Pattern Recognition 48 (2015) 837–848

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Matching based ground-truth annotation for online handwritten mathematical expressions

Nina S.T. Hirata *, Frank D. Julca-Aguilar

Department of Computer Science, Institute of Mathematics and Statistics, University of São Paulo, 05508-090 Rua do Matão, 1010 São Paulo, Brazil

ARTICLE INFO

Article history:
Received 25 September 2013
Received in revised form
8 September 2014
Accepted 18 September 2014
Available online 28 September 2014

Keywords:
Structural pattern analysis
Shape information
Handwriting recognition
Linear assignment problem
Ground-truth annotation
Mathematical expression dataset

ABSTRACT

Assessment of mathematical expression recognition at expression level only is not sufficient to diagnose strengths and weaknesses of different recognition systems. In order to make assessment at different levels possible, large datasets annotated with ground-truth data at different levels, such as at symbol segmentation, symbol classification, symbol/sub-expression spatial relationships, baselines or whole expression levels, are needed. Creation of ground-truthed datasets of handwritten mathematical expressions is a challenging task due to the need to cope with a large variability of symbol classes, expression layouts, writing styles, among other issues including the fact that manual annotation is an error-prone procedure. We propose an expression matching approach where symbols in a transcribed expression are assigned to the corresponding symbols in the respective model expression. Matching is formulated as a simple linear assignment problem where matching cost is defined as a weighted linear combination of local (symbol) and global (structural) characteristics. Once a symbol-to-symbol assignment is computed, not only symbol labels but all other ground-truth data attached to the model expression can be automatically transferred to the transcribed expression. We use two independent large test sets to empirically evaluate the influence of the cost function terms on matching performance. Results show mean symbol assignment rates above 99% on both sets, suggesting the potential of the method as an useful tool for helping the creation of ground-truthed online mathematical expression datasets.

1. Introduction

With the advent of tablet like devices, there is an increasing interest in online recognition of handwritten mathematical expressions (MEs). Interesting and useful applications of online recognition of MEs include numerous possibilities, notably those related to inputting mathematical notation into computer systems. In the last years, active research has been carried out in all aspects related to online recognition of handwritten MEs, including symbol segmentation and recognition [1–4], structural analysis [5–7,2], integrated approaches for recognition [8–12], and recognition evaluation [13–17]. Overview of ME recognition approaches and issues can be found in [18–20].

In the field of pattern recognition, a difficulty related to evaluating recognition performance is the scarcity of large and public datasets annotated with ground-truth information. Authors frequently consider their own datasets, making difficult the tasks of reproducing reported results and performing comparison among different methods. Efforts to address this issue have recently started to appear in the ME recognition community [11,13,21,22] and, following tendencies in the field, Competition on Recognition of Online Handwritten Mathematical Expressions (CROHME) has been recently established [23–25]. From the recognition rates reported in the CROHME editions, one can conclude that full recognition of MEs is still a challenging task. In order to advance development in this field, larger and diverse datasets for evaluating different aspects of recognition would be welcome. A comprehensive list of recognition evaluation related issues is given in [26], and importance of correct definition of ground-truth data and metrics for system evaluation are discussed in [16].

Often, for training and evaluating symbol recognition algorithms, users are required to enter several samples of each symbol, a tiring and boring task. From the user point of view, writing full expressions is more natural than writing several symbols individually and repeatedly. Thus, at an early stage of this research, while dealing with symbol recognition algorithms, we decided to collect samples of full expressions and then extract symbol samples from the expressions, rather than collecting samples of individual symbols. Moreover, it is likely that compared to the

* Corresponding author.
E-mail addresses: nina@ime.usp.br (N.S.T. Hirata), faguilar@ime.usp.br (F.D. Julca-Aguilar).

http://dx.doi.org/10.1016/j.patcog.2014.09.015
0031-3203/

symbols that are written individually, those obtained in this way will better resemble the symbols that actually occur in expression samples. A side effect of this approach is, however, the need to segment and label individual symbols in the sample expressions.

With respect to the problem of labeling individual symbols in handwritten MEs, some preliminary results obtained with an expression matching-based method were reported in [27]. The method assumes that the first expression is a model, with symbols previously segmented and labeled manually, and the second one is a transcription of the first, with symbols correctly segmented but not labeled. Then, the matching process associates each symbol in the transcribed expression to the corresponding symbol in the model expression, allowing automatic labeling of symbols in the transcribed expression. Segmentation is performed during writing, i.e., whenever the time gap between two strokes is larger than a user controlled threshold, they are separated. A bounding box enclosing the set of strokes considered as being part of one symbol is drawn dynamically, inducing users to undo the last written stroke whenever a non-desired stroke is joined to a previous symbol (see more details in [28]).

Given that current symbol classifiers perform very well [24], one could argue that labeling of individual symbols could be performed using one of those classifiers. However, one advantage of the matching approach is the fact that it also allows transferring of structural level ground-truth from one expression to the other. In contrast, only individual symbol class identification would not be sufficient for structural correspondence because a same symbol may occur repeatedly within an expression.

One of the contributions of this work relates to this observation. We propose a general framework for the creation of ground-truthed online ME datasets. The matching method is one of the main steps of this framework. The formulation of expression matching problem as a linear assignment problem presented here extends the method described in [27], including additional local features for the matching cost computation. The proposed framework, with discussions on how some other methods proposed in the literature for the creation of online ME datasets are related to it, and the details of the matching formulation are presented in Section 2.

At this point it is noteworthy to mention that our method does not make any specific assumptions related to context. It only assumes that the “objects” to be matched (in our case, mathematical expressions) are bidimensional structures composed by atomic units (in our case, individual symbols) and that they are at a similar scale and also spatially aligned. For instance, the symbol features for the definition of matching cost are expressed by means of shape dissimilarity (and not considering a specific symbol classification method). Hence, although the target objects in this work are mathematical expressions, the proposed framework and matching method could be adapted for chemical equations or even some types of 2D diagrams.

In Section 3, we detail the metrics used to evaluate matching performance and present a thorough evaluation of the proposed method on two large independent datasets, our own dataset (ExpressMatch dataset [29]) and the MfrDB dataset [22]. Matching results show that an overall mean symbol assignment rate superior to 99% is achieved. One aspect of special interest in this work is the evaluation of the influence of structural and symbol cost terms in matching performance. In Section 4 we examine some poor performing matching pairs and list some common types of symbol assignment errors together with a discussion on why they occur and how some of them could be fixed. Finally, in Section 5 we present the conclusions and point some future work.

2. Proposed matching approach

Since correct ME recognition requires not only correct symbol recognition but also correct understanding of the spatial arrangement among symbols, to improve overall ME recognition rate it is necessary to assess and understand where recognition is failing and how different techniques perform at different recognition levels. To that end, it is important to thoroughly experiment methods and techniques on large and statistically representative datasets. In order to automate experimental evaluation, datasets annotated with ground-truth data at different levels of ME structure are required.

Based on the idea of matching expressions discussed in the Introduction, we propose the following general procedure for generating samples of ground-truth annotated MEs:

1. creation of a corpus of model sample expressions, with correctly segmented symbols and ground-truth data attached to them;
2. capturing of samples (input expressions) of the models by having users transcribing them;
3. segmentation of input expression symbols;
4. matching of input expression symbols to the corresponding ones in the model; and
5. transferring of ground-truth data from the model to the input expression.

2.1. Model expression creation and symbol segmentation

In step 1, model expressions can be generated using grammars that describe MEs as in [13], having the advantage that ground-truth of each model expression is known. However, defining grammars is not a simple task and some expressions may, even being syntactically correct, correspond to expressions that semantically are unnatural. Another way to create model expressions is by hand selecting them. Hand selection of models presents advantages related to an easier control of the creation process, allowing a corpus of model expressions to be built in such a way as to be statistically representative of a given domain, with types of expressions, symbols, notations and respective frequencies specified to follow the distribution observed in that domain. In this case, however, there is a need to manually create them and attach ground-truth data.

In step 2, images rendered from the LaTeX representation of model expressions (either directly generated by a grammar or hand generated) can be shown for transcription. Alternatively, handwritten expression images can be generated just by handwriting the expressions or by rendering them by composing handwritten individual symbols, as in [11]. However, mimicking an actual handwritten expression by composing individually written symbols (possibly by distinct individuals) is not simple. If model expressions are handwritten, then there is a need to manually segment the symbols.

Segmentation of symbols can be performed by a specific segmentation algorithm, or based on approaches that perform ink data capture and symbol segmentation simultaneously. In either case, an interactive segmentation correction procedure may be very helpful.

In this work, we hand-select expression models and handwrite them to be shown for transcription. Segmentation, both in model expressions and in transcriptions, is performed during ink capture as detailed in [28]. After symbols in a transcribed expression are correctly segmented, expression matching is applied to establish the symbol-to-symbol correspondence.

Alternatively, both segmentation and matching could be carried simultaneously as in [13]. Subsets of strokes are evaluated with respect to its likelihood of being the strokes of one symbol, and relationships between several neighboring subsets of strokes are analyzed in order to match a subset of strokes to a terminal symbol in the model expression generated by a grammar. The