



RECOGNIZING HANDWRITTEN MATHEMATICAL EXPRESSIONS

Pallavi Ratra

Department of Computer Science
Manipal University Jaipur
Jaipur, Rajasthan, India

Harsh Khanna

Department of Computer Science
Manipal University Jaipur
Jaipur, Rajasthan, India

Abstract This paper surveys the techniques to recognise handwritten mathematical characters in different expressions and to see the availability of systems that eventually understand and solve them. Humans are accustomed to writing mathematical expressions containing integrals, fractions, exponents or indices by hand. Entering each such expression into a computer is uncomfortable and tedious. Inspired by recent success in Neural Network Modelling, we discuss a proposed model that gives high accuracy in recognizing mathematical symbols and digits in different mathematical expressions using Neural Network Modelling for identification of offline handwritten expressions.

Keywords Mathematical expression recognition · Artificial neural networks · Offline character recognition

I. INTRODUCTION

Handwriting Character Recognition is a very captivating and challenging topic of research in image processing and pattern recognition. Several research works have been focusing on different methodologies in an attempt to reduce the processing time while simultaneously improving recognition accuracy. In general, handwriting recognition can be classified into two major categories, namely off-line and on-line handwriting recognition methods. In the off-line recognition method, the writing is normally captured optically using a scanner and the whole expression is presented as an image. In the on-line system, the 2D coordinates of successive points are indicated as a

function of time and the order of lines or strokes made by the individual is available as well.

The steps in any handwritten recognition system are pre-processing followed by segmentation, feature extraction and classification. Neural networks pick up from an image training set. The trained network then performs the character identification. Each neural network learns different things from the properties of different training images. It then looks for common properties in the test image to be classified. [1]

For centuries, writing things had been the most common way of communication. Nowadays, computers and the Internet are the indispensable way of modern communication; turning the world into a tiny city. However, writing things is still significant. Mathematics is widely used in almost all fields of science, such as physics, engineering, medicine, economics, etc. And this is the reason that mathematical expressions need to be mentioned in scientific documents. Many tools are available to achieve this task. However, most of those tools require some expertise to use them efficiently. Latex and MathML, for example, require knowledge of predefined sets of keywords to describe special mathematical symbols and functions in addition to spatial layouts. Other tools, such as Math Type, depend on a visual environment to add symbols using the mouse and require a lot of time. [7]

In the following sections, we discuss the theory behind offline character recognition and explore the literature review to study the various techniques for the same.



A. Algorithms and techniques

Artificial Neural Networks

Artificial neural networks (ANNs) are computing systems inspired by the biological neural networks that constitute animal brains. Such systems learn (progressively improve performance on) tasks by considering examples, generally without task-specific programming. [14] The original goal of the neural network approach was to solve problems in the same way that a human brain would. [15] An ANN contains a collection of nodes called artificial neurons. Each artificial neuron is connected to the other to transmit a signal from one to another.

Neurons and their connections typically have a weight that adjusts as learning proceeds, and may also have a threshold value. Typically, neurons are organized in layers. Signals travel from the first (input) to the last (output) layer, possibly after traversing the different layers, multiple times. [16]

B. Detailed Methodology

The process of character recognition consists of many phases such as pre-processing, segmentation, feature extraction, classification and recognition. The output of one step is the input of the next step. The task of pre-processing the dataset relates to the removal of noise and variation in handwriting.

Digitization: Digitization is the process of converting a paper-based handwritten document into an electronic format. The electronic conversion is accomplished by using a method whereby a document is scanned and an electronic representation of the original document of an image file format is produced.

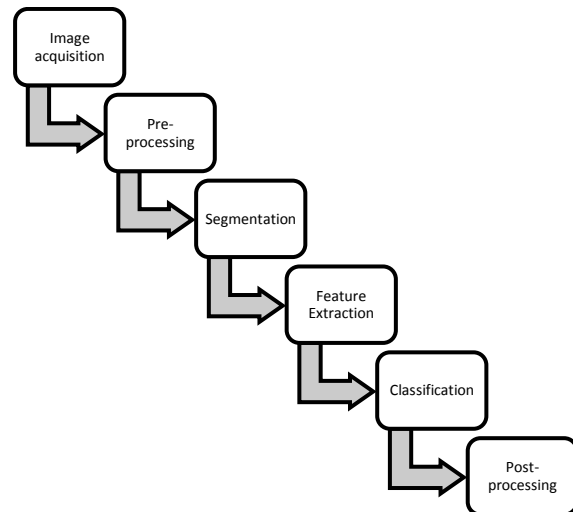


Figure 1 Methodology

Pre-processing: In the pre-processing phase, a series of operations are performed on the scanned input image. It enhances the image, making it suitable for segmentation. The grey-level character image is normalized into a window size. After noise reduction, a bitmap image is produced. Then the bitmap image is transformed into a thinned image.

Skew Correction: Images clicked through a camera may suffer from skew and perspective bias. These text regions are de-skewed using computationally efficient and fast skew correction techniques.

Binarization: A skew corrected text-region is binarized using a simple yet efficient binarization technique. The eight immediate neighbours around the pixel subject to binarization are also taken as deciding factors for binarization.

Segmentation: Segmentation is done by separation of the individual characters of an image. Segmentation of handwritten characters into different zones (upper, middle and lower zone) and characters is more difficult than that of printed documents that are in standard form. This is mainly because of variability in the paragraph, words of line and characters of a word, skew, slat, size and curved.

Feature Extraction: In this phase, the features of individual characters are extracted. The performance of each character recognition system depends on the features that are extracted. The extracted features from the input characters should allow classification of a character in a unique way.



After this, the **dataset is trained** using a certain model.

The dataset is then **tested** through experiments and results are recorded.

The results are studied and a **conclusion** is drawn. [2] [3]

II. LITERATURE REVIEW

J. Pradeepa, E. Srinivasan, S. Himavathi (2012) [1] identified the most suitable Neural Network for the design of hand written English character recognition system and compared techniques like back propagation neural network, nearest neighbour network and radial basis function network. In contrast to the General Offline character recognition method which involves Image acquisition, pre-processing, segmentation, feature extraction, classification and post processing, the method proposed combines the process of feature extraction and classification. Template matching, though easily implemented does not yield good result, hence Neural networks are used and are implemented using MATLAB. The image is taken as an input and is sent to all three types of NN-feed forward back propagation neural network, nearest neighbour network and radial basis function network which yield 94.15%, 86.96%, 89.42% accuracy respectively. The proposed system is found to be less complex and allows faster recognition of characters. All the different classifiers have been trained with 200 sets of data and extensively tested. To solve the English character recognition problem with interference of external noise, a new approach based on feature combination and BP network is presented by Yang Yang, Xu Lijia, Cheng Chen (2011) [4] by extracting the structural features and the statistical features from the English characters. This way, common problems like illumination variance, noise, complex and dirty background resulting in low recognition accuracy are being solved. Feature extraction is the core of a recognition system, it is further divided into- Statistical features and structural features. For statistical features, each character is depicted by a 5 X 7 matrix and then it is normalized and a statistical feature vector is created. Structural Features are used to show character structure. Characters are recognised with the help of some basic features like number of 90° angles, discontinuous curve, horizontal and vertical lines along with oblique line. Each character is designated a set for such values which makes the character recognition easier. Features from both the Statistical and Structural analysis are used in this paper and they are directly combined and sent to train the BP network. This

method shows superior recognition capability and good convergence speed while compared to other features classified by BP network. Another technique involves a complete OCR system for camera captured image/graphics embedded textual documents for handheld devices to counter problems like skew and perspective distortion, manual involvement in the capturing process, uneven and insufficient illumination, and unavailability of sophisticated focusing system. In this methodology at first, text regions are extracted and skew corrected. Then, these regions are binarized and segmented into lines and characters. Characters are passed into the recognition module. Ayatullah Faruk Mullah *et al.* (2011) [2] subdivided the characters into 62 classes which were based on similar looking characters, an accuracy of 93% was observed whereas for 73 classes, accuracy was 83%. Dan Claudiu Cireşan *et al.* (2011) [5] propose a model in which the nets have an input layer of 29×29 neurons followed by a convolution layer with 20 maps of 26×26 neurons and filters of size 4×4 . The next hidden layer is a max-pooling layer, with a 2×2 kernel which has its outputs connected to another convolution layer containing 40 maps of 9×9 neurons each. The last max-pooling layer is reducing the map size to 3×3 by using filters of size 3×3 . A fully connected layer of 150 neurons is connected to the max-pooling layer. The output layer has one neuron per class. The images from the training set were distorted at the beginning of every epoch (reference set) by a fixed value. Preprocessing done on the dataset depends on the writing style resulting in different aspect ratios of different characters. Before training, the width of all characters is normalized to 10, 12, 14, 16, 18 and 20 pixels. For each of the preprocessed or original datasets, five differently initialized CNNs are trained for the same number of epochs (reference set). Two handwritten character datasets are used - subsets from NIST SD 19 and digits from MNIST. The results show that error rates for digits are significantly lower than those for letters. Training nets with case-insensitive letter labels makes error rates drop considerably. A generic letter recognizer should therefore be trained on a merged letter dataset. All experiments use the same net architecture and deformation parameters. Further tests show that the distortion parameters used in this experiment are too big and using 20% lower values would decrease error rates by another 1.5%. Francisco Álvaro, Joan-Andreu Sánchez, José Miguel Benedí (2014) [6] assess the performance of three well-known off-line feature extraction techniques - Pattern Recognition and Human Language Technologies (PRHLT), FKI (well-known set of geometric features)



and RWTH. It also reports and analyses the results of using recurrent neural networks for mathematical symbol recognition. Mathematical expression recognition is commonly divided into three primary tasks: symbol segmentation, symbol recognition and structural analysis. The data set used for experimentation is - dataset of the last CROHME Competition. The paper reports experiments with online features, offline features and their combination. The advanced architecture Bidirectional Long Short-Term Memory (BLSTM) for RNN is known to have outperformed other classifiers in handwritten text recognition based on previous studies and therefore this classifier is used in the experiments. RNNs are a connectionist model containing a self-connected hidden layer. The combination of bidirectional RNNs and the LSTM architecture results in BLSTM-RNNs. These networks have outperformed standard RNNs in handwriting text recognition. Results show that offline features provided better results than online features, and their combination produced up to 20% relative improvement. FKI offline features presented the best performance as they obtained the best recognition rates using only 9 features.

Ahmad-Montaser Awal, Harold Mouchère, Christian Viard-Gaudin (2009) [7] propose a global learning method instead of using pre-trained classifiers. First, the symbol classifier will be trained from scratch in the context of the whole system. Different objective functions have been considered, including maximum likelihood (ML) and maximum mutual information (MMI). Comparing the results with previous studies which used a layered search framework for mathematical expression recognition performing a simultaneous segmentation and recognition, but using a classifier which is trained on isolated symbols. In the model proposed, both expression recognition and system training can be described with the same global architecture of expression recognizer system. Similarly, a global learning method to train the classifier directly from mathematical expressions instead of using a pre-trained recognizer from a set of isolated symbols is used. The model has a multi-layer perceptron neural network (MLP) as a classifier and uses a gradient based backpropagation algorithm. Experiments were carried out with a MLP architecture, using 100 neurons in the hidden layer and seven local features for every re-sampled points, resulting in a 210-dimension input vector. The dataset was a combination of synthetically prepared expressions and some real expressions. The results showed a large improvement when comparing the use of the classifier trained on the isolated symbols with the classifier trained in the global system. Nei Kato,

Masato Suzuki, Shin'ichiro Omachi, Hiroto Aso, Yoshiaki Nemoto (1999) [8] discussed four major procedures which include pre-processing, rough classification, fine classification and feature vector extraction. Pre-processing helps remove possible negative influence from image distortion. In pre-processing, normalization, smoothing and transformation based on partial inclination detection is done. TPID resolve partial inclination whereas normalization and smoothing brings the image to 64X64 dots and patches up the holes respectively. After pre-processing, Feature vector extraction takes place which involves contour extraction, dot orientation and vector construction. Since the number of Kanji and Kana is very large therefore, discrimination is done in two parts- Rough classification and fine classification. Rough classification is done using City Block Distance with Deviation (CBDD), the error rate of which is 4.34%. CBDD is the most effective discriminant function, the average accumulated recognition rate of 30 candidates is 99.86%. Fine classification is done using Asymmetric Mahalanobis Distance (AMD) because it can express distributions of images with a small number of parameters. AMD is derived by a probability density function of multivariate normal distribution and involves the number of dimensions of feature vector and the number of samples. It is implemented using C language on Sun Ultra 2. A new approach is used known as Watch, Append and Parse which is a novel end to end approach based on neural network was proposed by Jianshu Zhang *et al.* (2017) [9]. It learns to recognize handwritten mathematical expression in a two dimensional layout and outputs it in a one dimensional character sequences in LaTeX format which is a document preparation system. The problem of symbol recognition and structural analysis is handled by a watcher and a parser respectively. A convolutional neural network encoder is used to take input as the watcher and a recurrent neural network with an attention mechanism is used to decode it to generate output in LaTeX sequences. This new approach outperformed the older accuracy on the same dataset and gave an expression accuracy of 46.55% CROHME 2014 and 44.55% in CROHME 2016.

Hans-Jiirgen Winkler [10] proposed symbol recognition system based on Hidden Markov Models and different kinds of feature extraction algorithms are used for analysing on-line features as well as off-line features and combining the classification results. It is based on the soft-decision approach which is a decoding method within the symbol segmentation and recognition stage of the overall system. The dataset



contains only uppercase, lowercase alphabets and three symbols. It does not deal with widely used mathematical operators like plus, multiplication etc. Another method is which captures all recognizable interpretations of the input and organizes them in a parse forest from which individual parse trees may be extracted and reported. If the top-ranked interpretation is incorrect, the user may request alternates and select the recognition result they desire. The tree extraction step uses a novel probabilistic tree scoring strategy in which a Bayesian network is constructed based on the structure of the input, and each joint variable assignment corresponds to a different parse tree. Parse trees are then reported in order of decreasing probability. In this method devised by Scott MacLean, George Labahn (2015) [11] the work is done in the context of MathBrush pen-based math system. Using MathBrush, one draws an expression, which is recognized. Once the expression is completely recognized, it is embedded into a worksheet interface through which the user may manipulate and work with the expression via computer algebra system commands. A relatively new method has come up which is able to work against the difficulties and is able to recognize handwritten mathematical expression in better manner. This system given by Yassine Chajri, Belaid Bouikhalene (2016) [12] is based on four necessary steps: the first being the pre-processing techniques (normalization, filtering, binarization and skew detection and correction). The second and third steps are respectively: the expressions segmentation (connected component algorithm) and the features extraction (radon transform). The last step is the symbols classification (support vector machines) which gives an accuracy of 98%. A rather different approach given by Yusuke Takiguchi, Minoru Okada, Yasuji Miyake [13] is being followed in which the first step is to pre-process the raw input image so that it can be handled better in later steps. The second task is to perform segmentation on the image to have all pixels for one symbol in a set. The next step is to figure out the location and size of each single symbol with the information provided by the previous segmentation process. Then, a classifier is needed to recognize each symbol. At last, there should be some parser which in cooperate both the location and classification information to get the final result. For the parser, the work has not been started because the accuracy given after classification was not good.

III. CONCLUSION AND DISCUSSION

This review was done after surveying over 25 research papers. We found that the following techniques involving neural networks give us suitable results for the recognition of handwritten mathematical expressions. The Feed Forward Neural Network resulted in an accuracy of 94.15% [1]. The Nearest Neighbor Neural Network resulted in an accuracy of 86.96% [1]. The Radial Basis Neural Network resulted in an accuracy of 89.42% [1]. With Standard Template matching, the accuracy of 54.27 was achieved [1]. Structural and Statistical features with backpropagation network [4] gave very little or no error, the exact results not being mentioned in the research paper. Convolutional neural network [5] resulted in very high character recognition accuracy, 99% in some cases. The Bidirectional Long short term memory RNN (BLSTM-RNN) [6] resulted in high accuracy for mathematical symbol recognition. Multi-layered Perceptron Neural Network [7] resulted in an accuracy of 37.1% for complete mathematical recognition. Using a Convolution Neural network, which gives good accuracy for character recognition and the Bidirectional Long short term memory RNN (BLSTM-RNN), which gives good accuracy for mathematical symbol recognition, if implemented together can give good accuracy for complete mathematical expression recognition. The result of using this combination can then be compared with the results of using the Multi-layered Perceptron Neural Network which gives a fair accuracy for complete mathematical expression recognition.

IV. REFERENCES

- [1] Pradeepa J., Srinivasan E., Himavathi S., "Neural Network Based Recognition System Integrating Feature Extraction and Classification for English Handwritten" in IJE TRANSACTIONS B: Applications Vol. 25, No. 2, (May 2012) 99-106
- [2] Faruk Mollah Ayatullah, Majumder Nabamita, Basu Subhadip and Nasipuri Mita, "Design of an Optical Character Recognition System for Camera-based Handheld Devices", in IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 4, No 1, July 2011
- [3] Er. Bhatia Neetu, "Optical Character Recognition Techniques: A Review", in International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 5, May 2014
- [4] Yang Yang, Lijia Xu, Chen Cheng, "English Character Recognition Based on Feature



combination.” in 2011 International Conference on Advances in Engineering

[5] Claudiu Cireşan Dan and Meier Ueli and Maria Gambardella Luca and Schmidhuber Jürgen, “Convolutional Neural Network Committees for Handwritten Character Classification” in 2011 International Conference on Document Analysis and Recognition

[6] Álvaro Francisco, Sánchez Joan-Andreu, Miguel Benedí José, “Offline Features for Classifying Handwritten Math Symbols with Recurrent Neural Networks” in 2014 22nd International Conference on Pattern Recognition

[7] Awal Ahmad-Montaser, Mouchère Harold, Viard-Gaudin Christian, “Towards Handwritten Mathematical Expression Recognition”, in 2009 10th International Conference on Document Analysis and Recognition

[8] Kato Nei, Suzuki Masato, Omachi Shin'ichiro, Aso Hiroto, Nemoto Yoshiaki, “A Handwritten Character Recognition System Using Directional Element Feature and Asymmetric Mahalanobis Distance”, in IEEE Transactions On Pattern Analysis and Machine Intelligence, VOL. 21, NO. 3, MARCH 1999

[9] Zhang Jianshu, Du Jun, Zhang Shiliang, Liu Dan, Hu Yulong, Hu Jinshui, Wei Si, Dai Lirong, “Watch, attend and parse: An end-to-end neural network based approach to handwritten mathematical expression recognition”, in Pattern Recognition 71 (2017) 196–206

[10] Winkler Hans-Jiirgen, “HMM-based handwritten symbol recognition using on-line and off-line features” Unpublished report

[11] MacLean Scott, Labahn George, “A Bayesian model for recognizing handwritten mathematical expressions”, in PatternRecognition48(2015)2433–2445

[12] Chajri Yassine and Bouikhalene Belaid, “Handwritten Mathematical Expressions Recognition” in International Journal of Signal Processing, Image Processing and Pattern Recognition Vol.9, No.5 (2016), pp.69-76

[13] Takiguchi Yusuke, Okada Minoru, Miyake Yasuji, “A Fundamental Study of Output Translation from Layout Recognition and Semantic Understanding System for Mathematical Formulae”, Unpublished report

[14] Yang Ziji, Liu Youwu, “Investigating the influential factors on firefighter Injuries using statistical machine learning” in Proceedings of the 2018 International Conference on Machine Learning and Cybernetics, Chengdu, China, 15-18 July, 2018

[15] Vasuki A. and Govindaraju S., edited by Hemanth D.J., Vieira Estrela V., “Deep Learning for Image Processing Applications” Page 28

[16] Website - <https://www.dataversity.net/artificial-neural-networks-overview/>