Handwritten Mathematical Symbol Classification Using Layout Context

Ling Ouyang and Richard Zanibbi
Document and Pattern Recognition Lab
Rochester Institute of Technology, NY, USA
Email: lxo7459@rit.edu, rlaz@cs.rit.edu

A new descriptor, *layout context*, is introduced to describe the distribution of the surrounding symbols' positions relative to a target mathematical symbol within a mathematical expression. Using this new descriptor, a coarse classification is performed in which all mathematical symbols are classified into eight classes[3] as shown in Figure 1. These classes represent different centroid locations and surrounding regions associated with a symbol.

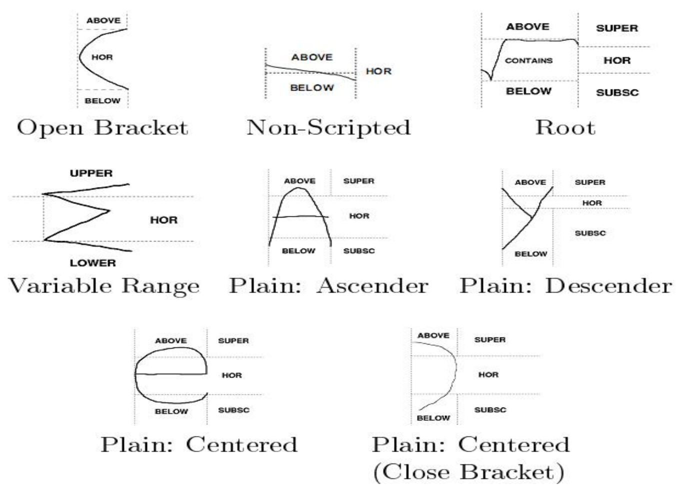


Figure 1: Eight layout classes [3]

The layout context is an extension of the concept of shape context, which was proposed in Belongie, Malik, and Puzicha[2]'s paper. Shape context is the distribution of the relative position of all the other points relative to the target point in an individual symbol's contour and proved very effective in handwritten digits classification. Extending this concept from an individual point on a character contour to a symbol in an mathematical expression, the distribution of relative positions of the bounding box centroid point of the other symbols relative to that of the target symbol is defined as the layout context. The histograms of the distribution of these relative coordinates are measured in a log-polar space which can make these descriptors more sensitive to positions of nearby symbols than to those symbols farther away. Here we use 5 bins for logarithm radial normalization distance distribution and 12 bins for the angle distribution to represent the histogram of relative coordinates to a reference symbol. The radial distance are normalized to achieve scale invariance of the symbol recognition. An example mathematical expression is presented in Figure 2(a). The vectors in Figure 2(b) express the positions of all the other symbols relative to the reference symbol "a". Figure 3 shows the layout context of some reference symbols in this expression.

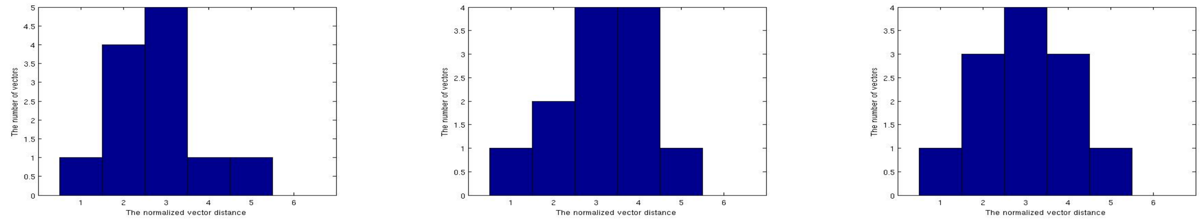
The reason to use layout context is to decrease the variability in feature values for symbols in the same type during the classification as this layout based descriptor pays less attention to the difference between the shape of individual symbols. For example, symbol "2" and symbol "b" in the Figure 2(a) expression would be classified as ascender in this coarse classification other than numeral and alphabet in a fine classification. Furthermore, the classification result could be used as a first step for some hierarchical classifiers. For example, we could first use the matching cost of the layout context together with some local visual features(aspect ratio difference and tangent angle dissimilarity between two symbols) as features during the first coarse classification and then the feature of shape context in the following fine classification for every symbol that in a layout class in Figure 1.

There are four main steps in our experiment of this coarse classification. At the first step, ground truth data of the bounding box and centroid of each simple is generated. Then the layout context of these symbols are calculated. After that, a matching process is performed between layout contexts of every two symbols within an expression and the final matching result that minimizes the total cost is selected. At last the layout context matching cost together

(a) Original mathematical expression

(b) Vectors from sample symbol "a" to all the other symbols

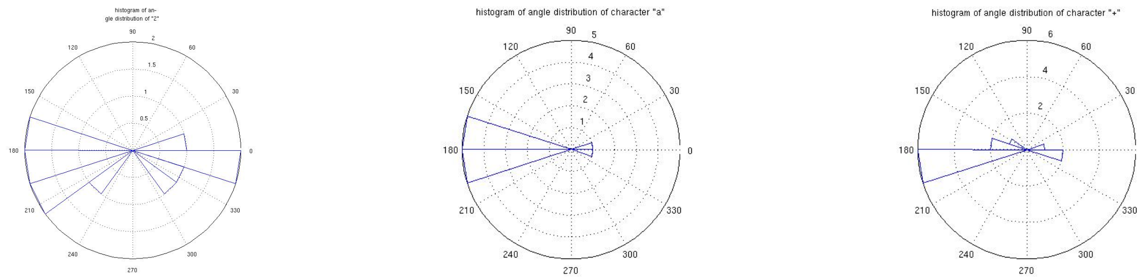
Figure 2: Original mathematical expression and vectors of the layout context



(a) Histogram of log radial distance distribution of "2"

(b) Histogram of log radial distance distribution of "a"

(c) Histogram of log radial distance distribution of "+"



(d) Histogram of angle distribution of "2"

(e) Histogram of angle distribution of "a"

(f) Histogram of angle distribution of "+"

Figure 3: Histogram of normalized radial distance and angle distribution

with some other visual appearance features such as the tangent angle dissimilarity between the symbols and aspect ratio of the bounding box are used to as the feature vector in the later classification.

Consider a symbol p_i and another symbol q_i in an expression and let C_{ij} represents the cost of matching the layout context $h(p_i)$ and $h(q_i)$ of these two symbols. As layout contexts are histogram of the distribution, it is nature to use the χ^2 test statistic[2] to represent the matching cost. This is because χ^2 test is commonly used to test whether or not an observed frequency distribution of the relative coordinates for the test symbol differs from a reference distribution, which is calculated from mathematical symbols from a certain layout class in Figure 1. In addition to the layout context matching cost, another cost based on the appearance is included, such as a measure of tangent angle dissimilarity, which is defined[1]as the half the length of the chord in the unit circle between two unit vectors that having angles θ_i and θ_j relative to the centroid of the target symbol, denoted as C_A. The combined matching cost is then computed as a weighted sum. Then the layout distance is denoted as this weighted sum cost and used as the measure distance of a K-NN classifier. The classification process will be individually performed for every symbol; A large number of training samples (histograms associated with symbols in training data) from different layout class are generated for the test symbol. The k-nearest-neighbor classifier will then find the k closest symbols in the histogram space, and then select the class that the majority of the k samples belong. After classification, a confusion matrix would be established to evaluate how the errors are distributed across the eight classes.

References

[1] S.Belonie and J.Malik. Matching with Shape Contexts. *IEEE Workshop on Contentbased Access of Image and Video Libraries*, 2000.

[2] J.Malik S.Belonie and J.Puzicha. Shape Matching and Object Recognition Using Shape Context. *IEEE Transactions On Pattern Analysis And Machine Intelligence*, 24(24):509–522, 2003.

[3] R. Zanibbi, D. Blostein, and J.R. Cordy. Recognizing Mathematical Expressions Using Tree Transformation. *IEEE Transaction On Pattern Aanlysis And Machine Intelligence*, pages 1455–1467, 2002.