문자 인식에서 분할 비용에 따른 문자 분할 연구

정 민 철

상명대학교 컴퓨터시스템공학과

Character Segmentation with Segmentation Cost in Optical Character Recognition

Minchul Jung

Department of Computer System Engineering, Sangmyung University

요 약

인쇄체 문자 인식에서 접합 문자는 주요한 에러 발생의 원인이다. 본 논문에서는 접합 문자를 분할하기 위해 두 개의 분할 비용을 정의한다. 첫째, 절단 비용은 한 패턴을 분할하는 데 얼마나 많은 블랙 픽셀이 분리되어야 하는가이다. 둘째, 접선 비용은 분할선이 얼마나 많은 블랙 픽셀과 화이트 픽셀 사이를 지나가는가이다. 폰트 분류기는 접합 문자의 후보 문자를 제공한다. 후보 문자의 문자 폭은 접합 문자를 분리하기위한 기준선을 제공하며, 그 기준선 부근의 픽셀들이 분할 가능 영역을 나타낸다. 절단 비용의 최소값과 접선 비용의 최대값이 되는 지점이 최종적으로 접합 문자를 분할하는 위치이다. 이렇게 정의된 절단 비용과 접선 비용을 가지고 접합 문자를 분할하면 보다 정확한 문자 분할을 하여 문자 인식에서 에러 발생을 줄일 수 있다.

1. Introduction

Character segmentation is to partition word images into isolated and complete characters, which in turn serve as input to a character recognizer. Segmentation procedures are more heuristic in nature than recognition procedures.

Character segmentation is fundamental and critical to character recognition, since character recognition relies on isolated characters and incorrectly segmented characters seldom are correctly recognized. Touching characters are responsible for the majority of errors in the machine-printed character recognition, since touching characters make it difficult to extract the exact set of features needed for the identification. Imperfect segmentation between adjacent characters accounts for the majority of misclassifications. More than half of all recognition errors come from touching characters.

As long as characters are correctly segmented, the degradation of individual characters does not significantly affect the overall system performance. If recognition is unacceptable, generally it is because characters were difficult to segment [1].

Some touching characters cannot be segmented unless the individual characters are first recognized. However, classification algorithms generally require an isolated

character input. In other words, proper character segmentation requires prior knowledge of which pattern forms a meaningful unit. This means that, although character recognition requires character segmentation as a previous step, character segmentation itself requires a character recognition capability [2].

The above statements might have the relation of "chicken and egg" but it is necessary to attack both of these problems simultaneously. Without understanding the symbols, there are no good criteria to avoid errors of segmentation. The segmentation of touching characters in a variable pitch font is one of the major remaining technical problems in OCR systems

This paper defines two character segmentation costs, which are a cutting cost and a tangent cost. These costs find an optimal segmenting path to segment touching characters. These costs can find an optimal segmenting path to segment touching characters and improve the performance of character segmentation.

2. Segmentation cost

2.1 Cutting cost

The width of a character in touching characters changes slightly according to a binarization threshold, a size normalization algorithm, and random noises. A cutting cost is defined to solve the problem.

The cutting cost is "*how many black pixels should be apart* in order to segment a pattern."

The width of a candidate character gives only a reference line to segment. A few neighboring columns of the reference line gives a possible cutting area. The cutting cost decides the best segmentation line in the area.

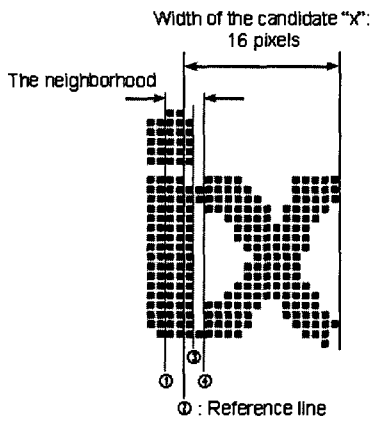

Figure 1. Cutting cost: the line ① has 22, the line ② has 21, and the line ③ and the line ④ have 3.

Figure 1 illustrates cutting costs in the segmentation. In Figure 1, the candidate 'x' gives 16 widths for the segmentation position, the line ②. This segmentation is incorrect.

However, if we consider the cutting cost, the segmentation will be correct.

That is, the reference line ② gives the neighborhood from the line ① to the line ④, and every cutting cost in the neighborhood, including the reference line, is calculated.

For example, the cutting cost of the line ③ and of the line ④ is 3 while the cutting cost of the reference line is 21.

The segmentation line should have the minimum cutting cost and be the nearest to the reference line. Therefore, the line ③ is selected in this example.

2.2 Tangent cost

In Figure 2, the reference line ① and the line ② have the same cutting cost of 4. Since the reference line itself is the nearest to the reference line - the closest to the width of a prototype, the reference line ① is chosen to segment.

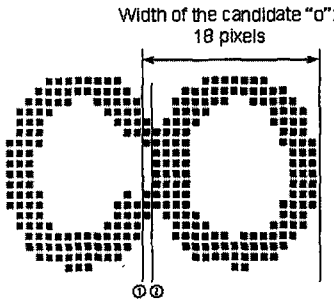

Figure 2. Tangent cost: the line ① has 2 and the line ② has 4.

In that case, the segmented character 'o' includes the part of character 'c' since it has been segmented improperly.

However, if we introduce the concept of a tangent line, this problem will be solved with ease. A tangent line is mathematically defined as a straight line touching a curve. In this research, it is defined as a boundary line between black pixels and white pixels.

A tangent cost is defined as *"how many black and white pixels* a line passes by consecutively".

The tangent cost makes it possible to find a tangent line in touching characters. The line ① has 2 consecutive pixels between black and white, and the line ② has 4 consecutive pixels. The line ② that has a greater tangent cost is selected as the segment line.

3. Conclusion

This paper presented a new character segmentation method that used two character segmentation costs, which are a cutting cost and a tangent cost. The cutting cost is defined as how many black pixels should be apart. The tangent cost is defined as how many black and white pixels a line passes by consecutively.

The cutting cost is finding the minimum cost to cut black pixels, and the tangent cost is looking for the maximum cost to cut black and white pixels.

Back to the Figure 1, we can see that the line ③ has the minimum cutting cost and also has the maximum tangent cost.

With these two segmentation costs, character segmentation can find an optimal segmenting path to segment touching characters and improve the performance of character segmentation.

4. References

[1] M. Bokser, "Omnidocument technologies", *Proceedings of the IEEE*, Vol. 80, No. 7, pp. 1066-1078, 1992.

[2] H. Fujisawa and Y. Nakano and K. Kurino, "Segmentation methods for character recognition: From segmentation to document structure analysis", *Proceedings of the IEEE*, Vol. 80, No. 7, pp. 1079-1092, 1992.

[3] R. Rubinstein, "Digital Typography: An Introduction to type and composition for computer system design", *Addison-Wesley*, 1988.