The Goal

User → Recognition System → Display
Pen Data → Semantic Interpretation
Typesetting Commands

The Recognition System

Pen Data → Glyph Classification → Stroke Set Partitioning
Combination Weightings

Glymph Parsing → Glyphs
Typesetting Commands

z(x, y) = \sin x \sqrt{1 - \sin^2 y}

Stroke Set Partitioning

Purpose
Since each stroke can only belong to a single glyph, it is necessary to find the best disjoint partitioning of the strokes into glyphs.

Additive Cost and Combination Weighting
Partitions are ranked based on the sum of the best possible glyph assignment for each set of strokes. Multiple stroke glyphs have an additional penalty in order to equalize the lower cost that comes from combining strokes. The additive nature of this cost function makes the problem amenable to dynamic programming techniques.

Order of Growth
Even with dynamic programming, however, the search space of this problem is exponential in size. Constraints need to be placed on the combinations explored search if it is not to take exponential time. One simple constraint is to limit the maximum number of stroke a glyph can be composed of to a reasonable number. Another might be to only consider strokes which occurred sequentially in time.

Minimum Spanning Tree Constraint
A better constraint is to consider only stroke combinations which form connected subtrees of a minimum spanning tree where each stroke represents a vertex and each edge is weighted according to some distance metric. One possible metric assigned the distance between two strokes to be the Euclidean distance between the centroids of their bounding boxes.

Glyph Classification

Purpose
Determining which class of glyph is best represented by a particular set of strokes.

Stroke Direction Invariance
Using the first and last points of the stroke, attempt to flip strokes so that they have a canonical direction, allowing the user to write simple strokes in either direction. Strokes determined to be closed or near closed are not flipped.

Stroke Order Invariance
When creating multiple strokes glyphs, the strokes are ordered based on the angle between their upper edge and last point to allow the user to enter strokes in any order.

Preprocessing
The points in a set of strokes are shifted, scaled, and then linearly resampled along the stroke arc lengths.

Principal Component Analysis
The glyph vector is then projected into the space spanned by the principle components of all the glyphs in the training set in order to reduce noise and increase recognition accuracy.

Gaussian Density Estimation
A potential glyph is then assigned a cost for each glyph type based on a Gaussian density estimated from a set of examples glyphs from that class.

Glyph Parsing

Purpose
Assign characters to glyphs based on inter-glyph geometry

Geometric Grammar Types
Each character instantiates a single geometric grammar type, according to its allowable relationships with other characters. Geometric grammar types include:

- Simple Character
- Delimiters
- Fraction Bar
- Square Root
- Grouping Operators
- Accent
- Crossing Strokes

Box Relationships
Using simple relations between their bounding boxes, glyphs are assigned the best characters consistent with their geometric relationships. For example, if an up/down relationship is found, some possible interpretations include an accent over a simple character or a simple character over a fraction bar.