Fuzzy Visibility Graph for Structural Analysis of Online Handwritten Mathematical Expressions

Arnaud Lods, Eric Anquetil, Sébastien Macé

Fuzzy Visibility Graph for Structural Analysis of Online Handwritten Mathematical Expressions

Arnaud Lods
Learn&Go, Univ Rennes, CNRS, IRISA
F-35000 Rennes
arnaud.lods@learn-and-go.com

Éric Anquetil
Univ Rennes, CNRS, IRISA
F-35000 Rennes
eric.anquetil@irisa.fr

Sébastien Macé
Learn&Go
F-35000 Rennes
sebastien.mace@learn-and-go.com

***Abstract*—This paper presents a fuzzy visibility graph representation for handwritten mathematical expressions (HME) computed over segmented symbols using learned fuzzy landscape (FL) models. The learned FL models define the relative positioning of a pair of symbols using both their morphology, their typology and their context. A Random Forest Classifier uses this relative positioning to qualify relationships between symbols. The valued fuzzy visibility graph with the FL membership is produced from this classifier's output. This graph offers an explicit representation of the HME bi-dimensional structure which is then parsed with a set of rules to produce the recognized HME. We evaluate the performance of this system on the task of HME structure recognition using provided segmented symbols with experimental results on both CROHME 2014 and 2016 datasets. We obtain results up to par with the state-of-the-art thus proving that our fuzzy visibility graphs are a strong representation for mathematical expression parsing.**

***Index Terms*—fuzzy visibility graphs, fuzzy landscapes, handwritten mathematical expression, structure recognition, structure analysis.**

I. Introduction

The recognition of handwritten mathematical expression (HME) is both a challenging and interesting research domain. A system able to solve this task spares users to learn specific presentational markup language, such as LaTeX, to input mathematical expression on a computer using a keyboard. The recognition task is separated in three sub-tasks: symbol segmentation, symbol classification and structure recognition. From a set of input strokes received using an adapted interface such as a pen-based tablet, the goal is to produce the best representation of the 2D mathematical structure.

The domain of education is an interesting context of application for such a system. More precisely, we are interested by the early mathematical education of elementary children when learning arithmetical operation. Fig 1 shows an 8-year old child solving two subtractions using a pen-based tablet. When children are still learning the basics while solving problems given by a teacher, they are prone to make mistakes. For instance, it can be a simple calculus errors (Fig 1 right operation contains such error), a symbol misplacement or the omission of an algorithmic step (forgetting to report a carry-over). On one side, we want to cover the recognition task by translating the input into a valid operation to confront the result. On the other side, we want to analyze the input to give an appropriate feedback on mistakes. To complete this task, we need a representation of the bi-dimensional arithmetic operation structure easy to parse to extract a valid operation and with valued explicit features between related symbols that can be used to analyze the symbols layout. In this paper we keep the focus on the task of HME recognition from a set of segmented symbols.

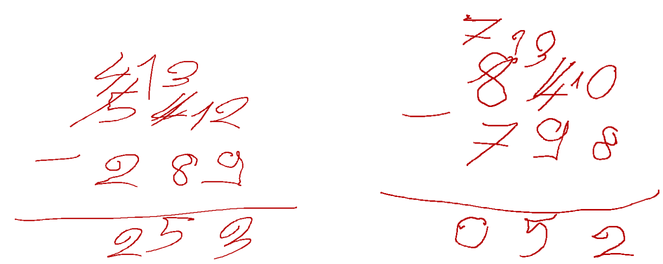


Fig. 1: Substractions solved by an 8-year old child.

We propose to tackle this problematic with Fuzzy Visibility Graph (FVG) by extending the notion of visibility graph using fuzzy landscapes (FL) models that are able to create human-like representation of learned relative positioning in the space. FL were first proposed to be used on pictures [1] for medical uses and then later expanded with learned models on online signals [2] for Chinese symbols. The fuzzy membership of a FL model describe the belonging of an object to a specific learned area, in our case the match of a symbol to a learned mathematical relation in regards to another symbol. Such models are both explicit and qualitative enough to both understand a child input and give visual feedback. These models in addition to symbols typology represents strong features for mathematical relationship classification, and graph deduced from them can be easily parsed for the structure recognition task.

The rest of the paper is structured as follow. First we present the state-of-the-art systems tackling the problematic of HME recognition in Section II. Then we present our FVG representation in Section III. In Section IV we present our system's performances on the structure recognition rate from sets of symbols on both public available dataset CROHME 2014 [3] and CROHME 2016 [4] before presenting a quick overlook of the use of FVG representation for the analysis of a given arithmetic expression. We conclude our work and the system's future uses in Section V.

II. Related works

We present a brief overview of the state-of-the-art. The three tasks of HME recognition were either solved with sequential or integrated solutions. Their goal is to produce a Symbol Layout Tree (SLT) that represents the formula. In addition, recent end-to-end neural network were proposed that directly translate the expression into a LaTeXformula.

*1) **End-to-end neural network**:* These systems make use of recent improvements in machine translation using Encoder Decoder with deep network [5]. Such system was adapted by Deng et al. [6] for the recognition of offline HME. A Convolutional Neural Network extracts a feature map which is then encoded by a recurrent neural network (RNN) and decoded by another RNN. Their system uses the recognized context while parsing the expression. Zhang et al. proposed in [7]–[9] several improvements of an initial system with the same initial architecture with an attention-based parser to generate the sequence. They use a coverage model to avoid over and under parsing and use Multi-Scale Attention with a DenseNet encoder by adding temporal attention and combining offline and online features. Though getting the best scores on the overall task of HME recognition, the formula cannot be used for layout analysis for our task.

*2) **Sequential solutions**:* Eto et al. [10] use **k-Nearest Neighbors** graphs to join strokes nearby in spaces. Centers of symbols are used for relative positioning, and link are affected a cost. To recover the correct expression the path with the lowest cost is chosen. Lei et al. [11] propose a **Line-of-Sight** graph to construct a visibility graph. Vision angles from the center of a stroke to all other strokes' convex hulls is computed and the direct line of sight are conserved. Symbols relationship are determined using Shape Context (SC) features and the resulting graph is parsed [12] using Edmond's algorithm to find the SLT. Broader concept of visibility can cover all related symbols but also put in relation plenty of unrelated symbols. Moreover, the features used for relationship classification are not explicit enough for the layout analysis we want to produce.

*3) **Integrated solutions**:* Most integrated solutions use grammar and parsing algorithm to complete all three tasks concurrently. Le et al. [13] use **time-series** with a 2D Stochastic-Context-Free Grammar (SCFG) and a modified Cocke-Younger-Kasami (CYK) to parse the graph. By limiting the search space to adjacent strokes in time, they greatly reduce the search space complexity and it was improved in [14] by using a XY reordering to reorder delayed strokes. Other papers use **relative positioning** around a given stroke: given its bounding-box the search space is separated in different rectangular areas, each one representing a mathematical relationship. Alvaro et al. [15] use a Support-Machine-Vector (SVM) to determine the structural relationship between symbols from geometric features extracted from bounding boxes. Julca-Aguilar et al. [16] generate hypothesis graphs symbols and a graph grammar using a top-down algorithm to parse and identify the sub-graph with the best interpretation. The mathematical expression is parsed with a SCFG grammar and the CYK algorithm. Zhang et al. [17] generate hypothesis graphs by using relative positioning, time gap and direct line of sight. A Bidirectional Long Short-Term Memory neural network produces the resulting SLT graph. In these cases the search space becomes either too restricted with temporal information to capture all related symbols or the features used for the relative positioning are too general and not explicit enough for later analysis.

To solve the features disadvantages used in previous works, we propose the use of a more adapted set of features using FL to classify relationship between symbols and to valuate of our FVG with explicit features for layout analysis.

III. Fuzzy visibility graph

The features used to restrict the search space in Section II are able to construct a valid mathematical expression but limit the graph representation. Using the time information, we may ignore related symbols in space. The visibility defined by Line-of-Sight graphs can link unrelated symbols. Features computed using bounding-boxes or strict rectangular areas are not explicit enough to define precise mathematical relation given symbols of different morphology. In this section, we present the use of FL models to compute membership value to specific learned area so that the unique morphology of each symbol is taken into account. Using such features, we can link all related symbols one to another without limiting our representation. Moreover, we can value our graph with explicit features that can then be directly used for analysis. An overview of the system is shown in Fig 2.

The fuzzy landscapes were first introduced for images by Bloch et al. [1] and used in the medical field to match objects relative positions with different morphology given a specific direction. The goal is to approach the human vision of the search space to determine the relative positioning of two complex objects without using a simplified representation such as the minimal bounding box that can group all points of an object. The adequacy of a point p with a reference point o defined by the Equation 1 in regards to a specific direction represented by the vector $\vec{\mu_\alpha}$ is given by the angular difference between the angle vector defined by the two points and the selected direction vector. Using this equation, we can compute the adequacy of the point p for a reference object R by being the best adequacy in regards to all points composing R given a specific direction vector (Equation 2). The adequacy of an object A for a reference object R is defined as the mean of each adequacy point (Equation 3).

$$v_\alpha(p) = max(0, 1 - \frac{2}{\pi}.arccos(\frac{\vec{op}.\vec{\mu_\alpha}}{\vec{op}})), \forall p \in S \quad (1)$$

$$\mu^R(p) = max_{q \in R} v(p-q), \forall p \in S \quad (2)$$

$$M^R(A) = \frac{1}{|A|}\sum_{p \in S} \mu^R(p) \quad (3)$$

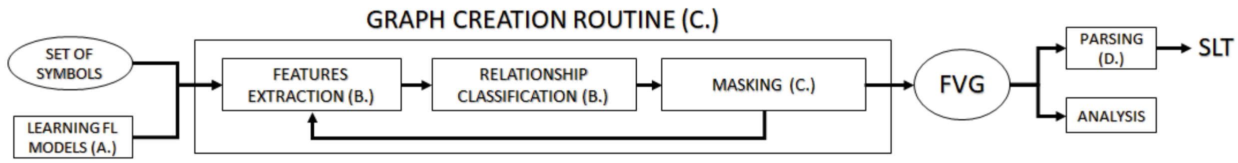
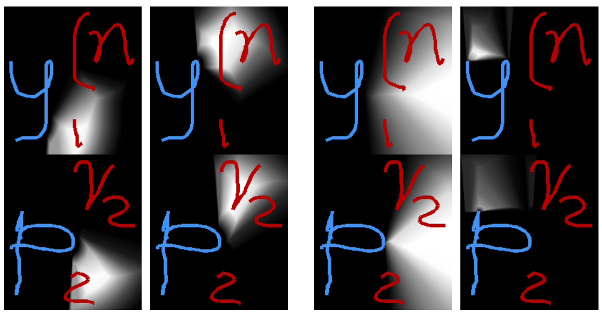


Fig. 2: Overview of our system. Using a set of segmented and classified symbols and a set of FL models learned as the input, a FVG is created from the models and additional features. The FVG is parsed to obtain the SLT.



(a) Subscript (b) Supscript (c) Right (d) Above

Fig. 3: learned FL models for different mathematical relationships for two symbols with different morphology.

Delaye et al. [2] extends the notion of fuzzy landscape on online data for Chinese symbols. The goal is to model complex relationships instead of using only relative positioning for one direction. In the context of recognition of HME, the mathematical relationship identified are $[Right, Below, Above, Inside, Superscript, Subscript]$. For a given relationship and a given direction, a new FL is learned. The space is divided into angular areas, and a threshold is used to adjust the membership of points in each area to fit the observed distribution of points in said area. The output is a new FL for the considered direction that corresponds to the relationship observed. Once the FL for the four cardinal directions (up, down, left, right) and a distance FL are learned for a relationship, a conjunction of each adequacy computed by Equation 4 are combined using a fuzzy conjunctive operator (t-norm) to compute the adequacy of an object for the model.

$$a^R(A) = T(\mu^R(A), v^R(A)) \tag{4}$$

Fig 3 shows the application of this work for different mathematical relationships with two different symbols. A point with a perfect adequacy for the considered relationship in regards to a specific symbol is white. We can observe that the morphology of both symbols produces two similar yet different acceptable area in the space. We propose to extend the notion of graph of visibility by using these FL models learned to represent mathematical relations to construct the FVG. The goal is to keep all related symbols linked one to another with their explicit membership values to each FL models for later layout analysis and provide a graph representation that is easily parsed to produce the valid SLT.

A. Learning fuzzy landscape models

Initially 6 different FL models are learned corresponding to the 6 mathematical relationships defined: $[Right, Below, Above, Inside, Superscript, Subscript]$. Fig. 3 shows the Subscript, Superscript, Right and Above models learned and applied to two symbols. Using Eq. 4, for each pair of symbols we compute the adequacy of the second symbol in regards to the first for each FL model. It produces a vector of features of size 6. For instance given the first pair y, i, the corresponding vector computed is Right: 0.30, Below: 0.0, Above: 0.0, Inside: 0.0, Superscript: 0.0, Subscript: 0.93. This feature vector will be fed to the classifier to determine the relationship between the pair of symbols.

The nature of the relations observed and of each unique pair of symbols make such models too large to correctly model such precise relationship. This is due to both the high variability between different users' inputs as well as the high variability of morphology of symbols. We define three different sets of symbols based on their morphology: Ascendant (b, h ...), Descendant (y, p ...) and Centered (a, r ...) and learn specific FL models for each of these unique relations: from the initial $Right$ relationship, we learn 9 refined relationships: $RightAscendantAscendant, RightAscendantDescendant...$. The same variability between pair of symbols is observed for $Subscript$ and $Superscript$ relationship, thus we also learn new refined models for them.

In the same way, both $Below$ and $Above$ relationships observed vary depending on the parent symbols: fraction bar, $\sum$, $\lim$ ... the former model being more expanded in the up or down direction than the later. We also learn specific models for the pairs to learn more adapted models for these specific relationships.

This learning process refine the different FL models initially learned, thus transforming our features vector of adequacy of size 6 to a feature vector or size 35, each feature corresponding to the adequacy computed for the pair of symbols considered for a given FL model. These features will be fed to the classifier to determine the presence or not of a relationship between two symbols.

B. Features extraction

FL models membership: As said previously, a features vector of size 35 is extracted from the pair of symbols using all FL models learned. Relative positioning alone may misclassify ambiguous relationship given the relative positioning due to the high variability from users for given pair of symbols. Two pair of different symbols can fit unlikely models because of misplacement from the user. A 2 can be placed to fit the $Superscript$ model of a symbol $+$, yet such relation is unlikely. Other pair of symbols have higher probability of being as subscript than right relationship. We define a set of features to make use of the typology of symbols. Using such information will help the classifier to label the edges between symbols with the correct relationship while still keeping the information of fitted models as edges' values.

Symbols typology: We define 8 different symbols typology observed: [Number, Lowercase letter, Uppercase letter, Operator, Geometrical operator, Big operator, Parenthesis, Greek letter]. For both symbol of a given pair of symbols, we extract a vector of size 8: each value is discrete and set either as 0 or 1 corresponding to the sub-class of the considered symbol. Thus, a features vector of size 16 is produced and added to the previous 35 features to help determine conflicting relationship base on relative positioning alone.

Geometrical features: In addition to the previous mentioned features, we also use geometric features to refine relationship classification: distance and offset of center of bounding box and center of mass, width and height difference.

C. Graph creation routine

The Algorithm 1 presents the graph routine creation. Using the previous described features, we train a classifier to determine the relationship between a pair of symbols among the six different mathematical existing relationships. To discriminate between symbols with and without relations, we add a supplementary class $Junk$ that is learned using all pair of symbols in the ground-truth that do not have identified relations.

To introduce the notion of visibility, in a similar fashion to [11], we use previous related symbols to further mask the search space. To take into account the context of the pair of symbols searched, we propose the addition of a mask in the fuzzy membership computation of a symbol. All symbols that were defined as having a relationship with the current symbol s are kept in memory and their body act as a mask for further away symbols. The value membership of a point p is now computed using Eq. 5. f is a mask of value $m \in [0,1]$ activated when the vector $\vec{op}$ is inside the set of angular areas masked O by previous related symbols (Eq. 6). For a given symbol s, the other symbols are sorted by the minimal distance between all of their points. The learned classifier is used to determine the relationship between the symbols. If a relationship is detected ($\neq Junk$), a directed labeled edge is created between the two symbols with the detected relationship and the FL models' membership computed are kept for later analysis. The vision angles from the parent symbols to the related symbol are computed and kept in the set of obstructed angles used for membership computation.

$$v_\alpha(p) = max(0, 1 - \frac{2}{\pi}.arccos(\frac{\vec{op}.\vec{\mu_\alpha}}{\vec{op}}) - f), \forall p \in S \quad (5)$$

$$f = \begin{cases} if \vec{op} \in O & m \\ else & 0 \end{cases} \quad (6)$$

Algorithm 1 Fuzzy visibility graph construction

INPUT: E (empty set of edges), S (set of symbols), F (set of FL)
for $s \in S$ **do**
  blocked_angles = {}
  for $t \in S \neq s$ ordered by minimal distance **do**
    for $f \in F$ **do**
      $v = 0$
      for $point \in t$ **do**
        $v = Equation4$
      $membership[f] += Equation3$
    $features = Ref$III-B
    if $relation(features) \neq JUNK$ **then**
      $E+ = (s, t, relation)$
      $update\_blocked\_angles(blocked\_angles, relation)$

D. Parsing

Figure 4b shows an example of straightforward parsing to construct the SLT representing the mathematical formula. The FVG is constructed over segmented symbols with edges labeled with the detected relationships. The parent symbol, which is the leftmost symbol on the main baseline, is the symbol that does not have any directed edges toward it. The edges labeled as $[Below, Above, Inside]$ identify sub-expressions. These sub-expressions are parsed using the following rules:

Rule 1: Child symbols of sub-expressions $[Below, Above, Inside]$ are treated as sub-graphs. All edges with outside nodes are removed (Fig 4b).

Rule 2: The parent node of the directed sub-graph extracted from [Below, Above, Inside] is kept. All other edges from the main symbol to the child symbols are removed (Fig 4c).

Now that the redundant relationships for sub-expressions are removed, we can parse the remaining edges in the same manner.

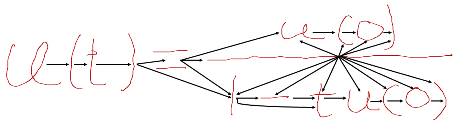
Rule 3: For each node with a duplicate relationships [Right, Subscript, Superscript], only the edge to the nearest symbol related is kept (Fig 4d).

Rule 4: For each node with two or more parents, edges from nodes in a child relationship are removed.

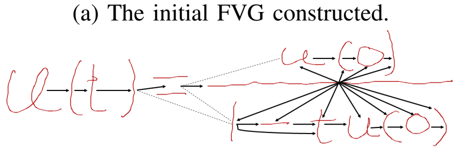
Parsing the graph with these rules assures that no symbol has two directed edges toward it and no symbols has two directed edges out of it with the same labeled relation. The resulting graph is a SLT representing the recognized formula.

IV. Experiments

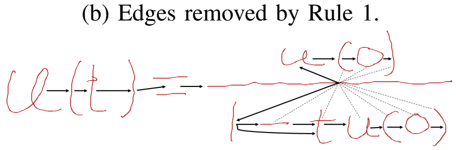
Datasets. We use data from the CROHME 2014 and CROHME 2016 competitions to experiment our system and confront our results to known systems. CROHME 2014 has 8833 training and 986 test expressions. CROHME 2016 has 9100 training and 1147 test expressions.



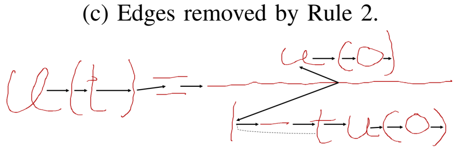
(a) The initial FVG constructed.



(b) Edges removed by Rule 1.



(c) Edges removed by Rule 2.



(d) Edges removed by Rule 3. For this HME the SLT is obtained at this step.

Fig. 4: Parsing a fuzzy visibility graph to find the SLT by consecutively applying described rules.

Task and performance metrics. CROHME 2016 [4] introduced the new sub-task of structure recognition with provided segmented and labeled symbols on the CROHME 2014 [3] datasets. The structure recognition rate (without relations labels) and the expression recognition rate with up to two errors for each expression are computed.

Training phase. For the relationship classifier, we train several Random Forest classifiers using a gridsearch and cross-validation for CROHME 2014 using 90% of the training data and keeping 10% for the validation set to find the best hyper-parameters. The number of trees $\{20, 30, 40, 50, 100, 200\}$ and the maximum depth of a tree $\{30, 40, 50, 100\}$ were tested, and the best hyper-parameter is $\{n_{tree} = 100, depth = 50\}$. Several classifiers are learned using different combinations of features to show the improvement with extended set of features (Table I): (a) features extracted from fuzzy landscapes (41 features), (b) using the topology of symbols using the output of a symbol classifier (57 features), (c) using the topology of symbols using the provided symbols' labels (57 features) and (d) adding the geometrical features (67 features). We use in (c) the information of a symbols' classifier to show that even using the output from a simple classifier, the imperfect sub-classes of symbols help the relationship classification. The classifier used is a Support-Machine-Vector learned using HBF49 features [18], which achieve an accuracy rate of 83% on CROHME 2014 isolated symbols tests. The results presented against the other competing systems in Table III use the system (d) for the sub-task of the structure recognition with provided symbols positions and labels.

Experimental Results. Table I presents the results obtained with different set of features for relationship classification. As expected, using the ground-truth symbols' labels increased the structure recognition rate, as it helps the classifier to not give impossible label to some couple of symbols. We also find out that using a simple classifier to determine the symbols typology still improves the structure recognition rate. These features can be used in a global system using previous classified symbols.

TABLE I: Structure and expression recognition rate using our system with different set of features on CROHME 2014 dataset

	Expression **Rec. Rate**
(a) FVG (41)	69.44
(b) + classified symbols (57)	72.26
(c) + ground truth symbols (57)	77.81
(d) + geometrical features (67) (our final system)	78.43

Table III presents the results using the system (d) compared to other systems that evaluated themselves on the same task of structure recognition with provided symbols. MyScript [4] achieves the best results on the structure and expression recognition rate, but use a private and much larger dataset for training. We obtain the third best results, being up-to-par with the WIRIS [4] system. As we are putting more emphasis on positioning and allow invalid relationships, our system ends up producing several invalid mathematical formulas. In this regard probabilistic grammar based methods such as WIRIS [4] are more adapted as grammar's rules only accepts valid mathematical formulas to be parsed. Using an adapted grammar on our graphs could be considered to parse and only accept valid ME.

TABLE II: Structure recognition with provided symbols on CROHME 2014

	Structure	**Structure + Labels**		
	Rec. Rate	Rec. Rate	<= 1 err	<= 2 err
MyScript* [4]	**90.67**	**84.38**	**85.90**	**87.62**
Wiris [4]	86.61	78.80	80.42	82.75
MST [12]	76.66	67.44	-	-
BLSTM [16]	69.27	64.81	67.34	70.69
CYK [13]	70.99	61.46	63.89	66.84
FVG (d)	87.76	78.43	81.54	83.28

*Large, independent training set used. Others use CROHME 2014 Train

TABLE III: Structure recognition with provided symbols on CROHME 2016

	Structure	**Structure + Labels**		
	Rec. Rate	Rec. Rate	<= 1 err	<= 2 err
FVG*	91.11	85.79	88.84	90.23

*Extended set of geometrical features used for CROHME 2016 Test set.

Fig 5 shows a FVG constructed over an arithmetic operation. The links are valued with the membership of the best FL models. Symbols are matched using a graph edit distance

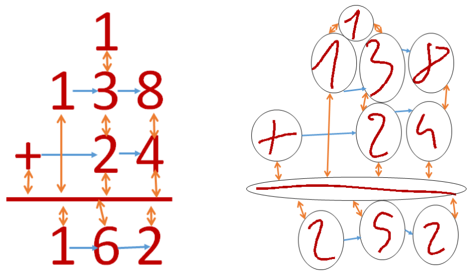


Fig. 5: Left: Expected graph from a simple arithmetic operation, Right: FVG computed over a handwritten arithmetic operation

algorithm between an expected ground-truth graph (left) and a built FVG (right). For each missing or unexpected matches on edges, associated FL are used to correct the positioning. In this example, the carryover has Above edges with two symbols. The learned Above FL for the expected symbol would be used to correct the mis-positioning. In the same way, matched symbols with different labels would be highlighted.

V. Conclusion

We present a new fuzzy visibility graph representation for handwritten mathematical expressions. Given a set of symbols, the system extracts features from learned fuzzy landscape models each representing observed mathematical relations. A Random Forest classifier determines if there is a relationship and its nature between each pair of symbols using these fuzzy memberships, geometrical features and recognized or provided symbols labels. The resulting graph is a valued fuzzy visibility graph that is easily parsed given a set of rules to remove redundant detected relations to construct the valid mathematical expression.

Given a provided set of segmented symbols with their labels, the algorithm gets results up to par with the state-of-the-art for the same task on the public dataset CROHME 2014. Using the output of a simple symbol classifier for the symbols labels, the recognition rate is still competitive.

Using such features, this valued graph is able to arrange explicitly all symbols one to another and can be used to produce a qualitative analysis of the symbols' layout. In future works the fuzzy landscapes may be refined even further depending on symbols' types and paired with a features selection strategy to keep only relevant landscapes. We will also use this graph representation in a complete system intended for arithmetical operation learning. Given a problem, the children's inputs will be confronted to expected models to puzzle out their mistakes and improve the learning process by displaying adapted visual feedbacks.

Acknowledgment

With the support from the LabCom **ScriptAndLabs** founded by the **ANR** ANR-16-LVC2-0008-01. With the support from the ANRT.

References

[1] I. Bloch, "Fuzzy relative position between objects in image processing: a morphological approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 7, pp. 657–664, 1999.

[2] A. Delaye and E. Anquetil, "Fuzzy relative positioning templates for symbol recognition," in *2011 12th International Conference on Document Analysis and Recognition (ICDAR)*, pp. 1220–1224, IEEE, 2011.

[3] H. Mouchere, C. Viard-Gaudin, R. Zanibbi, and U. Garain, "Icfhr 2014 competition on recognition of on-line handwritten mathematical expressions (crohme 2014)," in *2014 14th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pp. 791–796, IEEE, 2014.

[4] H. Mouchère, C. Viard-Gaudin, R. Zanibbi, and U. Garain, "Icfhr2016 crohme: competition on recognition of online handwritten mathematical expressions," in *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pp. 607–612, IEEE, 2016.

[5] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[6] Y. Deng, A. Kanervisto, J. Ling, and A. M. Rush, "Image-to-markup generation with coarse-to-fine attention," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 980–989, JMLR. org, 2017.

[7] J. Zhang, J. Du, S. Zhang, D. Liu, Y. Hu, J. Hu, S. Wei, and L. Dai, "Watch, attend and parse: An end-to-end neural network based approach to handwritten mathematical expression recognition," *Pattern Recognition*, vol. 71, pp. 196–206, 2017.

[8] J. Zhang, J. Du, and L. Dai, "Multi-scale attention with dense encoder for handwritten mathematical expression recognition," in *2018 24th International Conference on Pattern Recognition (ICPR)*, pp. 2245–2250, IEEE, 2018.

[9] J. Zhang, J. Du, and L. Dai, "Track, attend, and parse (tap): An end-to-end framework for online handwritten mathematical expression recognition," *IEEE Transactions on Multimedia*, vol. 21, no. 1, pp. 221–233, 2019.

[10] Y. Eto and M. Suzuki, "Mathematical formula recognition using virtual link network," in *Proceedings of Sixth International Conference on Document Analysis and Recognition*, pp. 762–767, IEEE, 2001.

[11] L. Hu and R. Zanibbi, "Line-of-sight stroke graphs and parzen shape context features for handwritten math formula representation and symbol segmentation," in *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pp. 180–186, IEEE, 2016.

[12] L. Hu and R. Zanibbi, "Mst-based visual parsing of online handwritten mathematical expressions," in *2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pp. 337–342, IEEE, 2016.

[13] A. D. Le and M. Nakagawa, "A system for recognizing online handwritten mathematical expressions by using improved structural analysis," *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 19, no. 4, pp. 305–319, 2016.

[14] A. D. Le, H. Dai Nguyen, and M. Nakagawa, "Modified xy cut for re-ordering strokes of online handwritten mathematical expressions," in *2016 12th IAPR Workshop on Document Analysis Systems (DAS)*, pp. 233–238, IEEE, 2016.

[15] F. Álvaro, J.-A. Sánchez, and J.-M. Benedí, "Recognition of on-line handwritten mathematical expressions using 2d stochastic context-free grammars and hidden markov models," *Pattern Recognition Letters*, vol. 35, pp. 58–67, 2014.

[16] F. Julca-Aguilar, H. Mouchère, C. Viard-Gaudin, and N. S. Hirata, "Top-down online handwritten mathematical expression parsing with graph grammar," in *IberoAmerican Congress on Pattern Recognition*, pp. 444–451, Springer, 2015.

[17] T. Zhang, H. Mouchère, and C. Viard-Gaudin, "A tree-blstm-based recognition system for online handwritten mathematical expressions," *Neural Computing and Applications*, pp. 1–20, 2018.

[18] A. Delaye and E. Anquetil, "Hbf49 feature set: A first unified baseline for online symbol recognition," *Pattern Recognition*, vol. 46, no. 1, pp. 117–130, 2013.