ICFHR 2012 – Competition on Recognition of On-line Mathematical Expressions (CROHME 2012)

H. Mouchère and C. Viard-Gaudin
IRCCyN/IVC – UMR CNRS 6597
Ecole Polytechnique de l'Université de Nantes
harold.mouchere@univ-nantes.fr
christian.viard-gaudin@univ-nantes.fr

D. H. Kim and J. H. Kim
Division of Computer Science
Korea Advanced Institute of Science and Technology
ofreunde.edward@gmail.com
jkim@kaist.edu

U. Garain
Computer Vision and Pattern Recognition (CVPR) Unit
Indian Statistical Institute,
utpal@isical.ac.in

Abstract

This paper presents an overview of the second Competition on Recognition of Online Handwritten Mathematical Expressions, CROHME 2012. The objective of the contest is to identify current advances in mathematical expression recognition using common evaluation performance measures and datasets. This paper describes the contest details including the evaluation measures used as well as the performance of the 7 submitted systems along with a short description of each system. Progress as compared to the 1st version of CROHME is also documented.

1. Introduction

As a research problem, automatic recognition of mathematical expressions (ME) exhibits several fascinating challenges [1]. This recognition problem is different from the traditional OCR problem. Presence of two-dimensional structures, enormous uncertainties and ambiguities in handwritten expressions makes the automatic understanding problem difficult and at the same time enticing for the researchers. Achieving success in this domain would in turn progress the state of the art in understanding of visual languages.

The birth of CROHME (Competition on Recognition of Online Handwritten Mathematical Expressions) aimed at bringing the researchers under a common platform so that they share the same dataset for their respective research and report performance of their systems on a common test data. The first version of CROHME was organized with ICDAR 2011 [2]. The evaluation results nicely addressed the achievement of the systems and pointed out the issues for future research.

The second version of CROHME, organized with ICFHR 2012, shares the same spirit. Number of participating groups is increased as it was expected. Interestingly, industry research labs also participated along with University labs. The training dataset is augmented by adding more samples to represent more varieties and real life difficulties in the dataset. A new test set is developed and several new analyses are done as part of the evaluation.

The rest of the paper is organized as follows. Section 2 provides an overview of the format of the competition, its organizers, the participants, data set, evaluation strategies, etc. Section-3 gives elaborate information on data format and organization of the data set and its content and coverage. Section 4 briefly describes the working principles of the participating systems. Section 5 presents the evaluation results, analysis of the results and announces the winner of this competition. The following section, i.e., section-6 concludes the paper.

978-0-7695-4774-9/12 $26.00 © 2012 IEEE
DOI 10.1109/ICFHR.2012.215

2. Overview of the competition

The same three research labs who organized CROHME 2011 conducted this competition. Initially, ten research groups registered themselves for participating in this event. Eight research groups participated in the development process, i.e., received the training dataset and attempted to develop their systems. Finally, six teams submitted their systems. However, an additional seventh system was developed by one of the organizing groups and hence, it was not considered as a competing system.

Table 1. CROHME 2011 vs. CROHME 2012

Features	CROHME 2011	CROHME 2012
# Registrations	6	10
# Training samples	921 (divided into two parts)	1336 (divided into three parts)
# Grammars	2	3
# Test samples	348	488
# Systems	5	7

The dataset has three parts. Part-I and Part-II were present in CROHME 2011. A new part, i.e. PART-III is added in this competition. Each part is characterized by its respective grammar. The details about the grammars are given in the next section.

A new test dataset, different from CROHME 2011, is formed for evaluating CROHME 2012 systems. Like training samples, test expressions are also divided into three parts conforming to the grammars defined for each one. The training data was distributed two and half months before the evaluation of the systems, while testing was done at the organizers' end. Five parameters as explained in section-5 are measured for evaluating the recognizers. However, the final rating is based on the expression recognition accuracy for Part-III test samples. The details of evaluation are reported in section-5. Table 1 summarizes the significant differences between CROHME 2011 and 2012. Table 2 shows the number of samples in CROHME 2012 training and test datasets.

3. Data format, training and testing sets

The ink corresponding to each expression is stored in an InkML file. An InkML file mainly contains three kinds of information: (i) the ink: a set of traces made of points; (ii) the symbol level ground truth: the segmentation and label information of each symbol of the expression; and (iii) the expression level ground truth: the MathML structure of the expression.

The two levels of ground truth information (at the symbol as well as at the expression level) are entered manually. Furthermore, some general information is added in the file: (i) the channels (here, X and Y); (ii) the writer information (identification, handedness (left/right), age, gender, etc.), if available; (iii) the LaTeX ground truth (without any reference to the ink and hence, easy to render); (iv) the unique identification code of the ink (UI), etc.

Table 2. Data in CROHME 2012

Dataset	Number of expression samples in		
	Part-I	Part-II	Part-III
Training	296	921	1336
Test	108	301	488

The InkML format makes references between the digital ink of the expression, its segmentation into symbols and its MathML representation. Thus, the stroke segmentation of a symbol can be linked to its MathML representation.

The type of expressions that are allowed to appear in a part is dictated by the corresponding grammar, which is checked on the LaTex string. The grammar for Part-I accepts only 41 terminals and imposes limitations among logical relationships: (i) only one symbol in subscript or superscript is allowed, (ii) no recursive fraction is there, however, a sum of fraction or fraction of sum may appear but no fraction of fractions can be allowed, (iii) no product of fraction can appear in the expressions. Two permissible recursive expressions are: (a) repeated sum, i.e. sum of sums is permitted and (b) nested root, i.e. a square root can be found in other square root.

The grammar for Part-II expression is less restricted than that of Part-I. The number of terminal symbols is increased to 60. There are no limits on recursions of operations like sum, product, function call, fraction, root, sub/superscript on symbols, etc. Grammar-III further relaxes restrictions. Though a human readable version of Grammar-III was not provided, a parsing script is made available to the participants. The number of terminal symbols is increased to 75 (mainly adding brackets, some Latin, Greek letters and set operators). For dataset request and explicit grammar rules one may look at the competition site, i.e. http://www.isical.ac.in/~crohme. Validators are available to check whether a given expression conforms to a particular grammar. The validator extracts the LaTeX string of the expression and parses it to validate whether it is accepted by a grammar (i.e. Grammar I, II, or III).

MathML normalization: As the evaluation attempts to do an exact matching of MathML structures, the MathML output produced by the recognition systems should use the same structure as the ground-truth. The problem is that several MathML valid structures can

represent the same expression. Hence, normalized descriptions with predefined rules are checked and processed during evaluation.

4. Overview of the participating systems

Ten participants originally registered for participating in the competition. Two teams remained silent after receiving training data and the other eight teams continued to work on developing/improving their systems for the competition. Finally, two teams decided not to participate in the blind evaluation process and six systems were submitted for evaluation. A system developed by one of the organizing labs also participated in the evaluation process. Hence, altogether seven systems were evaluated. Out of these seven systems, five systems participated in CROHME 2011. So CROHME 2012 introduces two new systems one of them is from an industry. Table-3 provides the affiliations of the participating teams.

System-I: This recognition system is based on stochastic context-free parsing of two-dimensional (2D) grammars [3]. The recognition process is a CYK-based parsing method such that the parsing table is initialized with several segmentation and symbol recognition hypotheses. Then the parsing algorithm obtains the most probable hypothesis according to the given grammar. Thus, the system solves jointly the symbol recognition and structural analysis of the handwritten mathematical expression by using both online and offline information.

System-II: This system is developed at the Institute for Language and Speech Processing, Athena and named as Math-ILSP system. The system incorporates the following two major modules: (i) symbol recognition based on a template elastic matching distance between pen direction features [4] and (ii) structural analysis of the ME based on extracting the baseline of the ME and then classifying symbols into levels above and below the baseline. The symbols are then sequentially analyzed using six spatial relations and the respective 2D structure is interpreted to give the resulting MathML representation of the input expression.

System-III: This system [5] is based on the PhD work of A. M. Awal adapted to this new competition by S. Medjkoune. It aims at handling mathematical expression recognition as a simultaneous optimization of expression segmentation, symbol recognition, and 2D structure recognition under the restriction of a mathematical expression grammar. The approach transforms the recognition problem into a research of the best possible interpretation of a sequence of input strokes. The symbol classifier is a classical neural network, a multilayer perceptron, which has the capability to reject the invalid segmentation hypotheses, unlike most existing works. The originality of the system stems from the global learning schema. This learning allows training the symbol classifier directly from mathematical expressions. The advantage of this global learning is to consider the junk examples and include them in the symbol classifier knowledge. Furthermore, we have proposed a contextual modeling based on structural analysis of the expression. This analysis is based on models learnt directly from the expressions using the global learning scheme, most of the expressions used from training come from the HAMEX dataset [6].

Table 3. CROHME 2012 participating groups

System	Group	Country
I	University of Valencia	Spain
II	Athena Research Center	Greece
III	University of Nantes*	France
IV	Rochester Institute of Technology	USA
V	Sabanci University	Turkey
VI	University of Waterloo (new comer)	Canada
VII	Vision Objects (new comer)	France

** This is the system from one of the organizer.*

System-IV: Rochester Institute of Technology (NY, USA) submitted this three-stage system comprised of a fuzzy segmenter, a Hidden Markov Model classifier, and a DRACULAE parser [7].

Strokes are pre-processed to detect some specific conditions to guide the subsequent merging. The *merge* membership values of the strokes are defined by comparing the stroke distances against predefined threshold values. All possible sequences of merge decisions are considered for sequences of strokes/segments where segmentation is not determined precisely. These (local) segmentation alternatives are scored by the product of *merge* and *split* (set to '1') membership. An upper bound of 10 adjacent strokes in one of these *fuzzy* regions is set to reduce combinations.

The HMM used for classification is similar to that in [8], trained on the Part 3 data, but with an additional angular feature. The final segmentation obtained by greedy selection of the highest probability for each local *fuzzy* segmentation. A second segmentation index is obtained using the sum of the top-1 HMM classification probability, and its division by the average probability produced for symbols of the correct class after training. This sum is associated with each stroke belonging to a segment/symbol, and then added across strokes in a *fuzzy* segmentation. The

final segmentation probability is defined using a histogram over the fuzzy and HMM-based scores, to estimate the highest probable valid segmentation.

Finally, a DRACULAE parser is used to produce the final parse result from symbols and their bounding boxes [7]. Additional tree rewriting rules are added to correct common classification errors (e.g. recognizing 'log' as '10g').

System-V: The system from Sabanci University, Turkey, uses a 2D-stochastic grammar to parse the handwritten ME. During the parsing of the input, grammar rules are applied iteratively until no more rules can be applied. Each rule generates a token and the system aims to build one token representing the whole expression at the end of the recognition process.

A grammar rule is applicable for a given set of tokens (initially recognized symbols) if the applicability predicate of the rule decides that the required relationship (up, down, inside etc.) roughly exists between the given tokens. Each rule generates a new token from the neighboring tokens generated in the previous stages. For instance, the subscript rule expects the subscript token to be roughly below and to the right of the main token. A likelihood score is assigned to the generated token based on the likelihood of the component symbols and the likelihood of the 2D relationship between the component symbols. In case of a subscript, the likelihood value depends on the relative position and size of the two tokens.

In this system, the grammar rules are applied without any particular order, because the system generates all likely interpretations of a given input string, along with their likelihoods. The system can be for longer expressions, as it keeps track of all possible likely interpretations of neighboring tokens. In future work, we will look at speeding up the system by expanding the most likely tokens first. Details of the system can be found in [9]. It is to be noted that this system does not incorporate any special measure to handle Part-III expressions. Results reported for this system on Part-III expressions are basically achieved by the system tuned for Part-II dataset.

System-VI: The Waterloo recognizer [10] was developed for the MathBrush pen-math system [11], and is based on relational grammars and fuzzy sets. It works in three stages: symbol recognition, parsing, and tree extraction. In the symbol recognition step, candidate stroke groups are identified in the input by measuring the distance between nearby strokes and the degree to which stroke bounding boxes overlap. Input features such as dots and stacked structures (as in '≤ and '≡') are also identified and grouped. At this point, the input is typically oversegmented. Using a symbol recognizer [10], each potential group is assigned a list of candidate symbols with confidence scores.

The parsing step uses a fuzzy relational grammar and a variant of Unger's top-down parsing method to generate a shared parse forest representing all recognizable parses of the input. The grammar includes five relations: horizontal and vertical adjacency, subscript, superscript, and containment. Each production may use one relation, with each adjacent pair of RHS entries being required to satisfy the relation. To reduce the algorithm's complexity, only rectangular partitions of the input are parsed. The symbol recognition candidates and grammar relations further prune the search space. In this step, relation membership is based on the angle between bounding box centroids and bounding box overlap.

Finally, in the tree extraction step, individual parse trees are extracted in ranked order from the parse forest. The score of a tree is given by the geometric average of all the relevant symbol recognition scores and relation membership grades. In this step, more specific relation membership functions are used, which take into account the specific content of a parse tree by means of "relational classes". These classes include "baseline symbol", "ascender", "descender"', etc. The relation functions compare also relative heights, baseline positions, and the distance between symbols. For CROHME, only the first tree, representing the top-ranked parse, is used and converted to MathML.

System-VII: The MyScript Equation recognizer from Vision Objects (http://www.visionobjects.com), France is an on-line recognizer that processes digital ink. The overall recognition system is built on the principle that segmentation, recognition and interpretation have to be handled concurrently and at the same level in order to result in the best candidate. The Equation recognition engine analyzes the spatial relationships between all the parts of the equation, in conformity with the rules laid down in its grammar, to determine the segmentation of all its parts. The grammar is defined by a set of rules describing how to parse an equation, each rule being associated with a specific spatial relationship. For instance, a fraction rule defines a vertical relationship between a numerator, a fraction bar and a denominator. The Equation recognizer has also a symbol expert that estimates the probabilities for all the parts in the suggested segmentation. This expert is based on feature extraction stages, where different sets of features are computed. These feature sets use a combination of on-line and off-line information. The feature sets are processed by a set of character classifiers, which use Neural Networks and other pattern recognition paradigms. The Equation

recognition engine includes a statistical language model that uses context information between the different symbols depending on their spatial relationships in the equation. Statistics have been estimated on hundreds of thousands of equations. A global discriminant training scheme on the equation level with automatic learning of all classifier parameters and meta-parameters of the recognizer is employed for the overall training of the recognizer.

5. Evaluation

For each system, five aspects are measured. They are (i) *ST_Rec*: the stroke classification rate, representing the percentage of strokes with the correct symbol, (ii) *SYM_Seg*: the symbol segmentation rate, defining the percentage of symbols correctly segmented, (iii) *SYM_Rec*; the symbol recognition rate, computing the performance of the symbol classifier when considering only the correct segmented symbols, (iv) *STRUCT*: the MathML structure recognition rate, computing the percentage of expressions (MEs) having the correct MathML tree as output irrespective of the symbols attached to its leaves, e.g., these two expressions share the same MathML structure: "$x^2 - 1$" and "$2^a + b$". The last measurement is (v) *EXP_Rec*: the expression recognition rate, which informs the percentage of MEs totally correctly recognized. This is a very challenging indicator since the slightest error anywhere in the ME prevents to count it. However, in order to have a better insight of the capacity of the respective systems, we also extend this indicator with (vi) *EXP-Rec_1, _2, _3*, giving the percentage of MEs recognized with at most 1 error, 2 errors and 3 errors (in terminal symbols or in MathML node tags) given that the tree structure is correct.

In order to measure the progress of the systems, first, we benchmark them on the test dataset used for CROHME 2011, which was composed of two parts. This dataset was publicly available to the participants. The results are given in Table 4. It can be observed that a very significant increase in expression level recognition has been achieved in 2012 with respect to CROHME 2011. Four of these systems (I, III, VI and VII) are better than the winner (I) of the first CROHME contest [2] which achieved an expression level recognition accuracy of 19.8% on Part II test set.

As explained in Sections II and III, we have defined three levels in the competition this year compared to the two levels defined during CROHME 2011. Only the three training datasets were available for the participants. The results for CROHME 2012 are displayed in Table 5.

Table 4. Expression level recognition rates on the test dataset of CROHME 2011

System	Part I		Part II	
	System 2011	System 2012	System 2011	System 2012
I	29.28	33.15	**19.83**	34.48
II	0.0	17.13	0.0	7.76
III	40.88	63.54	22.41	47.41
IV	4.42	27.07	2.59	16.38
V	0.55	30.94	0.29	18.68
VI	N/A.	57.46	N/A	54.70
VII	N/A	91.16	N/A	85.34

Table 5. Main results on the test dataset of CROHME 2012

	System	*ST_Rec*	*SYM_Seg*	*SYM_Rec*	*STRUCT*	*EXP_Rec*
Part I	I	80.74	90.74	89.20	62.04	35.19
	II	59.14	73.31	79.79	21.30	8.33
	III	90.05	94.44	95.96	70.37	57.41
	IV	78.24	92.81	86.62	50.93	28.70
	V	61.33	72.11	87.76	37.04	22.22
	VI	89.00	97.39	91.72	78.70	51.85
	VII	97.01	99.24	97.80	91.67	**81.48**
Part II	I	85.05	90.66	91.75	50.17	33.89
	II	58.53	72.19	86.95	12.29	6.64
	III	82.28	88.51	94.43	49.83	38.87
	IV	76.07	89.29	91.21	27.57	14.29
	V	49.06	61.09	88.36	17.61	7.97
	VI	90.71	96.67	94.57	69.44	49.17
	VII	96.85	98.71	98.06	88.37	**75.08**
Part III	I	79.85	91.95	86.25	42.21	22.75
	II	55.75	71.21	84.97	9.84	3.69
	III	78.94	87.75	91.38	36.89	25.61
	IV	72.12	87.51	87.62	23.77	9.43
	V	45.42	59.20	84.27	14.75	4.92
	VI	86.41	95.56	91.17	61.27	40.16
	VII	95.75	98.84	96.85	80.33	**62.50**

When comparing Parts I and II from Tables 4 and 5, we notice that the new test set for Part I is somehow easier than its last year's counterpart, but conversely Part II test set is slightly more difficult than it was in 2011. As mentioned in the call for competition, we have ranked the system according to the *EXP_Rec* rate obtained on Part III test dataset. **System VII (Vision Objects) is clearly the winner of this second edition of CROHME**. With this system, 62.5% of Part III expressions are fully recognized, and 80.3% have a correct layout structure. To have a better idea of the capabilities of the systems we have computed Table 6 that shows the expression recognition accuracies with one, two or three errors per expression.

Table 6. Expression recognition rates with 1, 2 or 3 errors on CROHME 2012 Part III

System	*EXP_Rec*	*EXP Rec_1*	*EXP Rec_2*	*EXP Rec_3*
I	22.75	34.63	40.98	42.21
II	3.69	6.35	9.02	10.45
III	25.61	36.07	39.14	39.96
IV	9.43	18.44	23.16	24.80
V	4.92	10.66	14.14	14.96
VI	40.16	54.10	59.02	61.89
VII	**62.50**	**78.89**	**81.76**	**81.97**

A gap exists between *EXP_Rec* and *EXP_REC_1*, showing that the corresponding systems have enough room for further improvements. Conversely, the narrow differences between *EXP_REC_2* and *EXP_REC_*3 show that when more errors go wrong, it is difficult to improve the accuracy by incorporating a single correction.

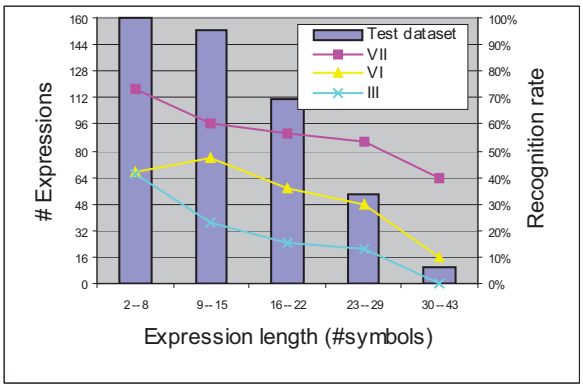

FIGURE 1. Recognition rates (*EXP_Rec*) with respect to the expression length on Part III

Another interesting analysis concerns the distribution of errors with respect to the size of the expressions. Of course, the longer the expressions, the harder it is to recognize them. Figure 1 illustrates this behavior, and displays analysis results for the three leading systems (III, VI, and VII). The best system succeeds to recognize 73% of the short expressions and 40% of the longest expressions.

6. Conclusion

The second edition of CROHME has confirmed the interest of the international community for this event. The number of participants has increased, from five for CROHME 2011 to seven in 2012, with the presence of six academic teams and one company. Five groups are coming from Europe and two from North America. We would like to encourage new comers, specifically from Asia, to join the subsequent CROHME session.

The evaluation results show the considerable improvements over CROHME 2011. The expression level accuracy has achieved an impressive rate, i.e. 62.5% by the winning system, while the other systems have also made significant progress. However, there is still room for more improvements, and we plan to pursue this initiative in near future. We would like to extend the contest with an additional Part IV which would include more symbols, specifically all related to set theory, and to adapt the grammar to support the Boolean notation with an over-bar to denote a negation, as in $\overline{a \oplus \bar{b}} = \bar{a}.b + a.\bar{b}$.

References

[1] R.H. Anderson, "Syntax-directed Recognition of Hand-printed Two-dimensional Mathematics," Doctoral Dissertation, Dept. of Engineering and Applied Physics, Harvard University, 1968.

[2] H. Mouchère, C. Viard-Gaudin., D. H. Kim, J. H. Kim and U. Garain, "CROHME2011: Competition on Recognition of Online Handwritten Mathematical Expressions," in Proc. ICDAR, pp. 1497-1500, 2011,

[3] F. Alvaro, J. Sanchez and J. Benedi, "Recognition of Printed Mathematical Expressions Using Two-Dimensional Stochastic Context-Free Grammars", in Proc. ICDAR, pp. 1225-1229, 2011.

[4] F. Simistira, V. Katsouros, and G. Carayannis, "A Template Matching Distance for Recognition of On-Line Mathematical Symbols," in Proc. ICFHR, pp. 415-420, 2008.

[5] A.-M. Awal, H. Mouchère, and C. Viard-Gaudin, "Towards handwritten mathematical expression recognition," in Proc. ICDAR, pp. 1046-1050, 2009.

[6] S. Quiniou, H. Mouchère, S. Peña Saldarriaga, C. Viard-Gaudin, E. Morin, S. Petitrenaud and S. Medjkoune, "HAMEX – a Handwritten and Audio Dataset of Mathematical Expressions," in Proc. ICDAR, pp.452-456, 2011.

[7] R. Zanibbi, D. Blostein, and J.R. Cordy, "Recognizing mathematical expressions using tree transformation," IEEE Trans. Pattern Analysis and Machine Intelligence, 24 (11), 2002, 1455-1467.

[8] L. Hu and R. Zanibbi, "HMM-Based Recognition of Online Handwritten Mathematical Symbols Using Segmental K-means Initialization and a Modified Pen-up/down Feature," in Proc. ICDAR, pp. 457-462, 2011.

[9] M. Celik and B. Yanikoglu, "Probabilistic Mathematical Formula Recognition Using a 2D Context-Free Graph Grammar," in Proc. ICDAR 2011, pp. 161-166, 2011.

[10] S. MacLean and G. Labahn, "A new approach for recognizing handwritten mathematics using relational grammars and fuzzy sets," in Int. J. on Document Analysis and Recognition (IJDAR), to appear.

[11] G. Labahn, E. Lank, S. MacLean, M. Marzouk, and D. Tausky, "MathBrush: A System for Doing Math on Pen-Based Devices," in Proc. IAPR Workshop DAS, pp. 599-606, 2008