Reprint: 1997 Int. Conference on Acoustics, Speech, and Signal Processing (ICASSP-97), Munich, Germany, Vol. 4, pp. 3377-3380, Apr. 1997

ON-LINE SYMBOL SEGMENTATION AND RECOGNITION IN HANDWRITTEN MATHEMATICAL EXPRESSIONS

Hans-Jürgen Winkler and Manfred Lang
Institute for Human-Machine-Communication
Munich University of Technology, Arcisstr. 21, 80290 Munich, Germany
{win, lg}@mmk.e-technik.tu-muenchen.de

ABSTRACT

This paper is concerned with the symbol segmentation and recognition task in the context of on-line sampled handwritten mathematical expressions, the first processing stage of an overall system for understanding arithmetic formulas.
Within our system a statistical approach is used tolerating ambiguities within the decision stages and resolving them either automatically by additional knowledge acquired within the following processing stages or by interaction with the user. The recognition results obtained by different writers and expressions demonstrate the performance of our approach.

1. INTRODUCTION

We are accustomed in writing mathematical expressions containing integrals, fractions, exponents or indices by hand, but there is no user-adequate solution for entering these expressions into a computer. The most natural way is offered by analyzing the handwriting, but next to symbol segmentation and recognition structure analysis is required for extracting the meaning of the two-dimensional symbol positioning [1][2]. But this is just one difference to handwritten word recognition. Furthermore, the symbol segmentation and recognition task within our application is complicated by some additional circumstances:

- symbols are placed above, below, or even within other symbols.
- the writing size depends on the symbol as well as on its meaning within the expression (e.g. an upper case „X“ has not to be larger than a lower case „x“).

Hence, symbol segmentation and recognition systems presented up to now require an unequivocal stroke positioning for segmentation and an unequivocal style of symbol writing for recognition. Based on these restrictions, within the processing stages hard decisions are done (similar to systems used for analyzing printed expressions [1]) tolerating almost none of the inaccuracies caused by handwriting [3][4].

In comparison, our soft decision approach presented at ICASSP 95 tolerates ambiguities within the two processing stages [5], namely the symbol segmentation and the symbol recognition stage, both presented independent of each other at ICASSP 96 [6][7]. A brief review on the main points is given next. Within this review especially the determination of the probabilities is focused because the combination of former independent stages required some changes. In the following the final decision stage of the symbol segmentation and recognition system is presented. Within this processing stage the probabilities calculated in the preceding stages are used for determining the most probable symbol sequence based on the handwritten input. Additionally, knowledge obtained by a verification concerning the mathematical syntax is used. The recognition results presented at the final section of this proposal illustrates the performance of our approach and system.

2. APPROACH AND SYSTEM OVERVIEW

Based on the on-line sampled sequence of strokes $L=(L_1,\ldots,L_N)$ a symbol hypotheses net (SHN) is generated containing symbol hypotheses $G(k,g)=(L_k,\ldots,L_{k+g})$ of the handwritten input. Thus, soft-decision segmentation is done transforming the incoming stroke sequence L into one or more different sequences $\{G^{(i)}\}$ of symbol hypotheses represented by the corresponding path through the SHN.

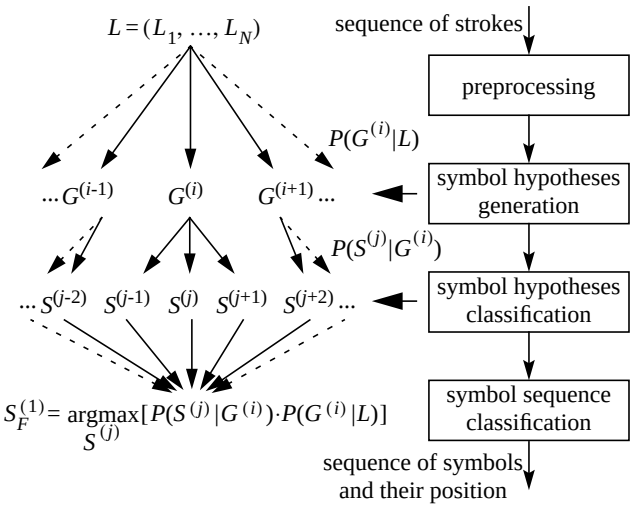

Figure 1: Statistical approach and system overview.

Reprint: 1997 Int. Conference on Acoustics, Speech, and Signal Processing (ICASSP-97), Munich, Germany, Vol. 4, pp. 3377-3380, Apr. 1997

Each symbol hypotheses is classified by a symbol recognition system based on Hidden Markov Models (HMMs) assigning different symbol recognition results $S(k, g, d)$ to each symbol hypotheses $G(k, g)$ of the SHN. Hence, this classification is a soft-decision process again, transforming each symbol hypotheses sequence $G^{(i)}$ into different symbol sequences $\{S^{(j)}\}$.

Each decision within the segmentation and recognition stage is done by a certain decision probability resulting to the sequence probabilities $P(G^{(i)}|L)$ and $P(S^{(j)}|G^{(i)})$. The final classification of the symbol sequence is based on these probabilities [5].

3. MAIN STAGES OF THE SYSTEM

3.1 Generating a symbol hypotheses net (SHN) [6]

The soft-decision segmentation by generating symbol hypotheses $G(k, g) = (L_k, \ldots, L_{k+g})$, $0 \le g \le 3$, such as illustrated by fig. 2, is based on:

- the unity measure $Z_C(k, g) \in \{0, 1\}$ determined by the complexities $(C_k, \ldots, C_{k+g})$ of the strokes. The complexity categorization $C_k \in \{C_P, C_S, C_C\}$ of each stroke L_k into one of the classes „Primitive“ (C_P), „Standard“ (C_S) or „Complex“ (C_C) is done by analyzing stroke-specific features.
- the unity measure $Z_G(k, g)$ of the strokes within $G(k, g)$, which is determined in several steps. First, by analyzing geometrical features between each stroke pairs L_k and L_{k+g}, the two by two unity measures $Z_P(k, g)$, $1 \le k \le N-g$, $1 \le g \le 3$, are calculated. For this calculation, knowledge obtained by a stroke pre-recognition stage is used additionally. Next, the two by two unity measures $Z_P(k, g)$ are combined in a certain manner resulting to the unity measure $Z_G(k, g)$ of the stroke sub-sequence $(L_k, \ldots, L_{k+g})$, $1 \le g \le 3$.

The overall unity measure $Z(k, g)$ of the strokes within the symbol hypotheses $G(k, g)$, $1 \le g \le 3$, is determined by the product of $Z_C(k, g)$ and $Z_G(k, g)$. By using two thresholds Z_0 and Z_1, $Z(k, g)$ is transformed to $\tilde{P}(G(k, g)|L)$ representing a probability measure that the stroke sub-sequence $(L_k, \ldots, L_{k+g})$ is a symbol of the handwritten input. The determination of the probability measure $\tilde{P}(G(k, 0)|L)$ (stroke L_k represents a symbol by itself) is done by analyzing the probability measures of all symbol hypotheses $G(k, g)$, $1 \le g \le 3$, concluding stroke L_k.

Each symbol hypotheses $G(k, g)$ with $\tilde{P}(G(k, g)|L) > 0$ is represented within the net (an example is given in fig. 4), the probability $\tilde{P}(G^{(i)}|L)$ of the path i through the SHN is defined by:

$$\tilde{P}(G^{(i)}|L) = \prod_{(G(k,g) \in \text{ path } i)} \tilde{P}(G(k, g)|L). \qquad (1)$$

Finally, normalization has to be done to $\sum_i \tilde{P}(G^{(i)}|L) = 1$ transforming $\tilde{P}(G(k, g)|L)$ to $P(G(k, g)|L)$ regarding

$$\sum_i P(G^{(i)}|L) = \sum_i \left(\prod_{(G(k,g) \in \text{ path } i)} P(G(k, g)|L) \right) = 1 \qquad (2)$$

as well as

$$\tilde{P}(G^{(i)}|L) / \tilde{P}(G^{(j)}|L) = P(G^{(i)}|L) / P(G^{(j)}|L). \qquad (3)$$

Thus, only the absolute values of the paths through the SHN are influenced but not their relative value among each other.

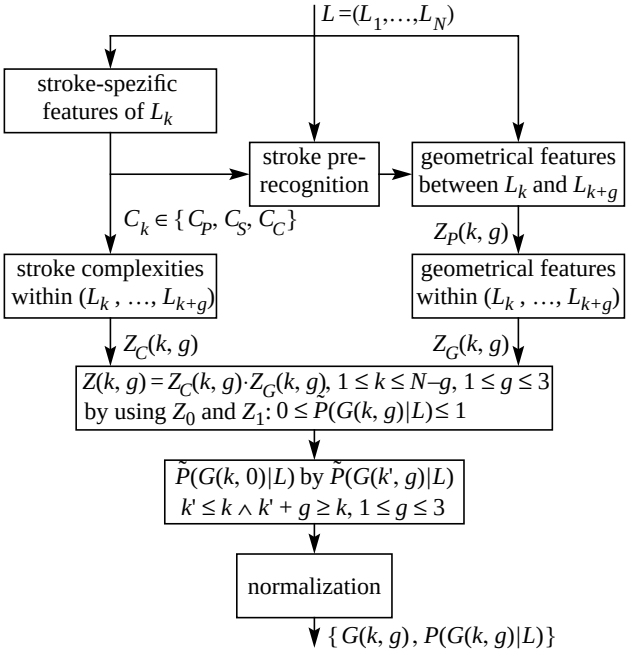

Figure 2: Detail view of the SHN generation stage.

3.2 Symbol hypotheses classification

Each element $G(k,g)$ of the SHN is regarded as a possible symbol of the handwritten input and therefore has to be applied to the symbol hypotheses classification system [5].

As illustrated in fig. 3, for each $G(k,g)$ preprocessing is done first, correcting the slant and slope of each symbol hypotheses as well as extracting parameters necessary for size and position normalization. Next, each symbol hypotheses is applied to a pre-recognition stage which is almost identical the stage used already for generating the SHN. However, this time pre-recognition is done for separating the symbols „Dot“, „Minus“, and „Fraction“ from the remaining symbols of the alphabet [6]. This separation is necessary because no „writing“ is done for a „Dot“ and the distinction between „Minus“ and „Fraction“, both represented by a horizontal line, requires contextual knowledge. Within the symbol hypotheses pre-recognition stage ambiguous recognition results between the symbols „Dot“ and „Minus“ as well as „Minus“ and „Fraction“ are tolerated. Symbol hypotheses rejected by the pre-recognition stage, i.e. symbol

Reprint: 1997 Int. Conference on Acoustics, Speech, and Signal Processing (ICASSP-97), Munich, Germany, Vol. 4, pp. 3377-3380, Apr. 1997

hypotheses not representing one of the above named symbols, are applied to a HMM-based classification system presented in [7].

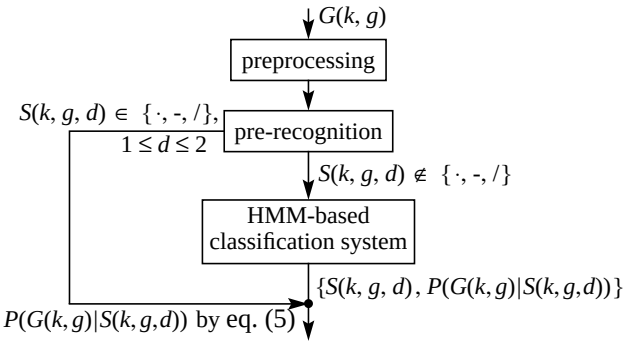

Figure 3: Overview of the symbol hypotheses classification stage, a detail view of the HMM-based classification system is given in [7].

Three different feature vector sequences $\{(x^{(F)})_{G(k,g)}\}$, $F \in \{O, V, H\}$ are extracted by the symbol hypotheses $G(k, g)$, one of them ($F = O$) generated by analyzing the temporal information during writing, the remaining two by the result (i.e. the image) of the writing. For each feature vector sequence consisting of $I^{(F)}$ feature vectors the generation probability $P((x^{(F)})_{G(k,g)}|\lambda^{(F)}_{S_a})$ is calculated using the Viterbi algorithm within the semicontinuous HMMs $\lambda^{(F)}_{S_a}$ representing the symbols S_a of the alphabet.

The symbol recognition result $S(k,g,1)$ as well as its alternatives $S(k,g,d)$, $d>1$, are determined by a weighted multiplication of the single generation probabilities, each of them additionally normalized to the number of feature vectors:

$$S(k,g,d) = \underset{\substack{S_a \notin \{S(k,g,1),\\ \ldots, S(k,g,d-1)\}}}{\operatorname{argmax}} \left[\prod_{F \in \{O,V,H\}} P((x^{(F)})_{G(k,g)}|\lambda^{(F)}_{S_a})^{w^{(F)}} \right], \quad (4)$$

$$w^{(O)} = 1/(2I^{(O)}),\ w^{(V)} = 1/(4I^{(V)}),\ w^{(H)} = 1/(4I^{(H)}).$$

The corresponding probabilities $P(G(k,g)|S(k,g,d))$ are determined by changing the *argmax*- to the *max*-operator.

For homogeneity, a generation probability analogous to the HMM-based symbol classification is assigned to the pre-recognized symbols by:

$$P(G(k,g)|S(k,g,1))|_{S(k,g,1) \in \{\cdot, -, /\}} = \underset{\forall S(k,g,1) \notin \{\cdot, -, /\}}{\operatorname{med}} [P(G(k,g)|S(k,g,1))]. \quad (5)$$

In the case of an ambiguous pre-recognition result, the generation probability of the corresponding recognition alternative $S(k,g,2)$, $S(k,g,2) \in \{\cdot, -, /\}$, is determined by using a very small but positive delta probability ΔP:

$$P(G(k,g)|S(k,g,2))|_{S(k,g,2) \in \{\cdot, -, /\},\ S(k,g,2) \neq S(k,g,1)} = P(G(k,g)|S(k,g,1))|_{S(k,g,1) \in \{\cdot, -, /\}} - \Delta P,\ \ \Delta P \to +0. \quad (6)$$

3.3 Symbol sequence classification

The information available at this final decision stage consists of the probabilities $P(G(k, g)|L)$ obtained by generating the symbol hypotheses $G(k,g)$ and, such as illustrated in fig. 4, the probabilities $P(G(k,g)|S(k,g,d))$ obtained by their recognition.

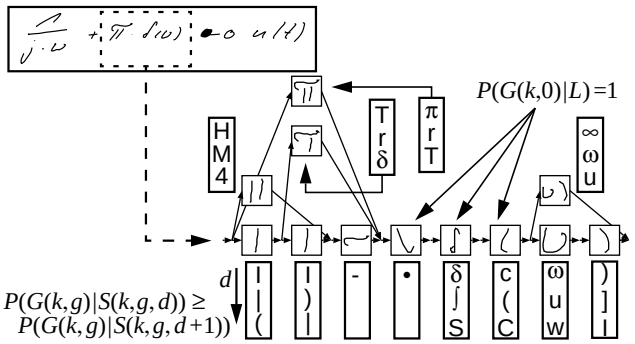

Figure 4: Information available for the final symbol sequence classification: an example containing the top-3 symbol recognition results to each $G(k,g)$.

Using the elements of the generated SHN, the decision criterion given in fig. 1 can be transformed to:

$$S^{(1)}_F = \underset{S^{(j)}}{\operatorname{argmax}} \left[\prod_{(S(k,g,d) \in \text{ path } j)} P(S(k,g,d)|G(k,g)) \cdot P(G(k,g)|L) \right]. \quad (7)$$

Using the Bayes theorem and assuming, that all „a-priori"-probabilities are constants, eq. (7) results in:

$$S^{(1)}_F = \underset{S^{(j)}}{\operatorname{argmax}} \left[\left(\frac{P_S}{P_G}\right)^{N^{(j)}} \cdot \prod_{(S(k,g,d) \in \text{ path } j)} P(G(k,g)|S(k,g,d)) \cdot P(G(k,g)|L) \right], \quad (8)$$

$$P_G = P(G(k,g)),\ P_S = P(S(k,g,d)),\ N^{(j)} = N(S(k,g,d) \in \text{path } j).$$

Focusing the left term within this decision criterion, either long or short paths (small or large number $N^{(j)}$ of symbols $S(k,g,d)$ within the path j) through the SHN will be preferred depending on the relation of the „a-priories" P_G and P_S. Within the right part, always short paths will be preferred caused by the normalization done during the symbol hypotheses generation and classification. Hence, normalization is done to the number of elements $N^{(j)}$ within the path j „neglecting" the „a-priories" and resulting to the final decision criterion:

$$\tilde{S}^{(m)}_F = \underset{\substack{S^{(j)} \notin \{\tilde{S}^{(1)}_F,\\ \ldots, \tilde{S}^{(m-1)}_F\}}}{\operatorname{argmax}} \left[\sqrt[N^{(j)}]{\prod_{S(k,g,d) \in \text{ path } j} P(S(k,g,d)|G(k,g)) \cdot P(G(k,g)|L)} \right]. \quad (9)$$

Finally, a verification concerning the mathematical syntax based on the number of parentheses, brackets, braces (a left one requires a right one of the same kind and reverse) and power symbols (always by pairs) is carried out. Recognition results failing this verification are considered as invalid.

Reprint: 1997 Int. Conference on Acoustics, Speech, and Signal Processing (ICASSP-97), Munich, Germany, Vol. 4, pp. 3377-3380, Apr. 1997

4. RECOGNITION RESULTS AND DISCUSSION

For the recognition experiment nine writer contributed five versions of 17 different expressions using each symbol of the given alphabet (currently 84 symbols). The number of symbols within the expressions are ranging from at least 13 up to 45 symbols, on average an expressions consists of 27 symbols. In comparison to this number, word recognition means recognizing 5 characters on average [8].

Within our recognition experiment the 10 most probable symbol sequences $\tilde{S}_F^{(m)}$, $m \le 10$, are generated by the decision criterion given in eq. (9) without using any language model (a very powerful knowledge source for the recognition of handwritten words or speech) or any symbol distribution knowledge but using the knowledge obtained by the quite simple syntax verification.

	Recognition rate of the expressions by $\tilde{S}_F^{(m)}$,				
	$m = 1$	$m \le 2$	$m \le 3$	$m \le 4$	$m \le 10$
Average:	44%	56%	60%	63%	72%
Writer-dep. range:	28% - 68%	39% - 79%	40% - 84%	45% - 84%	54% - 87%
Expr.-dep. range:	2% - 91%	4% - 96%	11% - 96%	11% - 96%	16% - 100%

Table 1: Average recognition results and their ranges depending on the writer and on the expression.

On average, 44% of the expressions are recognized error-free (i.e. each symbol within the expression) by $\tilde{S}_F^{(1)}$, depending on the writer it ranges from 28% to 68%. The average recognition rate is raising up to 72% within the $m \le 10$ most probable recognition alternatives, but the user has to do the selection.

Some well (completely error-free) and poor (at least one error within each $\tilde{S}_F^{(m)}$, $m \le 10$) recognized expressions are given in fig. 5. Within the poor recognized expressions the error position(s) and their kind are marked based on $\tilde{S}_F^{(1)}$.

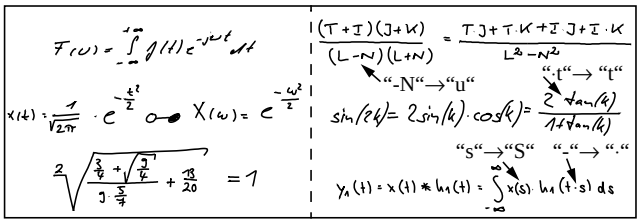

Figure 5: Some examples from the expression data base.

Calculating a symbol-normalized recognition rate analogous to [1] for achieving independency of the complexity of the expressions, the average recognition rate results in more than 95% ranging between 93% and 98% for the different writers. If an error occurs, in 80% of all cases the error is only based on a wrong symbol recognizer result (the symbol hypotheses $G(k, g)$ underlaying the symbols recognizer results $S(k,g,1)$ within the sequence $\tilde{S}_F^{(1)}$ coincide with the symbols of the handwritten input). Furthermore, the recorded symbol recognition errors are mainly caused by a mix-up of upper and lower case letters having the same shape [7].

5. CONCLUSIONS

Though the average symbol segmentation and recognition rate obtained by our system is more than 95%, „only" 44% of the mathematical expressions of our test data set are recognized completely error-free. This problem is based on the quite large number of symbols within an expression as well as on the missing of a language model usable for mathematical expressions. By displaying the 10 most probable recognition results to the user for selection, the expression recognition rate raises to 72% by using this kind of interaction. Thus, next to the combination with the structure analysis system presented in [2], another task is the implementation of a pen-based user interface enabling the user to make corrections exactly at the error position(s) either by choosing recognition and/or segmentation alternatives or by re-writing symbols.

6. REFERENCES

[1] H.-J. Lee, M.-C. Lee: *Understanding Mathematical Expressions using Procedure-Oriented Transformations*, Pattern Recognition 27(3), pp. 447-457, 1994.

[2] H.-J. Winkler, H. Fahrner, M. Lang: *A Soft-Decision Approach for Strucural Analysis of Handwritten Mathematical Expressions*, ICASSP 95, Detroit, pp. 2459-2462, May 1995.

[3] A. Belaid, J.-P. Haton: *A Syntactic Approach for Handwritten Mathematical Formula Recognition*, IEEE PAMI 6(1), pp. 105-111, 1984.

[4] Y.A. Dimitriadis, J.L. Coronado: *Towards an ART based Mathematical Editor that uses On-line Handwritten Symbol Recognition*, Pattern Recognition 28(6), pp. 807-822, 1995.

[5] M. Koschinski, H.-J. Winkler, M. Lang: *Segmentation and Recognition of Symbols within Handwritten Mathematical Expressions*, ICASSP 95, Detroit, pp. 2439-2442, May 1995.

[6] S. Lehmberg, H.-J. Winkler, M. Lang: *A Soft-Decision Approach for Symbol Segmentation within Handwritten Mathematical Expressions*, ICASSP 96, Atlanta, pp. 3434-3437, May 1996.

[7] H.-J. Winkler: *HMM-based Handwritten Symbol Recognition using On-line and Off-line Features*, ICASSP 96, Atlanta, pp. 3438-3441, May 1996.

[8] C.C. Tappert, C.Y. Suen, T. Wakahara: *The State of the Art in On-line Handwriting Recognition*, IEEE PAMI 12(8), pp. 787-808, 1990.