches to an acceptable value (Figure 8). MSE decrease after epochs is shown in Table 2. In neural network system initial weights are between -0.5 and +0.5. Learning parameter is chosen 0.2. Training samples are applied to network randomly. And 35 process units are used in hidden layer. For it has observed that system performance is higher with these values. After training, 110 characters were used to test system. 93 of them have been correctly classified by network. Therefore recognition performance of neural network is measured 84.5%. Some characters are misclassified as shown in Table 3.

Table 2: MSE Decrease after Epochs

<table>
<thead>
<tr>
<th>Epoch</th>
<th>MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>4,000</td>
<td>1.0270</td>
</tr>
<tr>
<td>20,000</td>
<td>0.2340</td>
</tr>
<tr>
<td>40,000</td>
<td>0.2290</td>
</tr>
<tr>
<td>52,000</td>
<td>0.2080</td>
</tr>
<tr>
<td>60,000</td>
<td>0.1476</td>
</tr>
<tr>
<td>80,000</td>
<td>0.0900</td>
</tr>
</tbody>
</table>

Figure 10: Recognition Accuracy with Different Representing of Characters
Performance Analysis Of Neural Network Handwritten Character Recognition System Using Cnn Edge Detection

<table>
<thead>
<tr>
<th>Test Character</th>
<th>System Output</th>
<th>Desired Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>ft</td>
<td>R</td>
<td>A</td>
</tr>
<tr>
<td>K</td>
<td>K</td>
<td>K</td>
</tr>
<tr>
<td>r</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>H</td>
<td>H</td>
<td>H</td>
</tr>
<tr>
<td>M</td>
<td>M</td>
<td>M</td>
</tr>
<tr>
<td>K</td>
<td>R</td>
<td>R</td>
</tr>
</tbody>
</table>

Table 3: Classification of Characters

7 CONCLUSIONS

In our handwritten character recognition system we use artificial neural networks that offer several advantages in character recognition despite the computational complexity involved. Using neural networks for CR has many advantages. Some of these include automatic training and retraining, less storage, robust performance, potential for parallelization, graceful degradation, working with noisy and missing data.

In this study, according to graphics, we select the initial connection weights in [-0.5, +0.5] range. Because in this state MSE converges to zero in less time. As it is more possible error to converge 0.0001 after enough iteration without so much time waste, the learning rate is chosen 0.2. And 35 neurons are used in hidden layer for time waste and error decrease.

Feature extraction is performed as pre-processing by using cellular neural networks. In future extraction we detected the edges of handwritten characters. Edge detection of an image reduces significantly the amount of data and filters out information that may be regarded as less relevant, preserving the important structural properties of an image. As a result of edge detection number of pixel ‘1’ in binary matrices of characters reduces, so operating time drops.

The experiment results show recognition rate is 84.5%. Although decrease of error is achieved it is searched if network performance increases with different parameters or image acquisition. Thus we reached an acceptable high accuracy. In another study of handwritten character recognition system an algorithm based on structural characteristics, histograms and profiles, is presented. The well-known horizontal and vertical histograms are used, in combination with the newly introduced radial histogram, out-in radial and in-out radial profiles for representing characters, as vectors. The K-means algorithm is used for the classification of these vectors. And their accuracy...
result is 72.8%. The highest rates are achieved on restricted tasks, such as the 99% rate achieved on machine-printed characters. For the characters in machine-print are not so distinct from each other. Because of training set images and test set images are similar, system can correctly and easily classify them. But as mentioned before, we acquired handwriting of different people. Therefore character shapes and sizes vary depending person. This decreases recognition accuracy.

REFERENCES