

# REAL-TIME STRUCTURAL ANALYSIS OF HANDWRITTEN MATHEMATICAL FORMULAS BY PROBABILISTIC CONTEXT-FREE GEOMETRY MODELLING

Jörg Hunsinger, Robert Lieb, Manfred Lang

Institute for Human-Machine Communication  
Technical University of Munich, D-80290 Munich, Germany  
{huj, lie, lg}@mmk.e-technik.tu-muenchen.de

## ABSTRACT

*Robust structural analysis techniques are crucial for the recognition of handwritten mathematical formulas. Therefore we integrated a novel probabilistic structural classification module into a single-stage semantic decoder, recursively taking relative symbol (or symbol group) positions and sizes into account. On the basis of Gaussian, statistically independent distributions for the resulting features we obtained a reliable structural assessment measure which is set against the corresponding symbol classification scores in a top-down chart parsing scheme.*

*The final recognition system performs at a writer specific accuracy of 95.0 %. Real-time inline processing is enabled due to an incremental breadth-first search strategy.*

## 1. INTRODUCTION

Online recognition of handwritten mathematical formulas basically requires three different problems to be solved [1]: 1) segmentation of handwriting strokes to symbol hypotheses, 2) handwritten symbol recognition, and 3) structural analysis of 2D symbol arrangements.

Classically, these tasks are performed step by step in a bottom-up multiple stage architecture, which may imply the following typical shortcomings:

- In the absence of higher level contextual information, symbol independent segmentation rules often fail.
- For similar reasons, symbol recognition errors occur due to missing higher level constraints or due to preceding missegmentations.
- In case of segmentation and/or symbol recognition errors, information is irretrievably lost at the borderline to any follow-up structural analysis stage.

Our single-stage top-down classification strategy significantly reduces such obstacles by

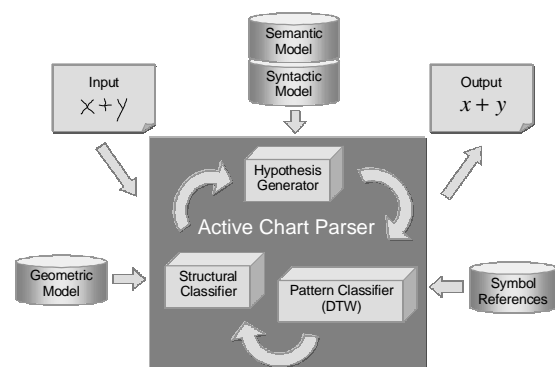
- allowing for semantically consistent (sub-)hypotheses only,

- incorporating semantic context into the syntactic symbol and placement analysis processes, and
- driving segmentation as an implicit aspect of symbol recognition under contextual constraints.

Since we aim at an integrated system architecture based on a *Multimodal Probabilistic Grammar* [2], a statistical measure for the structural contents of handwritten input was the next important system component to be implemented in this approach. This paper describes the detailed properties of this structural classification module as well as its contribution to the overall MAP classifier inside our probabilistic chart parser.

## 2. OVERVIEW

The basic system components and their interactions are displayed in Fig. 1. Essentially, incremental first-last processing of the given handwritten input is performed by cycling through the different chart parser modules; hereby one cycle corresponds to one additionally processed handwriting stroke. Regarding the general concept, especially our syntactic-semantic representations and their integration in a chart parsing mechanism, we refer to [3]. As can be seen from Fig. 1, an alternating evaluation of syntactic-semantic, graphemic, and geometric knowledge takes place.



**Figure 1:** Schematic system overview. For clarity, only components addressed in this paper are shown.

### 3. METHODS

This section presents the different parameters and procedures that make up our structural analysis component. After introducing our feature extraction methods, we specify how the corresponding probabilistic knowledge is gained and finally evaluated during recognition.

#### 3.1. Geometric Feature Extraction

In order to obtain a reliable probabilistic geometric model covering all the supported mathematical operators [2], the first step is to find a proper set of normalized geometric features that characterize the positional relations between all the occurring handwritten formula constituents.

**3.1.1. Starting Point.** Our top-down chart parser systematically scans the search space by unfolding *Semantic Structures*, i.e. hierarchic combinations of *semuns* (*semantic units*) [3], as compact representations of mathematical contents on the semantic level. Every semun with its corresponding type, value, and successor attributes refers to a certain mathematical operator or operand [2]. To compare a given semantic hypothesis with the acquired handwritten symbol sequence, the semantic hierarchy is mapped to a *Syntactic Network* consisting of interconnected *Syntactic Modules* (SM) in which writing order or symbol choice are modeled by transition or emission processes, respectively.

Accordingly, it makes sense to look for geometric features that can be calculated per semun so that 1) the geometric contribution to our probabilistic classification measure is smoothly integrated in the existing parsing mechanism, and 2) a robust context-free parameterization results.

**3.1.2. Geometric Elements.** In compliance with our SM definition [3] we consider positional relations between pairs of so-called *geometric elements* belonging to a particular semun. Such an element may either be one of the emitted handwritten symbols or the entire handwritten subexpression corresponding to one of the semun specific semantic successors. For example, the semun type SUM (simple addition) with its single symbol emission (“+”-sign) and its two successors (left and right addends) requires three element pairs to be processed.

For every pair of geometric elements 1) relative symbol positions and 2) relative symbol sizes are extracted. Recursive context-free geometry processing is achieved by merging relevant features derived from inside a specific semun (*element fusion*); the resulting medial feature vector is then passed back to the semantic predecessor (cf. below).

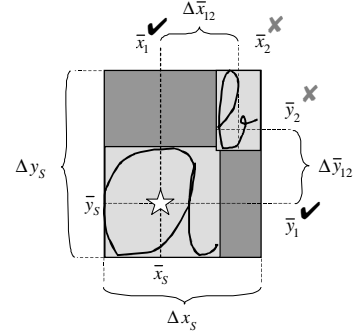
**3.1.3. Symbol Positions.** To take local position-

al offsets between syntactically correlated geometric elements into account, we first calculate symbol specific centers  $(\bar{x}_i, \bar{y}_i)$  of their surrounding rectangles  $(x_{i1}, x_{i2}, y_{i1}, y_{i2})$ :

$$\bar{x}_i = x_{i1} + \lambda_x(x_{i2} - x_{i1}); \quad \bar{y}_i = y_{i1} + \lambda_y(y_{i2} - y_{i1}) \quad (1)$$

Herein,  $\lambda_x, \lambda_y \in [0, 1]$  are weighting factors based on expert knowledge. The final features  $\Delta\xi$  and  $\Delta\eta$  for horizontal and vertical offset components result by rescaling the pairwise center distances to the current semun’s overall dimensions as sketched in Fig. 2. This implies a range of feature values from -1 to 1.

$$\Delta\xi_{ij} = \frac{\bar{x}_i - \bar{x}_j}{x_{S2} - x_{S1}} = \frac{\Delta\bar{x}_{ij}}{\Delta x_S}; \quad \Delta\eta_{ij} = \frac{\bar{y}_i - \bar{y}_j}{y_{S2} - y_{S1}} = \frac{\Delta\bar{y}_{ij}}{\Delta y_S} \quad (2)$$



**Figure 2:** Geometric feature calculation (positional components) for the semun type “POW”. The ☆ sign denotes the type specific overall semun center (Eq. 3) which is returned for recursive processing. ✓ (✗) components are regarded (disregarded) during element fusion.

Finally, the overall semun center is determined as follows (element fusion):

$$\bar{x}_S = \sum_{i=1}^n \varepsilon_{x,i} \bar{x}_i / \sum_{i=1}^n \varepsilon_{x,i}; \quad \bar{y}_S = \sum_{i=1}^n \varepsilon_{y,i} \bar{y}_i / \sum_{i=1}^n \varepsilon_{y,i} \quad (3)$$

The binary weights  $\varepsilon_x, \varepsilon_y$  include type specific expert knowledge in order to regard or disregard contributions from the different embedded geometric elements ( $1 \leq i \leq n$ ). For example, we suppress the positional contribution of the superscript term when merging the geometric elements of an exponential expression (Fig. 2). As soon as the semantic predecessor is processed as described above,  $\bar{x}_S$  and  $\bar{y}_S$  represent the center coordinates of the corresponding successor-type geometric element.

**3.1.4. Symbol Sizes.** Although a considerable portion of structural information is covered by symbol placement properties, a reliable and detailed structural analysis must also take relative symbol si-

zes into account. Especially nested expressions, e.g.  $x_{2y}$ , are usually written down by combining positional and scaling techniques. Again, we evaluate pairs of geometric elements per semun, this time by computing the ratio of the corresponding symbol sizes  $g$ :

$$\zeta_{ij} = \log_5 \frac{g_i}{g_j}; \quad g_i = \lambda_g (y_{i2} - y_{i1}) \quad (4)$$

As usual, expert knowledge serves to specify symbol dependent weights  $\lambda_g \in [0,1]$  which conform to common typing conventions. The special value  $\lambda_g = 0$  is used to exclude certain critical symbols (e.g. minus signs) completely from scaling considerations. Since the scaling range in typical handwritten formulas is below 5, these ratios are transformed to a logarithmic scale of base 5; in this way the interesting range of symbol sizes is mapped to a feature value range matching that of our positional features (cf. Sec. 3.2, Fig. 3). This convention makes sure that positional and scaling properties have a counterbalanced impact on the overall structural classification measure (cf. Sec. 3.3).

The medial symbol size inside the current semun is given by:

$$g_s = \sum_{i=1}^n \varepsilon_{g,i} g_i \Big/ \sum_{i=1}^n \varepsilon_{g,i} \quad (5)$$

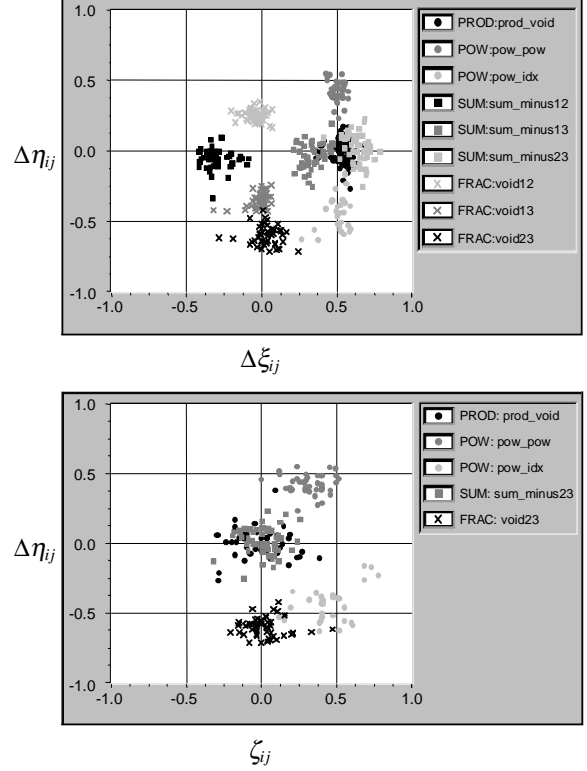
In analogy to Eq. 3, binary weights  $\varepsilon_g$  enable selective exclusion of irrelevant geometric elements from this value. It is then fed into the scaling feature computation for the predecessor semun as described in Sec. 3.1.3.

### 3.2. Parameterization

For every given geometric element pair (i,j) the above features yield 3-dimensional feature vectors  $\Gamma_{ij} = \langle \Delta\xi_{ij}, \Delta\eta_{ij}, \zeta_{ij} \rangle$ . The pairwise feature distributions as observed in a writer specific training corpus (cf. Sec. 3.4) are displayed in Fig. 3 for some typically competing mathematical functions. On the whole, the structural classes to be discriminated appear to be rather well-separated. Furthermore, the plots indicate that single Gaussians should be sufficient for modelling the different components.

To parameterize our structural classifier, we decided to assume statistical independence between the three feature vector components for the following reasons:

- The feature distributions do not significantly reveal statistical coupling effects,
- including covariances in our parameter sets would require inappropriately large training corpora per writer, and



**Figure 3:** Pairwise geometric feature distributions for selected semun types: product (PROD), sub-/superscript (POW), subtraction (SUM), and fraction (FRAC). The two-digit numbers (12, 13, 23) refer to the different pairs of geometric elements.

- even by using our inline processing scheme, a statistically independent implementation would not reach real-time performance.

Thus a number of at most  $3(n^2-n)$  model parameters, i.e. 3 arithmetic means plus 3 standard deviations per geometric element pair, must be estimated for every supported semantic type ( $n$  geometric elements).

### 3.3. Classification Measure

Since our chart parser incrementally rates hypotheses by adding up neglog scores derived from probabilistic measures, it is straightforward to define our structural classification score as the exponent of the respective Gaussian PDFs. The  $\Delta\xi$  contribution (and analogously those for  $\Delta\eta$  and  $\zeta$ ) for a pair (i,j) of geometric elements then reads:

$$B_{ij}^{(\Delta\xi)} = \left( \frac{\Delta\xi_{ij} - \mu_{ij}^{(\Delta\xi)}}{\sigma_{ij}^{(\Delta\xi)}} \right)^2 \quad (7)$$

$\mu_{ij}$  and  $\sigma_{ij}$  denote the Gaussian mean or standard deviation, respectively, for element pair (i,j). The total structural score for a semun  $S$  results by sum-

$$\frac{x+2}{x^6+x^4-x^2+1} = \frac{\frac{3}{8}}{x-1} + \frac{-\frac{1}{8}}{x+1} + \frac{-\frac{1}{4}x-\frac{1}{4}}{x^2+1} + \frac{-\frac{1}{2}x-1}{(x^2+1)^2}$$

$$\int \frac{1}{x\sqrt{a^2x^2+c^2}} dx = -\frac{1}{c} \ln \left| \frac{c+\sqrt{a^2x^2+c^2}}{x} \right|$$

$$\sum_{k=0}^n \binom{n}{k} = (1+1)^n = 2^n$$

$$\sin \alpha - \sin \beta = 2 \cos \frac{\alpha+\beta}{2} \sin \frac{\alpha-\beta}{2}$$

$$e = \lim_{x \rightarrow \infty} \left(1 + \frac{1}{x}\right)^x$$

$$\int_0^1 \frac{\ln x}{\sqrt{1-x^2}} dx = -\frac{\pi}{2} \ln 2$$

$$(x^x)^x = x^{x^x} = x^{x^x}$$

$$u(x, y) = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{y f(t)}{y^2 + (x-t)^2} dt$$

**Figure 4:** Test formula samples and results. The dashed circles point out deficiently recognized areas.

ming over all features and geometric element pairs as follows:

$$B_S^{(\Gamma)} = (2n_{\Delta \xi, \Delta \eta} + n_{\zeta})^{-\alpha_{\Gamma}} \sum_{\forall ij \in S} (B_{ij}^{(\Delta \xi)} + B_{ij}^{(\Delta \eta)} + B_{ij}^{(\zeta)}) \quad (8)$$

By exponentiating the number of actually included element pair scores (bracketed coefficient term) with a so-called *structural complexity weight*<sup>1</sup>  $\alpha_{\Gamma}$ , the structural contribution to the overall hypothesis score can be adjusted to that from the symbol recognition level: On the basis of around 2000 symbol and structure scores from our test formula corpus we derived an optimum value of  $\alpha_{\Gamma} = 1.25$  for this empirical constant; the majority of both types of scores is then located inside the common value interval [0,100].

At the same time – since  $\alpha_{\Gamma} \approx 1$  – the structural scores from different semun types are well-balanced against each other by being rescaled according to the number of scored geometric element pairs.

### 3.4. Training

For a writer specific evaluation we trained our structural classification layer using merely a corpus of 16 realistic handwritten formulas taken from [4]. In parallel we extracted a reference base of symbol patterns for our extended DTW symbol recognition layer [3]; for a better coverage of the supported symbol inventory we added another record for any favoured writing style of every symbol from our alphabet.

<sup>1</sup> Increasing this parameter upvalues structurally more complex hypotheses (corresponding to a higher number of geometric element pairs).

## 4. RESULTS & CONCLUSIONS

The recognition results in terms of 8 independent test formulas are shown in Fig. 4. Defining the recognition accuracy  $R$  as the overall quota of correctly classified symbols plus structural constituents, we obtain a value of  $R = 95.0\%$ .

Summing up, we have shown that a single-stage probabilistic approach including geometric-structural properties is capable of robustly decoding handwritten mathematical formulas. Our compact writer specific prototype runs under real-time conditions while utilizing a strokewise inline processing scheme. Although it seems that even small training corpora yield suitable structural model parameters, it should be examined to which extent the system benefits from larger data pools. Moreover, we aim at implementing a writer independent version as well as a refined alignment between the different involved classification layers.

## 5. REFERENCES

- [1] Blostein D. and Grbavec A., *Recognition of Mathematical Notation*, in Wang P.S.P. and Bunke H., editors, *Handbook on Optical Character Recognition and Document Image Analysis*, chapter 21, pp. 557-582, World Scientific Publishing, 1997.
- [2] Hunsinger J. and Lang M., *A Speech Understanding Module for a Multimodal Mathematical Formula Editor*, Proc. ICASSP 2000, Istanbul, Turkey, Vol. IV, pp. 2413-2416, June 2000.
- [3] Hunsinger J. and Lang M., *A Single-Stage Top-Down Probabilistic Approach towards Understanding Spoken and Handwritten Mathematical Formulas*, Proc. ICSLP 2000, Beijing, China, Vol. IV, pp. 386-389, October 2000.
- [4] Råde L., Westergren B., and Vachenaue P., *Springers mathematische Formeln*, second edition, Springer-Verlag Berlin/Heidelberg, 1997.