Recognition of Offline Handwritten Mathematical Expressions

Manisha Bharambe
Research scholar
Yashwantrao Mohite College,
Bharati Vidyapeeth Deemed University, Pune

ABSTRACT
Recognition of Handwritten Mathematical Expression (ME) is one of the most fascinating and challenging research area in the field of Image Processing and Pattern Recognition. The recognition of handwritten mathematical expression is difficult due to variability of the symbols in an expression and its two Dimensional structure. This paper deals with the recognition of handwritten logical mathematical expressions. The strength of the proposed approach is efficient preprocessing, feature extraction and segmentation methods .

Keywords
Pattern recognition, Handwritten logical Mathematical Expression, Preprocessing, Feature extraction, segmentation

1. INTRODUCTION
Since 1950, the recognition of handwritten mathematical expressions has been popular research area. Mathematical expression recognition is an important problem about pattern recognition, because it is an essential part of scientific literature[11]. Mathematical symbols set is very huge about 2000 symbols, so commonly used keyboard input is not sufficient. Also mathematics inputting and editing have been difficult due to their non-linear structure. Mathematical symbols can be written beside, above, or below in different sizes, fonts, typefaces, and font sizes used [20]. In recent years research towards recognition of handwritten mathematical symbols and expressions is getting increasing attention. The researchers proposed many tools to recognize mathematical symbols and expressions. Most of those tools require some expertise to use them efficiently. MathML and Latex [21] requires knowledge of predefined sets of keywords. A number of commercially successful products are available which recognize a users natural handwriting to perform simple tasks. Commercial program can not handle mathematical formulas. Many work with multicolumn variation do not work well unlike straight text. LATEX when used for writing mathematical formulae which involves searching through texts or long pdf files to find the command which corresponds to a desired symbol. New user can not guess command for symbol. Also it works on typeset mathematical formulae. Infty and Maple are used with online handwritten character recognition system. Infty editor computes online expression and give results. Infty Reader is used to recognized scanned images of clearly printed mathematical documents and outputs the recognition result in a XML format. The XML file output can be converted into LATEX, MathML, HTML for English text. Mathbrush is a commercial tool used for online mathematical symbol recognizer. Maple is a commercial software for computer algebra and works with online recognition of expression. MathType requires many mouse movements to add a symbol and needs lot of time. Since no tool available to recognize OCR handwritten mathematical logical expression, its recognition is a requirement. The problem of recognizing handwritten mathematics is significantly different from natural language recognition. There is no pre-defined context which gives constraints to possible symbols. Two dimensional structure together with implicit operators makes mathematical expression recognition a challenging problem.

2. LITERATURE REVIEW
In 2010, Ahmad-Montaser Awal et al[2] discussed some issues related to the problem of online mathematical expression recognition. The very first important issue is to define how to ground truth a dataset of handwritten mathematical expressions, and the next problem is that of benchmarking systems. This paper states that MEs must not be ground-truthed only by their content but foremost with their displayed symbols and their layout. The main difference between different ME recognition systems is the expression databases. Unlike from texts, there is no ME database publicly available. [4] Hans-Jurgen Winkle et al proposed research on online segmentation and recognition in mathematical expressions. Preprocessing is done by removing slant and pre-recognition is done for separating the symbols “Dot", ”Minus" and “Fraction" from the remaining symbol of the alphabet which have ambiguity, requires contextual knowledge. Within the processing stage. the probabilities calculated in the preceding stages are used for determining the most probable symbol sequence based on the handwritten input. The stroke based features and geometrical features are extracted for recognition. [5] Harold Mouchere et al. report on the third international Competition on Handwritten Mathematical Expression Recognition (CROHME), in which eight teams from academia and industry took part. Training dataset was expanded to over 8000 expressions, and new tools were developed for evaluating performance at the level of strokes as well as expressions and symbols.[6] Hsi-Jim Lee et al present a system to segment and recognize texts and mathematical expressions in a document. 4-Dimensional direction features are extracted from each image block and has been normalized, When the aspect ratio of a symbol is very small (smaller than the threshold TI), the horizontal feature and the two diagonal features are more important than the vertical feature. The system can be divided into six stages: page segmentation and labeling, character segmentation, feature extraction, character recognition, expression formation, and error correction and expression extraction. Similar symbols are grouped together and six groups are formed. Then applying heuristic rules syntax tree is generated, which form an expression. This paper mainly focus on segmentation of expression from documentation. In 2009, Kang Kim et al [8] presents a rule-based approach that utilizes some types of contextual information to improve the accuracy of handwritten mathematical expression(ME) recognition. The base system used for evaluation is a handwritten ME recognizer developed by Rhee et al. The system which is based on a layered structure search reported its recognition accuracy as 87.7% in symbol labeling including segmentation

and structuring, and 38.7% in ME level for KME-I database. Kim uses contextual rules of symbols to improve the accuracy of ME recognition to 77%. Sanjay S. Gharde et al[12] discussed the various steps of recognition process for simple mathematical equations. The feature extraction

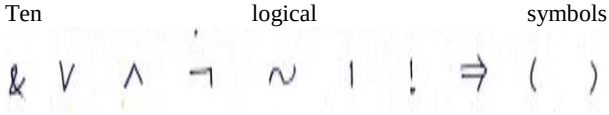
methods used were Zoning, Skeleton based direction, Projection histogram, Profiles: store boundary values from four directions (top, bottom, left and right) of symbols, and Structural features like crossing points, end points and loops also consider of symbols while extract the feature. ANN and SVM classifiers are used for recognition, results in 87.5%, 98.5% recognition respectively. They observed that support vector machine should be used as classifier for improving accuracy. [13] Stephen M. Watt et al presents a recognition system for handwritten mathematical symbols. Elastic matching is a model-based method that involves computation proportional to the set of candidate models. This recognizer can recognize digits, English letters, Greek letters, most of the common mathematical operators and notations with a database of 10,000 mathematical handwriting samples, results in 97% accuracy. [14] In 2004, Utpal Garain et al. aims at automatic understanding of online handwritten mathematical expressions (MEs) written on an electronic tablet. A context free grammar is used to parse the input expression. Contextual information has been used at different levels to increase the system efficiency. [15] In 2006, Xue-Dong Tian et al presents projective features and connected components labeling method to segment the symbols in expressions. Then the peripheral features and directional line element features are extracted from symbols. Finally, a coarse-to-fine classification strategy is employed to recognize symbols with these features.

Table 1: Literature review of Handwritten mathematical expressions

Paper	Online	Offline	Model / Features	Dataset Size	Accuracy Claimed
[2]	√		Multilayer Perceptron Neural network(MLP)	839 symbols: including digits, Roman letters, Greek letters.	87.5%
[4]	√		HMM	153 expressions Each consist of 27 symbols average	72%
[6]		√	Using heuristic rules/ Aspect ratio Vertical, horizontal, diagonal features	127 letters, 36 mathematical operators, 20 numerical Numbers	96%
[8]		√	Contextual rules of symbols(structural analysis)	KME-I	77%
[12]		√	ANN SVM/ Zoning, Skeleton based direction,Projection histogram, boundary and Structural features like crossing points, end points and loops	-	87.5% 98.5%
[13]		√	Elastic matching method Without using any structural and topological features	10000 symbols: including digits, Latin letters, some Greek letters and common mathematical operators	97%
[14]	√		Structural analysis using Context Free Grammar	34636	97.8% (structure Level)
[15]		√	projective features directional line element, and connected components labeling method	100 Chinese mathematical literatures, math handbook and math journals symbols were Used	98.22%

3. DATA COLLECTION

For the proposed work, data is collected from different users by different handwritten style. A standard database does not exist for logical symbol, the database is developed by collecting data from different writers. A4 sheets are used for data collection. Data is collected from twenty writers from different fields, each symbol written by each writer 10 times. Ten logical symbols



are used in the expressions. Database of 2000 symbols have been collected, The datasheets were scanned by scanner and individual symbol images are cropped from this scanned image manually, which results in gray scale image of symbol. The data of alphabets [a-z and A-Z] are collected from different users. A4 sheets are used to collect the data of alphabets. Database of 1560 letters have been collected from different writers . For 26 lowercase letters , 50 images of each lowercase letter are collected and 10 images for each uppercase letters are collected. The numeral 1 is frequently used as a subscript in the math expression, hence collected images of numeral 1.

4. RECOGNITION SCHEME AND ME AMBIGUITIES

This paper introduce the problem of mathematical logical expression recognition. This work also deals with two dimensional structure of expression having subscripts. Three sub problems are used to resolve ME recognition-segmentation, symbol recognition, expression interpretation. This paper emphasis on first two sub problems. There are many sources of ambiguity in ME recognition. The ME itself can be interpreted in different way. Some ambiguity in the expression is shown in the figure 1. The ambiguities in the character V and symbol V, symbol (and C are present in the expressions.

Figure 1 : a. Ambiguous expressions

Figure 1:b. Unambiguous expression

Mathematical expression can be recognized by using following steps.

Figure 2: Phases in logical ME recognition

4.1 Algorithm for Preprocessing of Expression:

1. Handwritten images of expressions written on A4 sheet are scanned and store in .jpeg format . The image is RGB image.
2. RGB image is converted into grayscale image.
3. Grayscale image is converted into binary image, using a suitable threshold value by Otsu's method.
4. The noise is removed from the image using median filter.
5. The morphological operation image opening is used to separate touching symbols, dilation is used to join unconnected pixels..
6. Image thinning is used to reduces space while storing topological information of an image.

5. SEGMENTATION

In the segmentation stage, The image of a sequence of characters is decomposed into sub-images of individual characters. In the proposed work, the pre-processed input image is segmented into isolated characters and symbols by assigning a number to each character using a labeling process. Each individual character is uniformly resized into 64x64 pixels for extracting its features.

5.1 Algorithm for segmentation

1. Bounding box is used to segment each character from the expression.
2. Once the characters are separated, the bounding box area and centroid features are used to find subscripts of the image. The y-coordinate of the centroid and bounding area is used for separating subscripts by using threshold value.
3. The segmentation separates the expression into two parts: subscripts and other characters.

The result after the segmentation of the ME is shown in the figure 3 and the table 2 shows the parameters of the bonding box used to separate the subscripts.

Original Image **Segmented Image** **Centroid Plot**

Figure 3. Result of segmented image

Table 2: Features detected using regionprops

Centroids (x,y) of image		**Boundingbox values of above expression**			
28.0832	**41.2020**	**8.5000**	**17.5000**	**43.0000**	**47.0000**
62.3832	**65.2523**	**59.5000**	**51.5000**	**6.0000**	**28.0000**
106.9294	**44.5853**	**88.5000**	**19.5000**	**34.0000**	**48.0000**
144.7848	**46.0583**	**135.5000**	**11.5000**	**25.0000**	**58.0000**
170.2336	**63.6642**	**167.5000**	**49.5000**	**6.0000**	**30.0000**

6. FEATURE EXTRACTION TECHNIQUES

Once the characters are isolated, features are extracted from the isolated image. This section introduce the techniques used for feature extraction.

6.1. Normalized Chain Code

Chain code representation gives the code of the boundary of character image. Using freeman chain code method, extract chain code of each image. Length of chain code varies from image to image. So the chain code is normalized by the following method[22]:

Step1: Find chain code of image using Freeman method.

Step 2: Find the frequency of each digit to get vector A1 of size 8.

Step 3: Take sum of all A1 elements (A2=ΣA1)

Step 4: Calculate probability of each digit A3=A1 / |A2|

After concatenation of the two vectors A1 and A3 , we get feature vector of size 16.

6.2 Moment Invariant Features

A set of seven 2-D moment invariant features are insensitive to translation, scale change, mirroring, and rotation. These can be derived from seven equations. It computes the moment invariants of the image and obtained seven-element row vector. Using these features we get a high degree of invariance. Hu's Seven Moment Invariants are invariant under translation, changes in scale, and also rotation. So it describes the image despite of its location, size, and rotation. The seven features of moment invariant are extracted from the image.

6.3. Density Feature: UDRL density (Up-Down,Right-Left)

Step 1: Image of size (32*32) is divided into zones. Each zone of size 8*8 gives 16 zones.

Step 2: Sum foreground pixels in each zone, get the 16 array (density of each zone).

Step 3: Extract height and width, difference between left and right zone density, and up and down density.

Up=sum of Density 1 to 8 zones, and Down =sum of density of 9 to 16 zones Left= sum of density of 1, 2,5,6,9,10,13,14 zones, and Right= total density-left

Step 4: Calculate difference, diff1=up-down and diff2=leftright

Step 5: If the diff1 >2 then d1=1, if diff1 <-2 then d1=2 else d1=0,

If diff2>2 then d2=1 , if diff2<-2 then d2=2 else d2=0**(Here** - 2 to 2 is error rate)

Step 6: To get average density, Combine two consecutive zones, and find sum of density, then sum is divide by number of pixels in that zone. Finally, we obtained 8 features of average density of 8 zones.

Step 7: feature vector of size 28 is consists of density of each zone(16 features), height, width, d1,d2 and 8 features of zoning.

6.4. Projection Histogram:

Projection histogram calculates foreground pixels in different direction. This work uses three type of projection histogram; Vertical, horizontal, and left diagonal.

To find image histogram , the image is resize to 32×32 . Feature vector of size 127 is consists of 32 vertical, 32 horizontal and 63 diagonal histogram.

7. CLASSIFICATION USING SVM

Support vector machines(SVM) is one of the supervised learning method. Support Vector Machines are based on the concept of decision planes that define decision boundaries. A decision plane is used to separate a set of objects having different class memberships. Support Vector Machine (SVM) is a classifier which construct hyper planes in a multidimensional space that separates cases of different class labels and perform classification. The linear kernel bsvm2 is a multiclass bias support vector machine that is used for training and built a model that assign a class label for each category.

8. EXPERIMENTAL RESULTS

The recognition system has been implemented using Matlab 7.0. Table 3 shows the result of recognition of isolated characters (symbol+ letters+numerals) using SVM classifier. These characters are used as a train data. The dataset consists of 2000 logical symbols and 1560 alphabets. A set of 50 handwritten logical Expressions consisting 273 characters have been tested with this train data.

Table 3: Recognition of isolated characters

Features	Feature vector size	Recognition rate(%)	
		Letters	symbols
Zoning density + average + height+ width (16+8+2+2)	28	96	82
Normalized chain code(NCC)(16)	16	93	91
moment invariant+NCC(7+16)	23	95.5	98.2
Projection histogram	127	95	92
NCC+zoning + (d1, d2) +projection histogram(16+8+2+127)	153	99.9	95.6

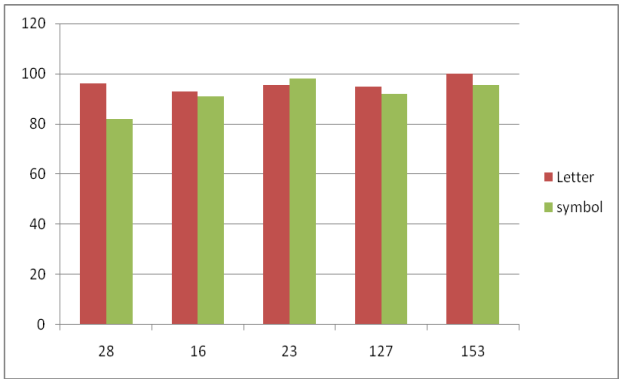


Figure 4. Feature vector Vs Recognition Rate

9. CONCLUSION

Recognition of 2-D mathematical expression is becomes more challenging. To improve the rate of recognition, this work evaluated different techniques. The recognition rate (R) is calculated as follow.

$$R = \frac{\text{No. of characters correctly recognized}}{\text{Total No. of characters}} * 100$$

It is observe that normalized chain code feature gives good recognition rate for symbol and letter when combine with other features. For the feature vector of size 23 includes normalized chain code and moment invariant features, the recognition rate of symbols and letters is high compare to other feature vector. A set of 50 handwritten logical expressions have been tested. The overall average recognition rate of the ME recognition is 93.8%. In this research expressions with superscripts have not been consider. The future scope is to recognize superscripts.

10. ACKNOWLEDGMENT

The author is grateful to Dr. M. S. Prasad for his contributive support and encouragement during this work

11. REFERENCES

[1] Ahmad Moritaser Awal, Harold Mouchere Christian Viard Gaudin. Towards Handwritten Mathematical Expression recognition. IEEE 978- 07095, 2009.

[2] Ahmad Montaser Awal, Harold Mouchere, Christian Viard Gaudin. The problem of handwritten mathematical expression recognition. ISBN, 978-0- 7695-4221-8,2010.

[3] Christopher Malon, Seiichi Uchid, Masakazu Suzuki, Mathematical symbol recognition with support vector machines, Pattern Recognition Letters 29 (2008) 1326–1332, Elsevier.

[4] Hans Jurgen Winkler and Manfred Lang. On-Line Symbol segmentation and recognition in handwritten mathematical expressions. 0-8186- 7919-0/97,IEEE

[5] Harold Mouch`ere_, hristian Viard-Gaudin_Richard Zanibbiy, Utpal Garainz, Dae Hwan Kimx and Jin Hyung Kimx, ICDAR 2013 CROHME:Third International Competition on Recognition of Online Handwritten Mathematical Expressions,

[6] His-Jian Lee And J. Wang. Design of a mathematical expression recognition system, 0- 8186-7128-9/95,IEEE

[7] Kang kim, Taik Rhee, Jae LEE. Utilizing consistency context for handwritten mathematical expression recognition. 978-0-7695-3725-2/2009 IEEE.

[8] Kazuki Ashida, Masayuki Okamoto, Hiroki Imai, Performance Evaluation of a Mathematical Formula Recognition System with a large scale of printed formula images, Proceedings of the Second International Conference on Document Image, Analysis for Libraries (DIAL'06),0-7695-2531-8/06, 2006 IEEE

[9] Lei Gao, Shulin Pan, Shen Jiao, An Analytic Hierarchy Process Based Method To Process Mathematical Expressions, Journal Of Theoretical And Applied Information Technology 31st January 2013. Vol. 47 No.3, Issn: 1992-8645

[10] M. Padmanaban, E. A. Yfantis. Handwritten character recognition using conditional probabilities.

[11] Qi Xiangwei Pan Weimin Yusup Wang Yang, The study of structure analysis strategy in handwritten recognition of general mathematical expression, International Forum on Information Technology and Applications, 978-0-7695-3600-2/09, 2009 IEEE

[12] Sanjay S. Gharde, Baviskar Pallavi, V K. P. Adhiya, Evaluation of Classification and Feature Extraction Techniques for Simple Mathematical Equations, International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868 Foundation of Computer Science FCS, New York, USA Volume 1– No.5, February 2012.

[13] Stephen M. Watt Xiaofang Xie, Prototype Pruning by Feature Extraction for Handwritten Mathematical Symbol Recognition,Department of Computer Science, University of Western Ontario,Canada

[14] Utpal Garain, B. B. Chaudhuri, R. P. Ghosh, A Multiple-Classifier System for Recognition of Printed Mathematical Symbols, Proceedings of the 17th International Conference on Patter Recognition (ICPR'04), 1051-4651/04 , IEEE.

[15] Xue-Dong Tian, Hai-Yan Li, Xin-Fu Li. Research on symbol recognition for mathematical expressions, 0-7695- 2616-0/2006 ,IEEE.

[16] Xie,Xiaofang. On the recognition of handwritten mathematical symbols. Proquest NR39341,2008

[17] Taik HeonRhee, JinHyungKim , Efficient search strategy in structural analysis for handwritten mathematical expression recognition, pattern recognition (ScienceDirect)0031-32,2009 Elsevier

[18] Francisco Álvaro, Richard Zanibbi, A Shape-Based Layout Descriptor for Classifying Spatial Relationships in Handwritten Math, 2013 ACM 978-1-4503-1789/4/13/09

[19] Sanjay S. Garde, Pallavi V. Baviskar, K. P. Adhiya, Identification of Handwritten Simple Mathematical Equation Based on SVM and Projection Histogram, International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231- 2307, Volume-3, Issue-2, May 2013

[20] Anita Jindal,Renu Dhir,Rajneesh Rani, Diagonal Features and SVM classifier for Handwritten Gurumukhi Character recognition, International Journal of Advance Reasearch in Computer science and software engineering, Vol 2, Issue 5, May 2012.

[21] G.G. Rajput, S. M. Mali, Marathi Handwritten Numeral Recognition using Fourier Discropters and Normalized Chain code, IJCA, Special issue ITRPPR, 2010