

EFFICIENCY OF USING THE HYBRID NEURAL MODEL CNN + LSTM + FUZZY CTC FOR HANDWRITING RECOGNITION

Turakulov Sh.

Tashkent University of Information Technologies named after Muhammad al- Kharizmi

<https://doi.org/10.5281/zenodo.8033725>

Abstract. *The article discusses the existing problems of handwriting recognition and the solution to the segmentation problem. The mode of operation and structural explanations of the model of the hybrid neural network CNN + LSTM + fuzzy CTS are given here. The efficiency of the hybrid model was demonstrated using a comparative analysis with the CNN, CNN + CTC, CNN + LSTM models. 90% of the recognition results for handwriting recognition were achieved in case of availability of sufficient dataset. Algorithms of fuzzy set theory for CTC in handwriting recognition have been developed. Based on these, effective results were obtained.*

Keywords: *recognition, segmentation, neural network, hybrid neural network, series segmentation, filtering.*

CNN- Convolution neural network

CTC- Connectionist Temporal Classification

LSTM- Long short-term memory

Fuzzy-fuzzy algorithms

1.Introduction. As the science and technology is developing in the society today, we can see that we achieved almost good indicators in resolving existing problems of recognition. These include facial recognition, eye recognition, handprint recognition, voice recognition and others. In text image recognition, FineReader has made it much more convenient for people with a 99 percent result for printable letters. There are also manuscripts that need to be digitized to automate their processing, retrieval and storage. Research on this issue is being conducted in various countries around the world. The main developments of our foreign colleagues in the field of handwriting recognition mainly consist of solving the problem of segmenting texts, to speed up the processing of information and get rid of clutters that interfere with working with the text, for example, missing or incomprehensible fragments, stains on paper, as well as increasing the number of languages. The recognition algorithm itself can be used for this purpose. However, despite the progress made by scholars so far, practical and theoretical problems have not been solved yet. [1,2]

2.Main part. There are steps to recognize handwritten texts and each step is important in recognizing them.

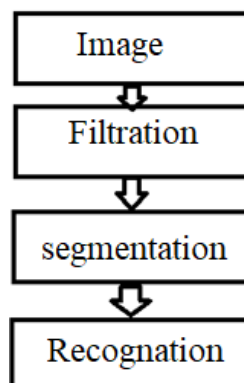


Figure 1. Steps of recognizing the text

In the process of recognition, an image is selected, however the image cannot always be what we expect. That is, it may be from an old source. We can see all kinds of dullness and blemishes. If it is written in a normal notebook, the lines will interfere. An excess of a certain amount of colors can lead to huge errors and taking this into account it has to be filtered.

Segmentation is the process that leads to the ethalon of recognition. There are different ways to recognize the text: line segmentation, word segmentation, letter segmentation. We chose row segmentation for the CNN + LSTM + CTC model.

The line segmentation has been improved with obscure set algorithms to ensure proper segmentation.

The effective organization of recognition depends on several factors and the following hybrid recognition algorithm has been developed taking into account that the text will not always be the image we expect.

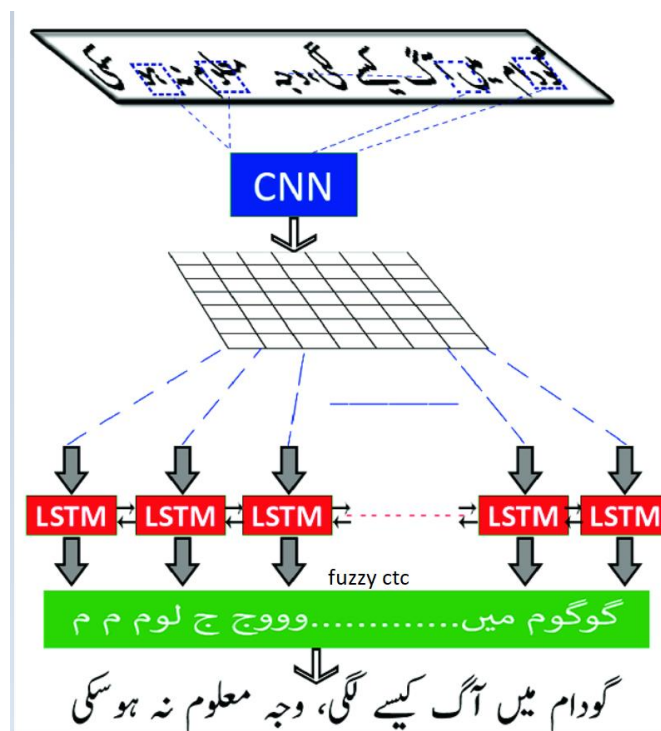


Figure 2. CNN+LSTM+fuzzy CTC working illustration

The general calculation procedure of CNN was chosen as follows:

CNN=40:3x3 pool 2x2, cnn=60:3x3, pool=2x2, lstm=200, drout=0.5. ctc loss

Recognition software is developed using Python. Here is the identification procedure based on its code. Here 40 filters and 3x3 pointers are considered as core function and we use one pool=2x2. We use 60 filters 3x3 core function, pool = 2x2 i.e. 2 CNN. They are used to get small input settings.

The CNN structure serves to obtain glyphs of given words from a lines of segmented images. The line image is large in size and can be swiped 40 times using the 3x3 core function and the pool 2x2. In the next cnn section, a swipe operation is performed with 60 3x3 core functions and pool = 2x2.

The Cnn neural network output class will be equal to the number of rows. In an interconnected neural network, a multilayer perceptron neural network was used and classified using inner layers with 16 neurons and a relay activation function.

On CNN, input neurons serve as numerical representations of image lines:

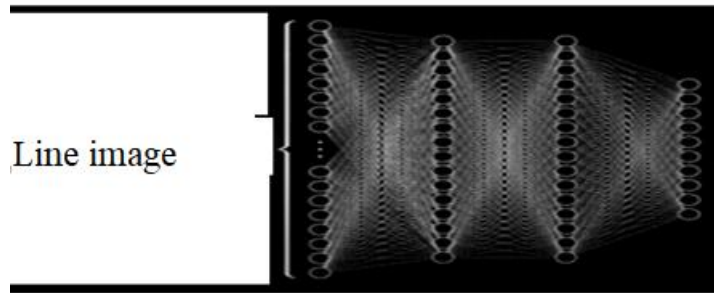


Figure 3. Image processing using neurons

If the image size is large, the number of input neurons will increase accordingly. For example, in a 28x28-pixel image, the number of input neurons is 784, which is 13,002 internal neurons in a multilayer perceptron neuron.

In CNN, we can extract important features of an image using a core matrix and have an active matrix using a pool swipe operation. Using this active matrix, we can store the basic parameters of the classes of this shape. This led to a shortening of the input neurons given to us. Let's look at it using symbols as an example:



Figure 4. Different supplies of each character

We can distinguish its vertical, horizontal, arc shapes using a CNN core matrix.

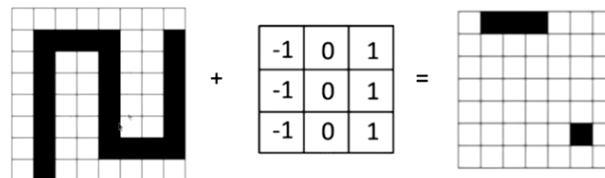


Figure 5. Extracting the horizontal parameters of the number

This means that the core matrix is used to isolate the main parts, which dramatically reduces the input values to the neuron.

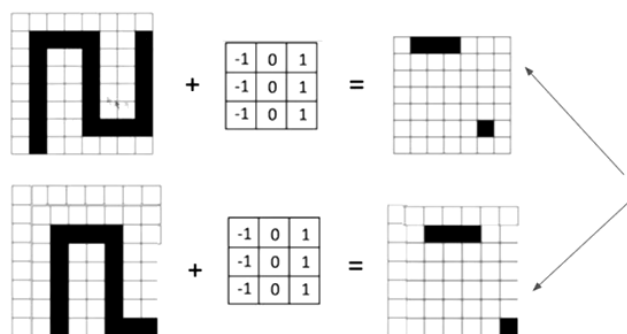


Figure 6. Separation of the main features by 2 core switches where image line is inserted into the CNN input layer.

$$(f * g)[m,n] = \sum_{k,l} f[m-k, n-l] * g[k,l] \quad (1)$$

Here f – input matrix, g – core matrix.

A new active matrix is formed by using a core of the given $n \times n$ size (1): a pool is made in the active matrix.

The results from the line segmentation will be separated and classified by the glyphs, i.e. the general forms of the words and letters, which serve as basis for each line.

CNN output values are considered as input values for the LSTM and serve to shape the grammar of the language. The output values at Fuzzy CTC are given and the results are expected to be obtained depending on the loss error.

We use fuzzy to calculate the loss value when separating the above letters. The loss function does not work very well in a loss-limited database. The reason is that some of the obscure sets can achieve good results at values being too small and others being too large.

We use unclear membership functions to reduce the value of the loss function. We attempt to construct mathematical models in order to mathematically describe information that is not clear for us.

This concept is based on the idea that elements with the same properties that make up a given set have different properties, which means that they can belong to a given set to different degrees. Based on this approach, comments such as "some element belongs to a given set" make no sense, because it is necessary to indicate to what extent or "how strong" a particular element satisfies a given set [8]. It is a carrier - a universal set to which all the results of observations in the field of evaluated quasistatistics are relevant. For example, if we are separating characters, the dots are the intersection of the real axis.

Set of pairs is called a fuzzy set \tilde{A} in a universal set $(\mu_{\tilde{A}}, \mu)$ juftliklar majmuiga aytiladi, where $(\mu_{\tilde{A}})$ is the degree to which the element belongs to an obscure set. The degree of membership is a number in the range [0, 1]. The higher the degree of membership, the more the element of the universal set relates to the properties of the obscure set [7].

An obscure set is a set of carrier values for which each value of the carrier corresponds to the degree to which this value belongs to set A. A set of points belongs to a given set to a certain degree μ and is called a membership function.

Membership function is a function that allows you to calculate the degree to which a desired element in a universal set belongs to an obscure set.

In many practical cases, the membership function must be evaluated on the basis of its partial information, let's say, it should be evaluated taking into account the values in a finite set of reference points $x_1, x_2 \dots x_n$.

In this case, it is said to be partially determined using a "commentary example". Figure 7 shows the main views of the membership function used in the theory of obscure sets.

Symmetric Gaussian membership function:
$$\mu(x) = \frac{(x-b)^2}{e^{2c^2}}$$

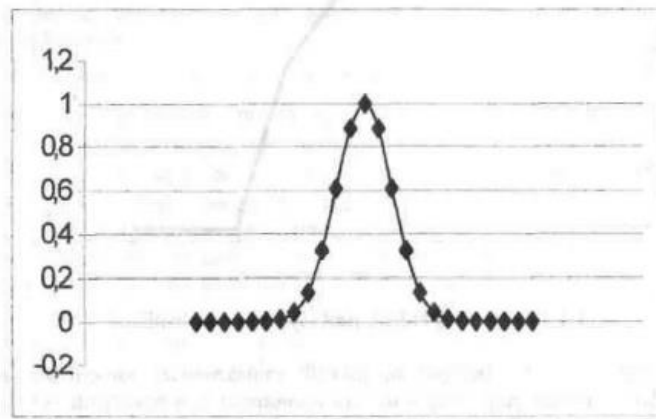


Figure 7. Membership function graph

It consists of x elements that are the carriers of an indefinite set $(x) > 0$: $\sup A = \{x \in X, \mu(x) > 0\}$. $\mu = \frac{1}{2}$ is called the transition point $x \in X$ of an obscure set A.

Set carrier, transition point and singleton. An obscure set $\mu(x) = 1.0$ the carrier of which is a single point derived from X is called a singleton. The height of a obscure set, a normal obscure set. The height of an obscure set A is the upper limit of the membership function: $g(A) = \sup_{x \in \mu A} \mu(x)$ If $\exists x \in \mu A(x) = 1$ an obscure set A is considered normal. The height of ordinary obscure set is equal to one [7].

With the help of membership function the nearest symbol is selected and the loss value is decreased.

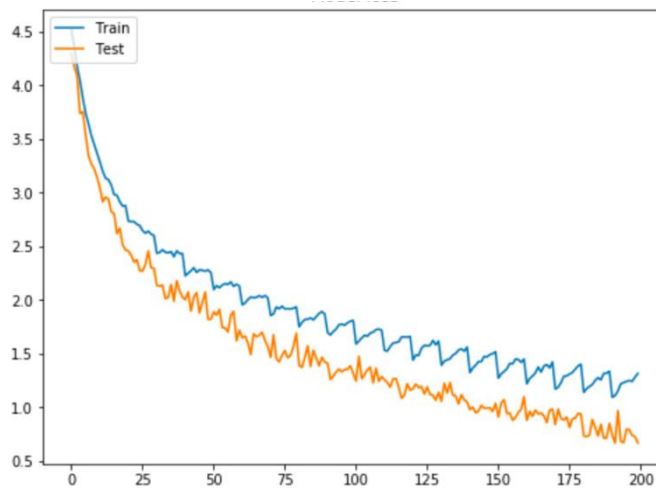


Figure 8. Comparison of taught model of loss values

The results are presented in tabular form.

Table 1

The comparative analysis of selected models

No	Data set	Title of the literature	Percentage of recognition
1	Arabic Handwritten Characters Dataset (https://www.kaggle.com/mloey1/ahcd1) CNN+LSTM	Work of Abdulla Avloniy	55%

2	Arabic Handwritten Characters Dataset (https://www.kaggle.com/mloey1/ahcd1) CNN+CTC	Muhammadzade Khadizade	60%
3	Urdu Handwritten Characters Dataset https://www.kaggle.com CNN+LSTM	Work of Abdulla Avloniy	62%
4	Urdu Handwritten Characters Dataset https://www.kaggle.com CNN+CTC	Muhammadzade Khadizade	60%
CNN+LSTM+fuzzy CTC			
№	Book names	Trained series	Obtained results
1	Turkiy Guliston yohud ahloq	1500	97.86
2	Guliston	2000	95.82
4	Kadizade Maxmed	3000	97.18

We observed the difficulty of achieving full results using CNN, CNN+LSTM, by the model we also achieved 50-60 percent recognition results in language grammar. The CNN + CTC model had many shortcomings in word recognition. We learned that it is possible to increase the recognition percentage of CNN + LSTM + fuzzy CTC by using the membership function to extract their gradients when extracting the order of the points. We received some results. It is worth to note that the grammar of the language, the order of the words and the fact that the recommendations and evaluations are obtained in the upper class lead to the effectiveness of the results when the letters are recognized by the image.

3. Conclusions. The problem of segmenting such texts as a result of the combination of letters and words in Arabic graphics causes a number of problems. For this reason, several segmentations were used. Hybrid neuron model CNN + LSTM + fuzzy CTC was used to recognize image text using a number of text glyphs. This resulted in significantly higher recognition results than when we used the CNN + CTC and CNN + LSTM models. In CNN + CTC and CNN + LSTM model letters were read and identified them based on letter models. In doing so, the LSTM serves to shape the grammar of the language. The CTC uses a neural network to form the word match of an existing recognized image. Using CNN + LSTM in normal Latin and cyrillic graphic letters we achieved an effective results. Manuscripts in Arabic script have a low percentage of recognition. When segmentation of letters and words were carried out using CNN + LSTM + fuzzy CTC, the recognition rate increased by 90%. CNN + LSTM + fuzzy CTC hybrid neuron model proved to be effective for Arabic graphics.

REFERENCES

1. N. Tagougui and M. Kherallah, "Recognizing online arabic handwritten characters using a deep architecture," Proc.SPIE, vol. 10341, pp. 1– 5, 2017. [Online]. Available: <https://doi.org/10.1117/12.2268419>
2. N. Tagougui, H. Boubaker, M. Kherallah, and A. M. Alimi, "A hybrid NN/HMM modeling technique for online arabic handwriting recognition," CoRR, vol. abs/1401.0486, 2014. [Online]. Available: <http://arxiv.org/abs/1401.0486>

3. D. Keysers, T. Deselaers, H. A. Rowley, L. L. Wang, and V. Carbune, “Multi-language online handwriting recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1180–1194, June 2017.
4. V. Ghods and M. K. Sohrabi, “Online farsi handwritten character recognition using hidden markov model,” *JCP*, vol. 11, pp. 169–175, 2016.
5. Y. Hamdi, A. Chaabouni, B. Houcine, and A. M. Alimi, “Offlexicon online arabic handwriting recognition using neural network,” pp. 1–5, 2016.
6. W. Yang, L. Jin, Z. Xie, and Z. Feng, “Improved deep convolutional neural network for online handwritten chinese character recognition using domain-specific knowledge,” pp. 551–555, Aug 2015.
7. J. Zhang, J. Du, and L. Dai, “A gru-based encoder-decoder approach with attention for online handwritten mathematical expression recognition,” 12 2017.
8. X. Xiao, Y. Yang, T. Ahmad, L. Jin, and T. Chang, “Design of a very compact cnn classifier for online handwritten chinese character recognition using dropweight and global pooling,” 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), vol. 01, pp. 891–895, 2017.
9. X. Zhang, F. Yin, Y. Zhang, C. Liu, and Y. Bengio, “Drawing and recognizing chinese characters with recurrent neural network,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 849–862, April 2018.