An algorithm to solve the draw issue of number depth recognition

Zhenhui Li[1, a], Xia Zhang[2,b]

[1] School of Information Engineering, Communication University of China, Beijing 100024, China;

[2] School of Information Engineering, Communication University of China, Beijing 100024, China

[a]13051352205@163.com, [b]zhangxia@cuc.edu.cn

Keywords: Depth recognition, Upper and lower deviation, hook

Abstract. This paper discussed that an algorithm solves the draw issue of number depth recognition. Depth recognition methods have used many fields and have wide development prospect. Many scientific research institutions and scholars are studying the depth recognition. When user writes in an MR system, a depth threshold based method is used to decide whether user’s finger has touched the table surface or not. Due to the accuracy limit of finger detection in the depth based system, this method may cause problems. To remove these extra points and at the same time keep real trajectory points as much as possible is the target of this task. This paper presents a new algorithm to solve the draw issue of number depth recognition.

1. Introduction

Depth of handwriting recognition refers to the written on handwritten equipment when the orderly track, the system automatically identify the corresponding characters. Actually, to convert the coordinates of the handwritten tracks to digital code is a mapping process, and the handwriting recognition is the most natural human computer interaction, and one of the most convenient means. Along with the mobile information tools such as smart phones, PDA's popularization, the handwriting recognition technology has entered the era scale application. Depth of handwriting recognition enables users to input text input by nature and the most convenient way, easy to learn and easy to use, which can replace the keyboard or mouse. Depth of handwriting recognition have wide development prospect and is worth further research value. Depth of handwriting recognition belongs to the category of character recognition and pattern recognition. Character recognition from the recognition process is divided into offline identification and online identification. [1]In the number depth recognition fields, there are a lot of has been applied to our real life. Such as road traffic sign digit recognition, vehicle license plate number recognition and so on.

2. Problem analysis

When user writes in an MR system, a depth threshold based method is used to decide whether user’s finger has touched the table surface or not (Indicated by Touch/Release event). Because of the accuracy limit of finger detection in the depth based system, this method may cause problems such as:

a. Extra trajectory points in the start and end part of a real writing stroke;

b. Extra trajectory points of connecting stroke between two real writing strokes.

These problems are referred to as the draw issue problem. To remove these extra points and at the same time keep real trajectory points as much as possible is the aim of this task. This paper solves first problem.

A stroke is generated between a Touch and Release event. So the bias between real trajectory and automatically segmented trajectory is essentially the bias between start/end point of GT and Touch/Release event point. Bias distance between a Touch event point and GT start point is called header part bias distance, and that between Release event point and GT end point is called tail part bias distance. As shown in Fig. 1. The metric of bias distance is used to evaluate the performance of draw issue problem solution method.

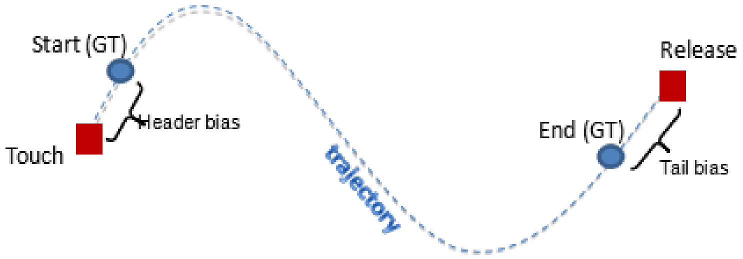


Fig. 1 Draw issue problem explanation

After checking, 1862 auto segmented samples can be successfully aligned with GT labels. (Actually a string which contains multiple character samples is discarded if it cannot generate the same number of segments by Touch/Release event, compared with GT segment labels) [3]

Table1.Statistics of dataset

Component	value
#GT	1862
Average length of GT character	2412.5
Accumulative bias dist.	34246
Accumulative header bias dist.	10400
Accumulative tail bias dist.	23846
Average bias distance per character	18.39

3. Method for removing extra trajectory near Touch/Release points.

The simplest way to solve draw issue problem is to ignore a fixed length of trajectory near Touch/Release event. However it may not work if (a). The draw issue part is long; removing a relative length of trajectory is not enough; (b).Or even worse if the real trajectory is short, since it will remove a large part of the real trajectory. Three results is list on table 2, which are aiming at minimizing accumulative bias distance, accumulative inner bias distance and outer bias distance, respectively.

Table 2.Result of removing fixed length of trajectory near Touch/Release point

Minimizing target	all bias dist.	inner bias dist.	outer bias dist.	header bias dist.	tail bias dist.
all bias dist.	16393	11743	4650	5314	11079
inner bias dist.	34246	0	34246	-	-
out bias dist.	46995	46995	0	-	-

4. Method based on Upper and Lower deviation (ULD)

In mechanical engineering, dimensioning parts on the basis of the basic value have upper and lower deviation. Such deviation can not only make up and down when the parts processing provides redundancy, but also provide the convenience for parts and parts of, so it can be introduced to here. Inner bias and outer bias are added around GT. If we do this, it can obtain much convenience to handle hook. So we can make this kind of algorithm named ULD. [2]

Bias distance is used as the metric to evaluate the performance of solutions. Bias distance is divided into inner and outer ones, depending on whether the segmented point generated by draw issue trimming algorithm is inside or outside GT segments (As explained by Fig. 2).

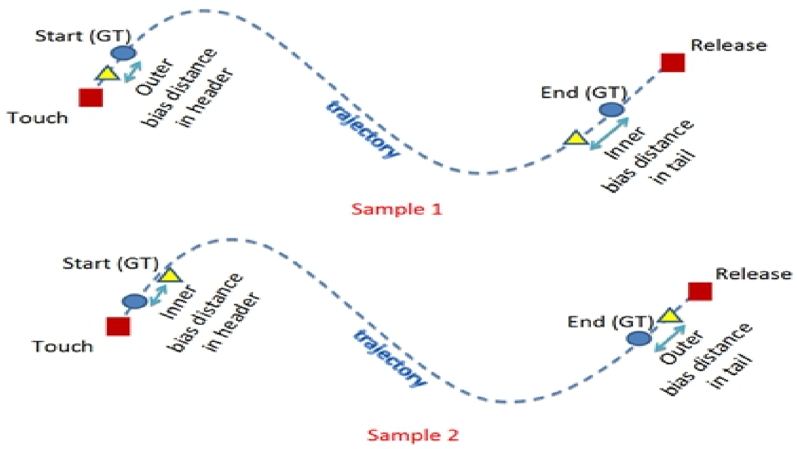
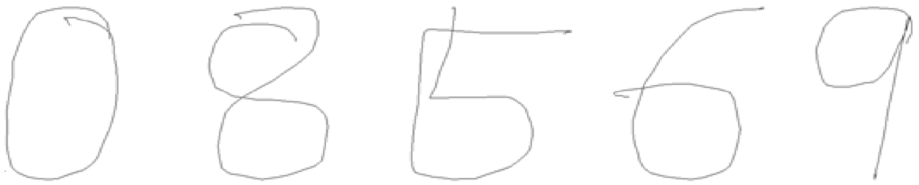
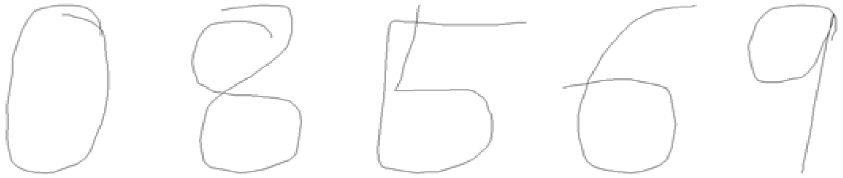

Start (GT)
Touch
Outer bias distance in header
trajectory
Release
End (GT)
Inner bias distance in tail
Sample 1
Start (GT)
Touch
Inner bias distance in header
trajectory
Release
End (GT)
Outer bias distance in tail
Sample 2

Fig. 2 inner and outer bias distances

Now it is used to solve the problem. It provides a much safer way of trimming draw issue part. To measure the inner and outer bias is a problem and a method. It was done on the basis of turning angle. Calculating the turning angle between two adjacent segments, when the angle is greater than a threshold and accumulative length is lower than another threshold, a hook is found and will be removed.



Trajectories before ULD



Trajectories after ULD

Table3 Experiment result of ULD

all bias dist.	inner bias dist.	outer bias dist.	header bias dist.	tail bias dist.
17741	4424	13317	8678	9063

5. Conclusion

Table 4 lists the comparison result to summarize.

Table4 Draw issue problem summarization

	All bias dist.	Inner bias dist.	Outer bias dist.	Header bias dist.	Tail bias dist	Average bias_dist/char
Draw issue	34246	0	34246	10400	23846	18.06
Trimming fixed length of trajectory	16393	11743	4650	5314	11079	8.8
ULD	17741	4424	13317	8678	9063	9.53

(1). Problem is more complex in the tail part than that in the header part.

-That means styles of leaving the table are more complex than that of touching the table, or hooks occur more frequently at the end of strokes.

(2). ULD can achieve much smaller inner bias distance.

-Although removing fixed length of trajectory can achieve smaller accumulative bias distance, it also yields large inner bias distance. That means it will risk of losing real stroke points.

(3). After ULD, average bias distance per character is 17741 / 1862 = 9.5.

-Ratio (bias dist/char length) = 9.5 / 2412 = 3.9‰. For depth base number recognition, draw issue problem is too minor to affect recognition result. However, it may affect recognition of characters written in the style of small size.

(4). After ULD, the remaining issues are mainly caused by the reasons below *(red* points are the GT points, and Green ones are the result of ULD method):

a) Jump points**.** Jump points exist and they can cause zigzag redundant trajectory (about 1300 accumulative bias distance is caused by this problem).

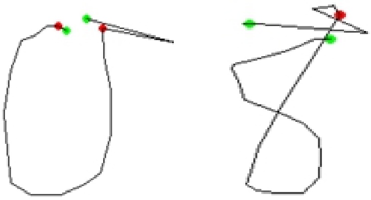


Fig.2 Samples with jump points.

b) Difficult cases. Sometimes, although there is large bias distance in a trajectory, however, for these cases, it's hard to trim draw issue part by algorithm. If the draw issue is trimmed like this, we will probably trim a real part in a trajectory. These are the cases such as in Fig. 3.

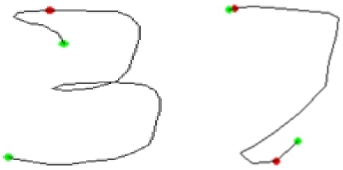


Fig.3 Samples that is hard to trim draw issue part automatically.

c). Inaccurate of GT labeling. Even accurate draw issue trimming result cannot insure the result segmented point is exactly the identical GT points. Let along cases that GT points are hard to label.

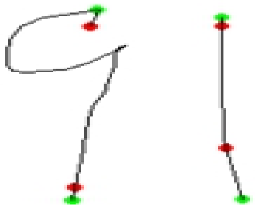


Fig. 4 Samples with ambiguous GT points.

References

[1] Zohre Sadrnezhad, Atefeh Nekouie, Majid Vafaei Jahan. Online handwriting Farsi character and number recognition based on hand movement direction using Hidden Markov Models Lupton (ICTCK 2014) November, 26-27.

[2] JinHong Han, Changchun Wang, Fundament of Interchangeability and Measurement, first ed, Peking University Press, Bei Jing, 2006.

[3] S. M. Razavi, and E. Kabir. A dataset for online Farsi handwriting. sixth conf. on intelligent systems, CIS 2004, pp. 228-225.