

REAL TIME GESTURE RECOGNITION WITH CURVE DETECTION

BACKGROUND OF THE INVENTION

51. Field of the Invention

The present invention relates to the enhancements and curve detection methods in the algorithm of the realization of real-time gesture recognition which uses a unique recognition method to match symbols and lines with database in
10 real-time.

2. Description of the Related Art

A primary goal of gesture recognition research is to create a system which
15 can identify specific human gestures and use them to convey information or to control device.

Speech and handwriting recognition systems are similar to gesture recognition systems, because all of these systems perform recognition of something that moves, leaving a "trajectory" in space and time. By exploring the literature of
20 speech and handwriting recognition, classification and identification schemes can be studied, which might aid in developing a gesture recognition system.

From some points of view, handwriting can be considered a type of gesture. On-line (also called "real time" or "dynamic") recognition machines identify handwriting as a user writes. On-line devices have the advantage of capturing the
25 dynamic information of writing, including the number of strokes, the ordering of strokes, and the direction and velocity profile of each stroke. On-line recognition systems are also interactive, allowing users to correct recognition errors, adapt to the system, or see the immediate results of an editing command.

Most on-line touch panels capture writing as a sequence of coordinate points. Recognition is complicated in part, because there are many different ways of generating the same character. For example, the letter E's four lines can be drawn in any order.

5 Handwriting touch panels must take into account character blending and merging. Also, different characters can look quite similar. To tackle these problems, handwriting touch panels pre-process the characters, and then perform some type of shape recognition. Preprocessing typically involves properly spacing the characters and filtering out noise from the touch panel. The more complicated processing
10 occurs during character recognition.

Another scheme draws its classification method from signal processing, in which curves from unknown forms are matched against prototype characters. They are matched as functions of time or as Fourier coefficients. To reduce errors, an elastic matching scheme (stretching and bending drawn curves) is used. These
15 methods tend to be computationally intensive.

Alternatively, pen strokes can be divided into basic components, which are then connected by rules and matched to characters. This method is called Analysis-by-Synthesis. Similar systems use dynamic programming methods to match real and modeled strokes.

20 Existing solution for gesture recognition used the neural network method, image processing methods or chain code method. Previously the algorithm described in the patent application whose title is 'Real Time Gesture Recognition', uses the chain code method as a primary base for extracting the features and then processing the features for recognition and matching. This previous method is indeed a new
25 invention but with some compromise on the accuracy.

SUMMARY OF THE INVENTION

The accuracy obtained in the method invented in the patent 'Real Time Gesture Recognition' achieves less than 80% accuracy. This method is suitable for only straight lines, but when curves need to be detected, the template size increases as a result of accommodating many possibilities of curve directions. Also this method cannot differentiate between spirals and normal curves. To solve the above problems and to improve the accuracy of the recognition and matching, the new curve detection in real time gesture recognition method was invented.

To solve the above problems, according to the present invention, a curve detection method is added to the real-time symbol gesture recognition method as shown in Figure 1, comprising:

1. A second pre-processing step.
2. Curve Analysis based on the extracted features.
3. Line Segmentation
4. Line and Symbol Template matching

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 shows the overall process of the algorithm used in the present invention for the curve detection in real time gesture recognition.

Figure 2 specifies the curve categories.

Figure 3 gives the expel curve directions.

Figure 4 explains the process of finding the curve orientation.

Figure 5 illustrates an example for cases with internal intersection.

Figure 6 shows the line template stucture.

Figure 7 shows the extra byte line data format for closed symbols

Figure 8 shows the extra byte line data format for spirals.

Figure 9 shows the extra byte line data format for lines with internal intersection.

Figure 10 shows the table to find curve direction

Figure 11 shows the table to find Line Template Data direction significance.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

5

In the curve detection in real time gesture recognition algorithm, the initial steps of preprocessing (dehooking and smoothing) and feature extraction remain unaltered when compared to the algorithm described in the patent application whose title is 'Real Time Gesture Recognition'. After the end point is obtained, and all the features are extracted, a second preprocessing step is performed. After this, the normalization steps are performed (recalculation and reordering), followed by the intersection check, curve analysis and the line segmentation steps. The connection check step in the algorithm has been removed and the template format of the Line is changed, to cater to the new features that are extracted. In Figure 1, the dehooking and smoothing steps form the preprocessing 101 and the feature extraction step 102 consists of orientation estimation, second preprocessing, recalculation and reordering and curve analysis and line segmentation.

The Second Pre-Processing is performed once all the directions are extracted and the end point is achieved, the extracted directions are pre-processed again. A threshold length is calculated based on the actual length of the line. The length of each line segment between the extracted directions is calculated. If this length is shorter than the threshold length, that particular extracted direction is discarded.

In the Curve Analysis step, in order to extract and match curves, curves are classified as either clockwise or anti-clockwise, as shown in Figure 2, based on the starting and end point of curve obtained after reordering. Each curve is associated with a possible set of directions, which make up the curve. An example is illustrated in Figure 3 for both clockwise and anticlockwise curves. Hence the given curve can be inferred as in clockwise or anticlockwise direction based on the orientations obtained after reordering.

Further, if the given symbol has just one curve without being in combination with other lines or curves, then the symbols can be either a six or a nine or a spiral or just an arc facing a particular direction.

5

Spiral (Full Spiral or Six or Nine) Detection:

The following steps are involved in Spiral Detection:

- 10 1. Find whether the line obtained by joining end point and start point intersects with any other line of the symbol.
2. If the number of intersections is only one then it is a 6 or 9, which can be determined by:
 - (1) Intersection in downward direction => spiral at bottom => 6
 - 15 (2) Intersection at upward direction => spiral at top => 9
3. If the number of intersections is more than 1, then the symbol is a Full Spiral.
4. Hence, based on the curve (clockwise or anticlockwise), number of intersections and the direction of spiral, the symbol is judged
- 20 to be either a 6 or a nine or a full spiral or none of the above.

Curve Direction Detection:

Once it is known that the symbol consists of only one curve with no other combinations and that it is not a spiral, then it has to be an arc facing a particular direction. Method to determine the direction of orientation of the arc is as follows:

1. First, find the center point of the curve. Steps are depicted in the Figure 4a.
- 30 2. Next, find the midpoint of the line joining the start and the end point of the symbol as shown in Figure 4b. Finding the slope of

line between center point and midpoint helps us determine the orientation of curve, Figure 4c.

3. According to the slope obtained from figure 4d, the orientation of the curve can be estimated as shown in Table of Figure 10, where X_I and Y_I are coordinates of the midpoint of line joining start and end point, X_C and Y_C are coordinates of the center point of the curve.

4. If the direction is E or F referring to Figure 11, i.e. a single curve in either clockwise or anti-clockwise direction, but has one intersection, it can be either a 4 or a 6 or alpha. In such cases, we first find the distance between the intersection point and the end points of the line segments involved in the intersection. Then a point is found on the line with longer distance, such that the point is of the same distance from the intersection point as the other line. Thus the mid-point of the line between the newly found point and the end point of the shorter line is found. The orientation of the line joining the point of intersection with the midpoint is found. Example symbols are 4, 6 (with an intersection) and alpha. The orientation is stored as a separate byte in the line template.

Illustration is available in figure 5a and 5b where X is Point of Intersection of line AB and DC

Since distance $XD < XA$, Point E on XA is such that $XE = XD$. M is the midpoint of the line DE. Thus the orientation of XM is found and stored.

Line Segmentation is performed before the matching process is started, when the extracted directions from the input lines are analyzed and grouped into arcs and lines.

Figure 6a shows the Line Template Header format and Figure 6b shows the data that is stored in the line template, which are the extracted directions. The meaning of direction number is shown in Figure 11.

Further, if Direction is 8 (a closed symbol) then it is followed by another 2 bytes as shown in Figure 7a and 7b.

The 1st Byte is as shown in Figure 7a, where

ABCD represents the minimum number of feature vectors of the symbol and

EFGH represents the maximum number of feature vectors of the symbol.

The 2nd Byte is as shown in figure 7b, where

EFGH represents the number of 90 degrees present in the closed symbol.

If Direction is 9 (a spiral) then it is followed by another 1 byte as shown in Figure 8, where

B- 1: Full Spiral; 0: Half Spiral

C- 1: Check R/L Spiral Condition; 0: Do not check R/L Spiral Condition

D- 1: Right Spiral; 0: Left Spiral

E- 1: Check T/D Spiral Condition; 0: Do not check T/D Spiral Condition

F- 1: Top Spiral; 0: Bottom Spiral

G- 1: Check Spiral Direction; 0: Do not check Spiral Direction

H- 1: Clockwise Direction; 0: Anti-Clockwise Direction

If the direction is E or F, i.e. a single curve in either clockwise or anti-clockwise direction, but has one intersection, it can be either a 4 or a 6 or alpha. In such cases, we first find the distance between the intersection point and the end points of the line segments involved in the intersection. Then a point is found on the line with longer distance, such that the point is of the same distance from the intersection point as the other line. Thus the mid-point of the line between the newly found point and the end point of the shorter line is found. The orientation of the line joining the point of intersection with the midpoint is found. The

line data byte is as shown in Figure 9, where

E- 1: Check for the orientation specified by FGH

FGH: Specifies the orientation.

- 5 Advantages of the Invention as compared to the prior method are that, this invention achieves greater accuracy of around 95%. Further there is a greater variety of symbols that can be detected and the arc or curve detection is excellent.

Claims:

1. A gesture recognition method for an electronic system built with a processor, a memory, a screen and an input device for accepting coordinates of drawn lines, comprises, after the steps of pre-processing and feature extraction steps:

- (a) a second pre-processing step,
- (b) curve analysis step based on the extracted features,
- (c) line segmentation step, and
- (d) line and symbol template matching step,

wherein the step (b) comprises determining presence of curve based on feature directions after recalculation and reordering steps.

2. The method according to claim 1, wherein the step (a) comprises:
determining a threshold length based on the actual length of the line,

determining a length of each line segment between extracted directions and

if this length is shorter than the threshold length, discarding particular extracted direction.

3. The method according to claim 1, wherein the step (b) further comprises classifying curves as either clockwise or anti-clockwise based on starting and end point of curve and orientations obtained after reordering.

4. The method according to claim 3, wherein the step (b) further comprises classifying given symbol, if it has just one curve without being in combination with other lines or curves, into one among a six, a nine, a full spiral, a closed symbol or just an arc facing a particular direction.

5. A gesture recognition method for an electronic system built with a

processor, a memory, a screen and an input device for accepting coordinates of drawn lines, comprises, after the steps of pre-processing and feature extraction steps:

- (a) a second pre-processing step,
- (b) curve analysis step based on the extracted features,
- (c) line segmentation step, and
- (d) line and symbol template matching step,

wherein the step (b) comprises following steps to analyze spiral:

finding whether line obtained by joining end point and start point intersects with any other line of the symbol,

based on the curve which is either clockwise or anticlockwise, number of intersections and direction of spiral, determining the symbol to be one among a six, a nine, and full spiral or none of the above.

6. The method according to claim 5, wherein the step (a) comprises:

determining a threshold length based on the actual length of the line,

determining a length of each line segment between extracted directions and

if this length is shorter than the threshold length, discarding particular extracted direction.

7. A gesture recognition method for an electronic system built with a processor, a memory, a screen and an input device for accepting coordinates of drawn lines, comprises, after the steps of pre-processing and feature extraction steps:

- (a) a second pre-processing step,
- (b) curve analysis step based on the extracted features,
- (c) line segmentation step, and
- (d) line and symbol template matching step,

wherein the step (b) comprises following steps to detect curve direction:

finding center point of the curve, midpoint of the line joining the start and the end point of the symbol, and finding slope of line between center point and midpoint, so as to determine orientation of curve.

8. The method according to claim 7, wherein the step (a) comprises:
determining a threshold length based on the actual length of the line,

determining a length of each line segment between extracted directions and

if this length is shorter than the threshold length, discarding particular extracted direction.

9. A gesture recognition method for an electronic system built with a processor, a memory, a screen and an input device for accepting coordinates of drawn lines, comprises, after the steps of pre-processing and feature extraction steps:

(a) a second pre-processing step,

(b) curve analysis step based on the extracted features,

(c) line segmentation step, and

(d) line and symbol template matching step,

wherein the step (c) comprises analyzing and grouping extracted directions into arcs and lines.

10. The method according to claim 9, wherein the step (a) comprises:
determining a threshold length based on the actual length of the line,

determining a length of each line segment between extracted directions and

if this length is shorter than the threshold length, discarding particular extracted direction.

11. A gesture recognition method for an electronic system built with a

processor, a memory, a screen and an input device for accepting coordinates of drawn lines, comprises, after the steps of pre-processing and feature extraction steps:

- (a) a second pre-processing step,
- (b) curve analysis step based on the extracted features,
- (c) line segmentation step, and
- (d) line and symbol template matching step,

wherein the step (d) comprises following step to match line:

matching feature vector of the line with reference feature vector stored in line database, number of bytes depending on the type of the symbol.

12. The method according to claim 11, wherein the step (a) comprises:

determining a threshold length based on the actual length of the line,

determining a length of each line segment between extracted directions and

if this length is shorter than the threshold length, discarding particular extracted direction.

13. The method according to claim 11, wherein the step (d) further comprises:

adding two bytes of information for a closed symbol specifying maximum and minimum number of feature vectors of the symbol and the number of 90 degrees present, and for a spiral, adding one byte specifying whether it is a full or half spiral, orientation of spiral being right, left, top or bottom and direction being clockwise or anti-clockwise.