ICDAR 2013 CROHME:
Third International Competition on Recognition of Online Handwritten Mathematical Expressions

Harold Mouchère*, Christian Viard-Gaudin*, Richard Zanibbi†, Utpal Garain‡, Dae Hwan Kim§ and Jin Hyung Kim§,
Email: {harold.mouchere,christian.viard-gaudin}@univ-nantes.fr, rlaz@cs.rit.edu, utpal@isical.ac.in
freunde.edward@gmail.com, jkim@kaist.edu

*LUNAM/IRCCyN/IVC - UMR CNRS 6597, Ecole Polytechnique de l'Université de Nantes; 44306 Nantes Cedex 03, France
†Department of Computer Science, Rochester Institute of Technology; Rochester, NY, USA 14623-5608
‡Computer Vision & Pattern Recognition Unit, Indian Statistical Institute; 203, B. T. Road, Kolkata, India
§Korea Advanced Institute of Science and Technology, 291 Daehak-ro, Yuseong-gu, Daejeon 305-701; Republic of Korea

Abstract—We report on the third international Competition on Handwritten Mathematical Expression Recognition (CROHME), in which eight teams from academia and industry took part. For the third CROHME, the 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. As an informal measure of progress, the performance of the participating systems on the CROHME 2012 data set is also reported. Data and tools used for the competition will be made publicly available.

I. Introduction

Currently, string languages (e.g. LATEX) and template editors are commonly used for entering math in technical documents, search queries, and in other applications. These input methods are unfamiliar and/or cumbersome for many users. Robust handwritten math recognition systems would allow for more natural math input interfaces; however, automatic recognition of handwritten math is a challenging two-dimensional parsing problem. A large variety of symbols are used in math expressions, and frequently there are ambiguities in symbol location, identity and layout in handwritten expressions. Even recognizing isolated expressions written using a tablet is difficult, such as those used in the Competition on Recognition of Online Handwritten Math Expressions (CROHME[1]). In this paper we report on the systems and results for the third CROHME competition. Descriptions of the first two CROHME competitions are available [1], [2].

Research into recognizing handwritten math began in the late 1960's, and since that time a number of research labs around the world have worked on the problem [3]. The relatively small community of math recognition researchers along with the absence of benchmark data sets, standard encodings, and evaluation tools limited the rate of progress on this problem. In particular, researchers have often used their own (frequently, private) data sets for evaluation. The CROHME competition was first held as a part of ICDAR 2011, with the intent to facilitate meaningful comparison of systems, and make it easier for newcomers to work on handwritten math recognition.

CROHME 2013 was organized by one European (IRC-CyN/IVC), two Asian (ISI/CVPR, KAIST) and one North American (DPRL/RIT) research lab. As seen in Table I, participation in CROHME has reached eight teams this year. With help from the community, the training set has been expanded more than six times to over 8800 expressions, and a new *writer-independent* test set of 671 expressions was collected by the IVC and DPRL labs: all writers in the test set are not included in the training set. As illustrated in Table I, over the years the difficulty of the CROHME recognition task and the size of both the training and testing data sets have increased.

New tools[2] were developed to collect symbol and stroke-level metrics from label graphs [4] (LgEval), and for converting between CROHME InkML and label graph files (CROHMELib). CROHME InkML files are in an XML format, using Presentation MathML for expression structure. The CROHME InkML viewer[3] visualizes stroke data along with stroke segmentation, symbol classification, and symbol layout.

TABLE I
CROHME COMPETITION METRICS

	2011	2012	2013
Expression Grammars	2	3	2
Max. # Symbol Classes	60	75	101
Training expressions	921	1341	8836
Test expressions	348	486	671
Registrations	6	10	9
Participating systems	4	6	8

[1] www.isical.ac.in/~crohme/

[2] www.cs.rit.edu/~dprl/CROHMELib_LgEval_Doc.html
[3] saskatoon.cs.rit.edu/inkml_viewer

Nine research groups registered to participate in the competition, each of which received the training dataset. Eight of these groups entered a system in the competition (see Table II). Two of the submitted systems were created by organizing labs (Univ. Nantes and RIT), leaving six systems eligible to win. Three of the participating groups competed in CROHME for the first time this year.

We summarize the participating systems in the next section, and then describe the competition protocol and results in the following sections.

II. System Descriptions

A description for each system that participated in CROHME 2013 (see Table II) is provided below. Participants were permitted again this year to use additional training data, provided that they informed the organizers; Table II indicates which training data set(s) were used by each group.

System I (Czech Tech. Univ.): MFR [5], [6] was developed at the Center of Machine Perception of Czech Technical University. Its preliminary version has been submitted to CROHME 2013. MFR recognizes online handwritten formulas in three independent phases for stroke segmentation, symbol recognition and structural analysis. Stroke segmentation identifies close stroke groups that satisfy several spatial constraints. Strokes may belong to multiple segments, leading to multiple segmentation hypotheses. Each segment is recognized by a classifier that combines template matching based on an elastic distance of strokes with a publicly available third party neural network classifier. Each symbol candidate is labeled by one or two characters and assigned a real-valued belief. The template matching based classifier is defined using only the CROHME 2013 training data set.

Ambiguities originating in segmentation and symbol recognition are resolved by the structural analyzer. It constructs multiple derivation trees bottom-up for a manually defined 2D context-free grammar. Derivation trees are ranked based on beliefs for symbol hypotheses and their spatial relations. These are determined by a statistical model learned from the competition training data. The tree with the highest belief is selected and converted to TeX or MathML.

System-II (Tokyo Univ.): The Nakagawa Laboratory of Tokyo University of Agriculture and Technology entered a system with three main modules: (i) segmentation based on symbol hypotheses net [7], (ii) symbol recognition based on combining online and offline methods [8], (iii) structural analysis of mathematical expressions based on a Stochastic Context Free Grammar. The grammar is written in Chomsky Normal Form, and each production is associated with a geometric relation. The CYK parser is used to find the most likely expression candidate for the input handwritten strokes. Three-fourths of the CROHME training data (6000 MEs) was used for training, and the remainder of the data to test the system during development.

System-III (Univ. São Paulo): This is an early prototype developed by Frank D. Julca-Aguilar at the University of São Paulo as part of his doctoral studies. Symbol segmentation and classification are formulated as a joint optimization problem, in which a template-based classifier assigns costs to candidate stroke partitions. Temporal and spatial constraints are used to reduce the number of candidate partitions. The chosen symbol segmentation and classification hypotheses are given by the partition with lowest cost. Structural analysis is performed by recursively extracting baselines [9], using contextual constraints to guide the selection of symbols on a baseline. In addition to the CROHME dataset, the system has been trained using samples from the ExpressMatch dataset [10]. The ExpressMatch dataset is composed of 468 expressions and a total of 5600 symbols.

System-IV (Univ. Valencia): This system is based on stochastic parsing of 2D context-free grammars [11]. Given a sequence of strokes, the parser builds several hypotheses for segmentation, symbol recognition and the structural relations among them. Symbol classification is performed by a Bidirectional Long Short-Term Memory Recurrent Neural Network (BLSTM-RNN) using on-line features, and spatial relations classification is carried out by Support Vector Machines (SVM) using geometric features. After the parsing process, the most likely hypothesis that accounts for the whole expression represents the recognized formula. The system uses both grammar constraints and contextual information to solve jointly all the problems involved in mathematical expression recognition.

System-V (RIT): Symbol segmentation considers strokes in time series, using a binary AdaBoost classifier to determine which stroke pairs to merge [12]. Symbol classification is performed by a set of boosted C4.5 decision trees obtained using AdaBoost.M1 [13]. Input strokes are resampled using cubic splines, with features including number of strokes, aspect ratio, covariance matrix of sample points, fuzzy histograms of line orientations, and a set of line crossings/cross-counts in different orientations. This produces a single symbol set.

The parser is a variation of Eto and Suzuki's Minimum-Spanning Tree (MST) based algorithm [14]. The parser recursively: 1) groups vertical structures (e.g. fractions, summations and square roots), 2) extracts the dominant operator (e.g. fraction line) in each vertical group, and then 3) locates symbols on the main baseline, and on the main baselines in superscripted and subscripted regions by finding an MST defined

TABLE II
PARTICIPATING GROUPS AND TRAINING DATA SETS USED BY EACH.
'ADDITIONAL' SIGNIFIES USED OF TRAINING DATA IN ADDITION TO
CROHME, AND 'OTHER' INDICATES USE OF A NON-CROHME DATA SET

System	Affiliation		Training
∗ I	Czech Technical Univ.	(Czech Rep.)	CROHME
∗ II	Tokyo Univ. of Agr. & Tech.	(Japan)	CROHME
∗ III	Univ. of São Paulo	(Brazil)	Additional
IV	Univ. of Valencia	(Spain)	CROHME
∗∗ V	Rochester Institute of Techn.	(USA)	CROHME
VI	Sabanci Univ.	(Turkey)	CROHME
VII	Vision Objects	(France)	Other
∗∗ VIII	Univ. of Nantes	(France)	CROHME

∗ : New CROHME participant   ∗∗ : Organizing lab

over candidate symbol pairs with their associated classes and spatial relationship, and then 4) repeats the procedure in nested regions of vertical structures (e.g. fraction numerators and denominators). Spatial relationships are classified as inline, superscript or subscript by a Support Vector Machine, using bounding box geometry and a shape context feature for the region around symbol pairs.

System-VI (Sabanci Univ.): The system from Sabanci University, Turkey, uses a 2D-stochastic grammar for parsing [15]. A grammar rule is applicable for a given set of tokens (initially, a set of recognized symbols) if the applicability predicate of the rule decides that the required relationship (up, down, inside etc.) exists between the given tokens. 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 a generated token based on the likelihood of the component symbols and the likelihood of the spatial relationship between the component symbols. In case of a subscript, the likelihood value depends on the relative position and size of the two tokens.

During parsing production rules are applied in an arbitrary order, because the system generates all legal interpretations of a given input string, along with their likelihoods. The system can be slow for longer expressions, as it keeps track of all possible interpretations of neighboring tokens. In future work, we will look at speeding up the system by expanding the most likely tokens first.

System-VII (Vision Objects): The MyScript Equation recognizer 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 produce the best candidate interpretation.

The equation recognition engine analyzes spatial relationships between all the parts of the equation, in conformance with a grammar, to determine the segmentation of all its parts. Each grammar rule is 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 system 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, using 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. The recognizer has been trained on writing samples that have been collected from writers in several countries.

This system was not trained using the CROHME training data set, but instead using roughly 30000 math expressions in a private corpus.

System-VIII (Univ. Nantes): This system simultaneously optimizes expression segmentation, symbol recognition, and 2-D structure recognition under the restriction of an expression grammar [16]. This strategy is adopted to prevent errors from propagating from one step to another. The approach transforms the recognition problem into a search for 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 invalid segmentation hypotheses, unlike most existing systems. The originality of this system stems from the global learning schema, which allows training a symbol classifier directly from mathematical expressions. The advantage of this global learning is to consider the 'junk' examples (i.e. invalid symbol segments) and include them in the training set for the symbol classifier's reject class. Furthermore, contextual modeling based on structural analysis of the expression is employed, where the models are learnt directly from expressions using a global learning scheme. This global approach is applicable for other 2-D languages, for example, flowchart recognition [17].

III. COMPETITION PROTOCOL AND DATA

Systems entered in the competition were required to recognize expressions consistent with two different expression grammars, which defined the set of legal symbol layouts. The first was the most complex expression grammar used in last year's competition (**Part 3**, from CROHME 2012), and the second was a new grammar, which included many new symbols (101 classes), including additional Latin and Greek letters, prime ($\prime$), pipe ($|$). We also added a new layout structure, for n-th roots (e.g. $\sqrt[3]{2}$). This more complex grammar is referred to as **Part 4** (CROHME 2013).

Both grammars include vertical layout structures such as fractions, square roots, subscripts, superscripts, and limits above and/or below summations and integrals. Grid-based and tabular structures such as matrices, choice notation, and cases in function definitions have not been included in CROHME to date. XML and human-readable representations of these grammars were provided to participants, and are included in the CROHMELib library. A java-based parser was also provided, to allow participants to check whether recognized expressions were consistent with one of these grammars.

Systems were ranked using data collected for the more challenging **Part 4** grammar. The results of the system for the **Part 3** grammar are also considered, as a way to informally measure progress in handwritten math recognition.

The training data was distributed over two months before the evaluation of the systems. Instead of distributing the test dataset, participating teams submitted their systems to the organizers, who then ran the systems on the test set.

A. Datasets and Expression Encodings

The training data for **Part 3** (1341 expressions) is a subset of the training data for **Part 4** (8836 expressions), but the two test sets are distinct from one another and both training sets. **Part 3**'s test set contains 486 expressions, and **Part 4**'s contains 671 expressions. In both cases the test expressions were collected from writers who were not included in the training data, making the task *writer-independent*, and as a result quite challenging. Training and test data for CROHME 2012 along with related tools were available from the International Association of Pattern Recognition (IAPR)[4].

For Part 4, the training data includes thousands of expressions from existing handwritten expression datasets, including (i) MathBrush (University of Waterloo) [18], (ii) HAMEX (University of Nantes) [19], (iii) MfrDB (Czech Technical University) [5], (iv) ExpressMatch (University of Sao Paulo) [20] and (v) the KAIST dataset. Due to differences in legal symbols and layouts, not all expressions in these data sets were consistent with the Part 4 grammar.

Participants' systems could produce CROHME InkML files or Comma Separated Variable (CSV) files representing label graphs [4]. CROHME InkML files describe how strokes are grouped into symbols and classified, along with a Presentation MathML representation of symbol layout. This is the encoding that participants could use to view stroke data and recognition results. A perl-based evaluation script was used to produce Table IV from CROHME InkML files.

Label graphs represent structure at the stroke level. Each stroke is represented by a node labeled with the class of its associated symbol, and all stroke pairs are associated by a pair of labeled directed edges. Edge labels represent whether two strokes are unrelated, belong to the same symbol, or belong to two symbols in a spatial relationship (relations: Right, Above, Below, Inside (square root), Superscript, or Subscript). The LgEval library was used to compare label graph files.

These two encodings provide us with complementary structure representations. The use of the two representations was helpful for debugging our evaluation scripts, as it allowed us to compare their results.

IV. RESULTS

We report results below, first for the **Part 3** data set to provide an informal measure of progress, and then for the new **Part 4 (2013)** data set, which was used to rank systems.

A. Part 3 (CROHME 2012 Test Set)

First, consider the evolution of the systems with regard to the hardest task from last year's competition. Table III shows that all systems that participated last year have improved in performance. This is an encouraging sign of progress, tempered to some degree by the possibility of (unintentional) fitting of the test set by participants. Amongst the new participants, system II performs very well in comparison to the other systems.

TABLE III
PERFORMANCE ON CROHME 2012 TEST SET (PART 3; 486 EXPRESSIONS). SYSTEMS DIFFER BETWEEN 2012 AND 2013

	2012	2013
VII	62.50	70.16
IV	22.75	24.90
II	-	23.46
VI	4.92	11.93
III	-	6.38
I	-	3.91
VIII	25.61	32.92
V	9.43	12.55

B. Part 4 (CROHME 2013 Test Set)

Comparisons were made at three different levels of abstraction: the expression level (Table IV), symbol level (Table V) and stroke level (Table VI). Table IV shows the expression recognition rate for each system, with systems sorted in decreasing order of their correct expression rate. Also shown are the number of expressions correctly recognized when 1 to 3 errors in symbol or MathML layout tag labels (e.g. msubsup, munder) are permitted. Allowing just one labeling error, all rates increase substantially, in many cases roughly doubling the expression rate.

TABLE IV
EXPRESSION-LEVEL EVALUATION, SHOWING $\leq 0\ldots3$ SYMBOL OR MathML STRUCTURE LABEL ERRORS (671 EXPRESSIONS)

	Correct (%)	≤ 1 error	≤ 2 errors	≤ 3 errors
VII	60.36	80.33	84.95	86.14
IV	23.40	37.85	44.71	47.84
II	19.97	34.13	40.83	42.92
III	9.39	18.48	24.14	27.27
VI	8.35	19.08	24.44	26.23
I	2.68	9.69	16.24	20.72
VIII	18.33	32.04	40.24	42.92
V	14.31	24.74	32.19	36.21

Table V provides recall and precision metrics for symbol segmentation, classification, and spatial relationship detection. Metrics are provided for symbol detection ('Segments') and detection of symbols with their correct classification ('Seg+Class'). A correct spatial relationship between two symbols requires that both symbols are correctly segmented and in the right relationship. For spatial relationships, we present both detection metrics for edges in the symbol layout trees ('Tree Rels.') along with these metrics in Directed Acyclic Graphs (DAGs) obtained by inheriting relationships down the layout tree, for example representing for x^{2a} that both the 2 and a are superscripted [4]. As seen last year, detecting spatial relationships was more challenging for participants than symbol detection and classification. Note that amongst the systems, some have stronger spatial relationship detection but weaker symbol detection than other systems (e.g. System II vs. System III).

Table VI shows the stroke level evaluation for segmentation, classification and relation analysis. For stroke pairs, the number of segmentation vs. relationship errors made by

TABLE V
Symbol-level Evaluation. Systems sorted by decreasing recall for correct symbol segmentation and classification (Seg+Class) (6082 symbols; 5409 layout tree edges; 29525 DAG edges)

	Segments (%)		Seg+Class (%)		Tree Rels. (%)		DAG Rels. (%)	
	Recall	Prec.	Recall	Prec.	Recall	Prec.	Recall	Prec.
VII	97.86	98.14	93.03	93.29	88.65	88.93	95.18	95.46
IV	84.97	87.08	73.94	75.77	49.73	51.48	76.34	79.86
II	80.70	86.35	66.41	71.06	22.44	27.00	45.78	63.00
III	85.24	77.93	62.63	57.26	53.24	48.17	88.46	78.34
VI	57.86	47.33	47.68	39.00	33.63	49.54	31.83	69.99
I	46.93	38.43	25.19	20.63	24.85	42.44	33.67	71.63
VIII	90.32	86.86	73.84	71.02	50.19	49.26	73.01	77.73
V	84.45	86.46	66.66	68.25	41.34	42.44	72.60	74.31

each system are shown. Also shown are summary metrics (whose definitions are available elsewhere [4]), for the average percentage of mislabeled node and edge labels per expression (ΔB_n) along with a variation designed to weight segmentation, classification and parsing decisions more equally (ΔE). Both are normalized by the number of strokes in each expression, e.g. for the best system on average 2.43% of the strokes and stroke pair labels in each expression are incorrect (4.27% in the re-weighed ΔE). These measures provide a fairer comparison between systems in terms of the number of stroke-level decisions that need to be made, without a thresholding effect caused by ignoring correct stroke-level decisions within incorrectly detected symbols and relationships at the symbol or expression levels. We can see that all systems mislabel at most 26.82% of the labels in a label graph for an expression incorrectly on average (42% in the re-weighted ΔE).

TABLE VI
Stroke-level evaluation. Systems sorted by increasing stroke label errors/ΔE. ΔB_n and ΔE are measured on directed label graphs (8548 strokes; 81007 (undirected) stroke pairs)

	Label Hamming Distances				μ error (%)	
	Strokes	Pairs	Seg	Rel	ΔB_n	ΔE
VII	537	1777	170	1607	2.43	4.27
IV	2187	9493	1201	8292	10.05	18.31
II	2748	19768	1527	18241	13.87	22.04
III	3415	15135	1262	13873	15.01	26.21
VI	4768	43893	5094	38799	27.60	36.67
I	6543	41295	5849	35446	26.82	41.63
VIII	2302	15644	4945	10699	12.06	19.30
V	2898	10803	1228	9575	12.71	22.80

V. Outcome

Eight groups submitted systems for the CROHME competition, with two submitted by a couple of the organizing labs. For this year's competition, we have selected two winners: 1) VisionObjects, which produced the strongest system using a large private training set, and 2) the University of Valencia, whose team produced the strongest system trained using only the CROHME training data set.

This systems submitted to this year's competition performed at similar levels of accuracy as last year, but on a harder task, with more symbols and structures in the expression grammar for Part 4. In the process, we have created new data sets and evaluation tools for the research community.

References

[1] 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:1497-1500, 2011.

[2] H. Mouchère, C. Viard-Gaudin, D. H. Kim, J. H. Kim and U. Garain, *ICFHR 2012 Competition on Recognition of On-Line Mathematical Expressions (CROHME 2012)*, in Proc. ICFHR: 811-816, 2012.

[3] R. Zanibbi and D. Blostein, *Recognition and Retrieval of Mathematical Expressions*, Int'l. J. Document Analysis and Recognition (IJDAR), 15(4):331-357, 2012.

[4] R. Zanibbi, H. Mouchère, C. Viard-Gaudin, *Evaluating structural pattern recognition for handwritten math via primitive label graphs*, in Proc. Document Recognition and Retrieval (DRR), 2013.

[5] J. Stria, M. Bresler, D. Průša and V. Hlaváč, *MfrDB: Database of Annotated On-line Mathematical Formulae*, in Proc. ICFHR:540-545, 2012.

[6] J. Stria and D. Průša, *Web Application for Recognition of Mathematical Formulas*, in Proc. 11th Conf. on Theory and Practice of Information Technologies (ITAT 2011), Vrátna dolina, Slovak Republic, 47-54, 2011.

[7] A.D. Le and M. Nakagawa, *A tool for ground-truthing online handwritten mathematical expressions*, in Proc. 16th International Graphonomics Society Conference (IGS), Nara, Japan, 2013.

[8] B. Zhu, J. Gao and M. Nakagawa, *Objective Function Design for MCE-Based Combination of On-line and Off-line Character Recognizers for On-line Handwritten Japanese Text Recognition*, in Proc. ICDAR:594-599, 2011.

[9] R. Zanibbi, D. Blostein and J. R. Cordy, *Recognizing mathematical expressions using tree transformation*, in IEEE Trans. PAMI, 24(11):14551467, 2002.

[10] F.D.J. Aguilar and N.S.T. Hirata, *ExpressMatch: a system for creating ground-truthed datasets of online mathematical expressions*, in Proc. 9th IAPR International Workshop on Document Analysis Systems (DAS), 2012.

[11] F. Álvaro and J.A. Sánchez and J.M. Benedí, *Recognition of on-line handwritten mathematical expressions using 2D stochastic context-free grammars and hidden Markov models*, in Pattern Rec. Letters, 2012.

[12] L. Hu and R. Zanibbi, *Segmenting handwritten math symbols using AdaBoost and multi-scale shape context features*, Proc. ICDAR 2013, to appear.

[13] R.E. Schapire and Y. Freund, *Boosting: Foundations and Algorithms*, The MIT Press, 2012.

[14] Y. Eto and M. Suzuki, *Mathematical formula recognition using virtual link network*, in Proc. ICDAR:762-767, 2001.

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

[16] A.-M. Awal, H. Mouchère and C. Viard-Gaudin, *A global learning approach for an online handwritten mathematical expression recognition system*, in Pattern Recognition Letters, 2012.

[17] A.-M. Awal, G. Feng, G., H. Mouchère and C. Viard-Gaudin, *First Experiments on a new Online Handwritten Flowchart Database*, in Proc. Document Recognition and Retrieval XVIII, 2011.

[18] S. MacLean, G. Labahn, E. Lank, M. Marzouk, and D. Tausky, *Grammar-based techniques for creating ground-truthed sketch corpora*, IJDAR 14(1): 1-21, 2011.

[19] S. Quiniou, H. Mouchère, S.P. Saldarriaga, C. Viard-Gaudin, E. Morin, S. Petitrenaud and S. Medjkoune, *HAMEX - A Handwritten and Audio Dataset of Mathematical Expressions*, in Proc. ICDAR: 452-456, 2011.

[20] F.D.J. Aguilar and N.S.T. Hirata, *ExpressMatch: A System for Creating Ground-Truthed Datasets of Online Mathematical Expressions*, in Proc. 9th IAPR International Workshop on Document Analysis Systems (DAS):155-159, 2012.