

A Neural Based Segmentation and Recognition Technique for Handwritten Words

Author

Blumenstein, Michael, Verma, Brijesh

Published

1998

Conference Title

1998 IEEE World Congress on Computational Intelligence Proceedings

DOI

<https://doi.org/10.1109/IJCNN.1998.687119>

Copyright Statement

© 1998 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Downloaded from

<http://hdl.handle.net/10072/15703>

Link to published version

<http://ieeexplore.ieee.org/Xplore/dynhome.jsp>

Griffith Research Online

<https://research-repository.griffith.edu.au>

A Neural Based Segmentation and Recognition Technique for Handwritten Words

M. Blumenstein and B. Verma

School of Information Technology, Griffith University, Gold Coast Campus, Qld 9726, Australia

E-mail: {M.Blumenstein, B.Verma}@eas.gu.edu.au

Abstract

Artificial Neural Networks (ANNs) have been successfully applied to Optical Character Recognition (OCR) yielding excellent results. In this paper a technique is presented that segments difficult printed and cursive handwriting, and then classifies the segmented characters. A conventional algorithm is used for the initial segmentation of the words, while an ANN is used to verify whether an accurate segmentation point has been found. After all segmentation points have been detected another ANN is used to identify the characters which remain following the segmentation process. The C programming language, the SP2 supercomputer and a SUN workstation were used for the experiments. The technique has been tested on real-world handwriting scanned from various staff at Griffith University, Gold Coast. Some preliminary experimental results are presented in this paper.

1. Introduction

Artificial Neural Networks have proven to be successful in many areas of pattern recognition [1], [2], [3], [4]. Many researchers have also applied the excellent generalisation capabilities offered by ANNs to the recognition of characters [5], [6]. Some researchers have used conventional methods for segmentation and recognition [7], while others have used ANN based methods for the character recognition process [8]. However, there have only been a handful of researchers using ANNs for the segmentation of printed and cursive handwriting [9], [10] followed by the subsequent recognition of characters. As is mentioned in [11], [12], segmentation plays an important role in the overall process of handwriting recognition. Unfortunately, not only is it a vital process but it is also one that has not achieved very accurate results.

This research attempts to integrate both conventional and intelligent methods for the segmentation of difficult printed and handwritten words, followed by the accurate recognition of characters. For the task of segmentation,

a simple heuristic segmentation algorithm is used which finds segmentation points in printed and cursive handwritten words. A neural network trained with valid segmentation points from a database of scanned, handwritten words is used to assess the correctness of the segmentation points found by the algorithm [13].

Following segmentation and verification, the resulting characters are then identified by another neural network. A certain number of characters are used for first training the neural network while the remainder of characters are used to test the neural network's generalisation capabilities. After the characters have been identified a postprocessing technique using a dictionary of words could be used for final word classification.

The remainder of the paper is broken down into 4 sections. Section 2 briefly describes the proposed techniques and algorithm, Section 3 provides experimental results, a discussion of the results follows in Section 4, and a conclusion is drawn in Section 5.

2. Proposed techniques

The following section addresses the steps that were required to preprocess and segment handwritten words using the aforementioned algorithm and an ANN. Also included in this section is the recognition process of handwritten characters after segmentation of whole words. The techniques used were: 1. Binarisation, 2. Segmentation using a heuristic algorithm, 3. Manual Segmentation, 4. Preprocessing, 5. Training of the ANN, 6. Testing the heuristic algorithm, 7. Extraction of individual characters from handwritten words, 8. Character training file creation, 9. Training of ANN and recognition of handwritten characters. An overview of the segmentation process for Steps 5 and 6 is provided in Figure 1.

2.1 Binarisation

After the images were acquired, they were converted into monochrome bitmap (BMP) form. Before any segmentation or processing could take place, it was

then necessary to convert the images into binary representations of the handwriting. A method previously used in [14], was employed for this purpose. At this stage, the handwriting could be used for processing in further steps.

2.2 Segmentation using a heuristic algorithm

A simple heuristic segmentation algorithm was implemented which scanned handwritten words for important features to identify valid segmentation points between characters. The algorithm first scanned the word looking for minimas or arcs between letters, common in handwritten cursive script. In many cases these arcs are the ideal segmentation points, however in the case of letters such as “a”, “u” and “o”, an erroneous segmentation point could be identified. Therefore the algorithm incorporated a “hole seeking” component which attempted to prevent invalid segmentation points from being found.

If an arc was found, the algorithm checked to see whether it had not segmented a letter in half, by checking for a “hole”. Holes, are found in letters which are totally or partially closed such as an “a”, “c” and so on. If such a letter was found then segmentation at that point did not occur. Finally, the algorithm performed a final check to see if one segmentation point was not too close to another. This was done by ascertaining if the distance between the last segmentation point and the position being checked was equal to or greater than the average character width of a particular word. If the segmentation point in question was too close to the previous one, segmentation was aborted. Conversely, if the distance between the position being checked and the last segmentation point was greater than the average character width, a segmentation point was forced. The heuristic algorithm is explained in [13].

2.3 Manual segmentation

To train the ANN with both accurate and erroneous segmentation points, the output from the heuristic segmentation algorithm was used. It was necessary to manually separate the coordinates provided by the algorithm into “good” and “bad” segmentation categories and save them to a file. These coordinates were then processed by a further step to produce a training file for the ANN.

2.4 Preprocessing

A preprocessing program was used to locate the coordinates produced in the previous step and to extract a segmentation point. The program first searched each binarised handwritten word, locating the appropriate

segmentation points to be included for training. Each segmentation point was then extracted from the binarised word and included in a training file. The width of the segmentation point used for training was 5 pixels, and the height depended on the height of the words used for training (The word with the greatest height was used as the segmentation point height). Also included in the training file were desired outputs for each segmentation point. 0.9 was used to represent an accurate segmentation point and 0.1 was used for erroneous segmentation points.

2.5 Training of the ANN

For this step, a multi-layer feedforward Neural Network trained with the backpropagation algorithm was used. The ANN was presented with the training pairs found in the previous step. Some of the successful structures of the ANN are shown in Table 1.

2.6 Testing the heuristic algorithm

After training of the ANN, handwriting used for testing was applied to the heuristic segmentation algorithm. Again, many segmentation points were found (some of which were incorrect). All segmentation points were then presented to the ANN to verify the accuracy of the segmentation points output by the heuristic algorithm. The ANN presented output indicating which segmentation points were correct and which were incorrect. Finally, only correct segmentation points were left and extra, non-legitimate points were removed.

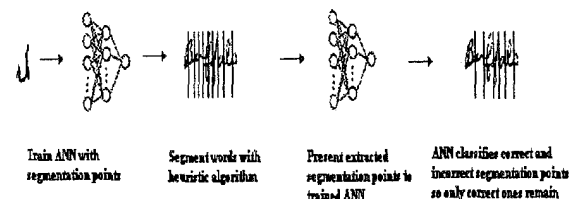


Figure 1. Segmentation Process

2.7 Extraction of individual characters from handwritten words

Following the identification of valid segmentation points in the handwritten words used for experimentation, individual characters were extracted. This was achieved by beginning a search at the origin of each word and looking for a segmentation point. Once a segmentation point was found, the first character was subsequently extracted. Subsequently the end of the character was set to the starting point of the next character and a search for the next segmentation point ensued and so on. This was repeated for all words.

2.8 Character training set creation

After all characters were extracted from all words, a raw character set was the result. This raw set required some preprocessing before a training set and test set could be used with the ANN. A simple algorithm found all characters which seemed to be too large. For example, as a result of segmentation error, there were still a small number of characters which were not properly segmented in Section 2.2. Secondly, there was a manual scan of the raw character file to ensure further accuracy of inputs. Lastly, a very simple normalisation algorithm was used to set all character matrices to the same size. The largest character was found in the set, and all other characters, smaller in size were padded with zeros vertically and horizontally. It was then possible to prepare two separate files used to train and test the ANN.

2.9 Training of ANN, recognition of handwritten characters

A neural network was trained with the data described in Section 2.8. Many experiments were conducted varying the settings and number of iterations to provide the optimum results. The most successful settings are displayed in Table 4. Following training, the ANN was presented with characters to test its generalisation capabilities. The results for segmentation and character recognition are presented in the following section.

3. Experimental results

3.1 Handwriting database

For preliminary experimentation of the technique detailed in Section 2, we used samples of handwriting from various subjects from Griffith University. Some examples are shown below in Figure 2.

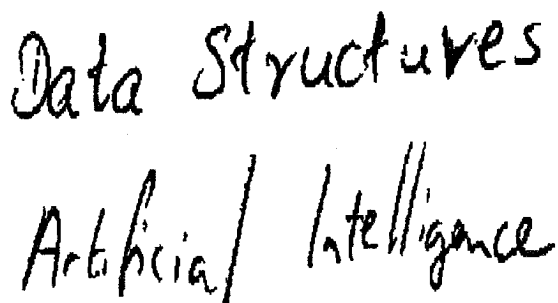
The image shows two lines of handwritten text in cursive. The first line reads 'Data Structures' and the second line reads 'Artificial Intelligence'. The handwriting is fluid and appears to be from a student or researcher.

Figure 2. Handwriting Samples Used for Training/Testing

3.2 Implementation and experimentation of the segmentation technique

Implementation and experimentation of the segmenter were performed on the SP2 Supercomputer at Griffith University, Brisbane. After implementation, the heuristic segmentation system in conjunction with the ANN was trained and tested on the scanned words mentioned in Section 3.1. The most successful settings for the ANN can be seen in Table 1. The settings which remained constant through all experiments included: learning rate and momentum, both set to 0.2, and the number of outputs which was 1. Experimental results for segmenting person number one's handwriting are presented in Table 2. Experimental results for segmenting person number two's handwriting are presented in Table 3.

Table 1. Settings for the ANN

Experiment #	# of Inputs	Hidden Units	# of Iterat.
1	430	25	300
2	496	15	300

Table 2. Results for 1st person

	Words	# of Seg. Points	Class. Rate [%]
Testing	Agnes	24	91.67
	Brijesh	27	85.19
	Comp. Sci.	40	82.5
	Neural Net	37	83.78

Table 3. Results for 2nd person

	Words	# of Seg. Points	Class. Rate [%]
Testing	intelligent	38	92.11
	segment.	49	87.76
	system	40	90.00
	technique	41	75.61

3.3 Implementation and experimentation of character recognition technique

Implementation and experimentation of the character recognition phase was also conducted on the SP2 Supercomputer. Many experiments were conducted, and the settings for the ANN producing the best results are presented in Table 4. Settings which remained constant throughout all experiments were again learning rate and momentum, which were set to 0.2. Also the number of outputs was constantly 26. Character Recognition experiments for characters extracted by the process described in Section 2, are presented in Table 5.

The aforementioned table displays character recognition results from person one, two and a combination of both.

Table 4. Settings for character recognition experiments

Experiment #	No. of Inputs	Hidden Units	No. of Iterat.
1	900	20	400
2	810	30	1500
3	900	30	500

Table 5. Character recognition experiments for person 1 and 2

Person #	# of words used for testing	Classification Rate [%]
1	17	58.82
2	27	74.07
1&2	44	63.64

4. Discussion of results

4.1 Heuristic segmentation

The segmentation program over-segmented words it was presented with. This allowed the ANN to then discard improper segmentation points and leave accurate segmentation points. Overall the whole process was very successful, however some limitations still exist.

Due to the fact that some words were not even legible by humans, it was excepted that in some cases the heuristic algorithm would not find enough segmentation points for use in further steps. This limitation shall be addressed by improving the algorithm and detecting or in some case ignoring more features to allow for more prospective points to be found.

The second limitation refers to the varying sizes of words input to the system. As the heuristic segmenter has a fixed character size threshold which is essential to the segmentation process for totally cursive writing, words containing letters which stray from the average size may generate problems. Further improvements to the algorithm will include a dynamic threshold to deal with varying character and word sizes.

4.2 Classification of segmentation points

The experiments conducted using our segmenter were preliminary, and did not allow for extensive exploration of various parameter settings. In the first experiment, using person one's handwriting, the ANN producing most successful results contained 25 hidden units and was trained for 300 iterations. For person two's handwriting, the best results were achieved with an ANN containing 15 hidden units and trained for 300 iterations.

Although many experiments were conducted, future experiments may allow for further exploration of different parameter settings.

Considering the difficulty of some of the words used for testing and training, the results produced were very high. As could be seen in Table 2 and 3, the ANN gave very high results for verifying segmentation points in words used for testing (segmentation points the ANN had never seen before). On closer inspection it was found that the points that were incorrectly classified were in fact very ambiguous points in some of the difficult cursive handwritten words. In further experiments a much larger benchmark database of segmentation points shall be used for experimentation. This will allow the ANN to be exposed to a much greater range of correct and incorrect segmentation points, hopefully reducing ambiguity.

4.3 Classification of segmented characters

The experiments conducted classifying segmented characters produced some very promising results. It is important to note that after the characters in a word are segmented, some of the characters may sometimes be distorted, include extra fragments from other characters, or are truncated due to the segmentation process. Taking into account the difficulty of the classification process of these distorted and sometimes incomplete character the results obtained for person one's handwriting, the combined character set and especially the results obtained for person two's handwriting are very promising.

4.4 Comparison of results for segmentation points

As mentioned earlier, many researchers have used various techniques for the segmentation of characters in handwritten words. Segmentation accuracy rates of above 90% were achieved by Lee et al. [15], however the authors were only dealing with printed alphanumeric characters. Srihari et al. [11] obtained segmentation accuracies of 83% for handwritten zip codes (no alphanumerics). Finally, experiments conducted by Eastwood et al. [10], segmenting cursive handwriting produced a 75.9% accuracy rate. The authors used an ANN-based technique for segmentation with 100,000 training patterns. On average our segmentation accuracy for both preliminary experiments was just over 86%. Although our experiments were only preliminary, our results compare favorably with those of other researchers. In further work a much larger database of segmentation points shall be used for training the ANN, which should increase segmentation accuracy rates even further.

4.5 Comparison of results for segmented characters

Following on from section 4.4, it is possible to compare the results obtained in this research to those of other researchers classifying segmented characters. Srihari et al. [11] obtained a recognition rate of 63% for handwritten cursive characters. Yanikoglu and Sandon [8], achieved recognition rates of 50% for the recognition of letters from cursive text. For our experiments, averaging the results obtained for person one and two, we obtain a 66.45% recognition rate. Our combined database generated a 63.64% recognition rate. As can be seen our results compare favorably with those obtained by other researchers.

5. Conclusions

A heuristic segmentation and recognition technique using two multi-layered feedforward neural networks has been presented. Preliminary experiments were conducted on real-world handwritten words. The ANN-based segmentation technique produced very good results using a preliminary handwriting database. After segmentation the individual characters were used to train and test a further neural network. The classification of letters also produced some very encouraging results. This research is still ongoing, and many improvements and additions to preprocessing and postprocessing techniques shall be explored. Finally, a larger handwriting database shall be used in future experiments to show the full potential of the proposed technique.

References

- [1] A. Khotanzad, and J. Lu, "Shape and Texture Recognition by a Neural Network", *Artificial Neural Networks in Pattern Recognition*, Sethi, I. K., Jain, A. K., Elsevier Science Publishers B. V., Amsterdam, Netherlands, 1991, pp. 109-131.
- [2] B. Zheng, W. Qian, and L. Clarke, "Multistage Neural Network for Pattern Recognition in Mammogram Screening", *IEEE ICNN, Orlando*, 1994, pp. 3437-3448.
- [3] A. D. Kulkarni, *Artificial Neural Networks for image understanding*, Van Nostrand Reinhold, New York, 1994.
- [4] K. Han, and I. K. Sethi, "Handwritten Signature Retrieval and Identification", *Pattern Recognition Letters*, 17, 1996, pp. 83-90.
- [5] C. Y. Suen, and R. Legault, C. Nadal, M. Cherier, and L. Lam, "Building a New Generation of Handwriting Recognition Systems", *Pattern Recognition Letters*, 14, 1993, pp. 305-315.
- [6] S-W. Lee, "Off-Line Recognition of Totally Unconstrained Handwritten Numerals Using Multilayer Cluster Neural Network", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18, 1996, pp. 648-652.
- [7] R. M. Bozinovic, and S. N. Srihari, "Off-Line Cursive Script Word Recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11, 1989, pp. 68-83.
- [8] B. A. Yanikoglu, and P. A. Sandon, , 1993, Off-line cursive handwriting recognition using style parameters, *Technical Report PCS-TR93-192*, Dartmouth College, NH.
- [9] G. L. Martin, M. Rashid, and J. A. Pittman, "Integrated Segmentation and Recognition through Exhaustive Scans or Learned Saccadic Jumps", *International Journal of Pattern Recognition and Artificial Intelligence*, 7, 1993, pp. 187-203.
- [10] B. Eastwood, A. Jennings, and A. Harvey, "A Feature Based Neural Network Segmenter for Handwritten Words", *International Conference on Computational Intelligence and Multimedia Applications (ICCIMA '97)*, Gold Coast, Australia, 1997, pp. 286-290.
- [11] S. N. Srihari, "Recognition of Handwritten and Machine-printed Text for Postal Address Interpretation", *Pattern Recognition Letters*, 14, 1993, pp. 291-302.
- [12] M. Gilloux, "Research into the New Generation of Character and Mailing Address Recognition Systems at the French Post Office Research Center", *Pattern Recognition Letters*, 14, 1993, pp. 267-276.
- [13] M. Blumenstein, and B. Verma, "An Artificial Neural Network Based Segmentation Algorithm for Off-line Handwriting Recognition", *International Conference on Computational Intelligence and Multimedia Applications (ICCIMA '98)*, Melbourne, Australia, 1997, (Accepted).
- [14] M. Blumenstein, and B. Verma, "A Segmentation Algorithm used in Conjunction with Artificial Neural Networks for the Recognition of Real-World Postal Addresses", *International Conference on Computational Intelligence and Multimedia Applications (ICCIMA '97)*, Gold Coast, Australia, 1997, pp. 155-160.
- [15] S-W. Lee, D-J. Lee, H-S. Park, "A New Methodology for Gray-Scale Character Segmentation and Recognition", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 18, 1996, pp. 1045-1051.