A NEW APPROACH TO SEGMENT HANDWRITTEN DIGITS

Oliveira, L.S.; Lethelier, E.; Bortolozzi, F.; Sabourin, R.

Published in:
EPRINTS-BOOK-TITLE

A NEW APPROACH TO SEGMENT HANDWRITTEN DIGITS

L. S. Oliveira[1], E. Lethelier[1], F. Bortolozzi[1] and R. Sabourin[2]
[1]*Pontifícia Universidade Católica do Paraná - LARDOC*
Rua Imaculada Conceição 1155, 80215-901 - Curitiba, PR - BRAZIL
[2]*Ecole de Technologie Superieure*
1100, rue Notre Dame Ouest, Montréal (Quebec) H3C 1K3, CANADA
E-mail: {soares,edouard,fborto}@ppgia.pucpr.br, sabourin@gpa.etsmtl.ca

This article presents a new segmentation approach applied to unconstrained handwritten digits. The novelty of the proposed algorithm is based on the combination of two types of structural features in order to provide the best segmentation path between connected entities. In this article, we first present the features used to generate our basic segmentation points. Then, we define our segmentation paths depending on the encountered configurations with only few heuristic rules. Finally, we evaluate the output of our segmentation by using a combination of classifiers.

1 Introduction

The challenge of a segmentation technique lies in the decision of the best cut path to localize an entity to be recognized as a correct isolated character by the recognition system. The literature usually shows three different strategies to perform the segmentation : Segmentation-Recognition, [1] Segmentation-based Recognition [2] and Segmentation-Free systems [3].

Our works are based on the segmentation-based recognition strategy. The aim of this article is to show how we defined a new segmentation algorithm taking into account two complementary sets of structural features. The final objective of the module is to provide the best hypotheses list of segmentation paths without any a priori knowledge of the context, such as the number of characters to be segmented. Therefore we focused our segmentation work on the limitation of heuristic rules to consider most of configurations in connected characters. We worked with binary images with a 300 dpi resolution.

2 Generation of the Segmentation Features

In character segmentation, the choice of the features usually depends on the pecularities of the context. Since we need to express a large directional variability of the character strokes (touching digits), we chose 3 types of features: contour, profile and skeleton.

The contour is a bi-dimensional data where each contour point C_i is associated with the coordinates (X_i, Y_i) of the image. The profile image is obtained from a vertical projection of the first encountered transition, in both ways

In: L.R.B. Schomaker and L.G. Vuurpijl (Eds.), Proceedings of the Seventh International Workshop on Frontiers in Handwriting Recognition, September 11-13 2000, Amsterdam, ISBN 90-76942-01-3, Nijmegen: International Unipen Foundation, pp 577-582

top-down and bottom-up. From these both sets of features, we are able to localize the first list of potential cuts which correspond to the local minima of the contour and profile (Figure 1a and 1b). We define these points as *Basic Points* (BPs).

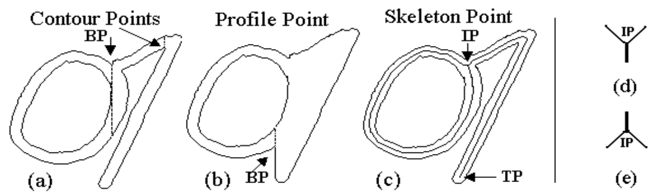

Figure 1: Segmentation points generated:(a) Contour (b) Profile (c) Skeleton Points (Intersections and Terminations) - Skeleton intersection points: (d) Class 1 (e) Class 2

Let us define now characteristics of the skeleton :

Definition 1: A "Terminal Point" (TP) is a point with a single black pixel in its 8-neighbors while an "Intersection Point" (IP) is the point with more than 2 black pixels in its 8-neighbors. The TP and the IP are called *characteristic points* of the skeleton.

Definition 2: A skeleton path ($P_{skeleton}$) is a pixel sequence of the skeleton where each extremity corresponds to a *characteristic point*.

During the detection process of the IPs, two particular configurations are first selected as showed in the second part of the Figure 1. We have noticed that particular digit connections, such as "00" for example, often generate these IPs classes. All other types of IPs are considered as a single IP class (Class 3). The first class (Figure 1d) contains all IPs with one segment in its lower part and two segments in its upper part. Depending on the Freeman directions, each class contains all the variations considering this definition. The second class (Figure 1e) is the symmetric of Class 1.

3 Determination of Segmentation Paths

Once selected from the image, the BPs and the IPs are compared altogether in order to determine the list of segmentation hypotheses. The algorithm scans all possible relationships between BPs and IPs and generates a set of segmentation paths where the goal is to get the correct segmentation paths (for connected characters) in the list of hypotheses.

This association takes into account the Euclidian distance between BPs and IPs. Both entities could indicate a possible stroke connection between two characters. To determine the proximity between points, we based our comparison on the estimation of the thickness of the strokes E_t, obtained with the projection of the density histogram. Then, two points BP_i and IP_j belong to the same neighborhood if :

$$d_E(BP_i, IP_j) \leq E_t \tag{1}$$

or

$$d_E(proj_y(BP_{ik}), IP_j) \leq E_t \quad \text{for} \quad k = 1, 2 \ldots n \tag{2}$$

is verified, where d_E is the Euclidian distance, $proj_y(BP_{ik})$ is the vertical projection of BP_i at the step k on the segment whose height is n. Note that (2) is checked only if (1) is not verified. The Figure 2 shows both configurations of neighborhood verification.

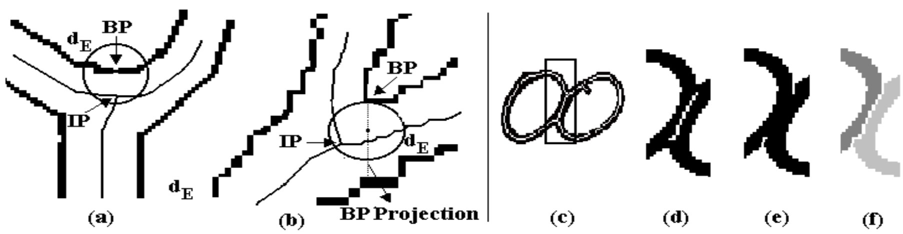

Figure 2: Distance verification between the BP and IP (a) Distance verified by equation 1 (b) Distance verified by equation 2 - Particular conections: (c) Skeleton (d) $P_{skeleton}$ (e) P_{bp-ip} (f) P_{seg}

Depending on the local configurations of points, the segmentation path (P_{seg}) can be generated straight away from the $P_{skeleton}$ or being orthogonal to it. In the first case, the system has found particular points of class 1 and class 2, as described in Figure 1 and tries to link the sequence of $P_{skeleton}$ between these two points. The second part of Figure 2 shows an example where $P_{skeleton}$ (Figure 2d) is associated with the joining segments between BPs and IPs, denoted (P_{bp-ip}) (Figure 2e) to form the final and correct segmentation path (P_{seg}) (Figure 2f). Thus, we can define the segmentation path : $P_{seg} = P_{skeleton} \bigcup P_{bp-ip}$

In the second case, the segmentation path is orthogonal to $P_{skeleton}$. To determine the possible cuts around each IP, the algorithm performs the following tests : (a) If the considered segment of the skeleton is a stroke-end (TP exists on the segment), then the cut is allowed if the segment length is significant; (b) The cut length should be minimum in the neighborhood of IP. To find the best position, the cut is evaluated in $2r_e$ pixels, where r_e is the IP radius, for the minimum distance where a transition is encountered.

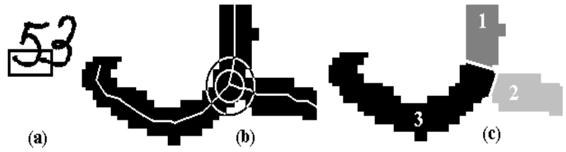

Figure 3: Orthogonal cuts: (a) Original image (b) Potencial region (c) Cuts realized

The Figure 3b shows the specific regions where the search of the cut path is performed around the IP. For each pixel of the $P_{skeleton}$ included in the biggest circle, the orthogonal cut path is evaluated and compared to the previous minimum length previously calculated. Finally, the algorithm performs the cut path for each segment around the IP. When the configuration is not verified by the first test, the cut is not performed (Figure 3c) for the segment 3. In some cases, none IP is detected in the image, even if it exists a connection between two characters. Then, the algorithm provides a cut path from the detected BPs in order to avoid segmentation errors.

4 Evaluation of the Segmentation

In order to evaluate our segmentation algorithm, we used three Multi-Layer Perceptrons (MLPs), trained on three types of features : Concavity Measures (k_1) [4], Edge Maps (k_2) [5] and a mix Concavity - Contour (k_3). This classifiers are combined using 3 strategies (Media, Product and Cascade) in order to improve the efficiency and accuracy if the recognition module [6]. The training database contains 9,500 images (10 classes) of naturally isolated handwritten digits extracted from our laboratory database of 2,000 brazilian checks. The evaluation was runned with another set of the same database splited into two sub-bases : 900 images of connected digits (pairs and triples for the most of them) and 3,500 images of naturally isolated digits. The latter base was used to compare the effects of the segmentation module on the classification performances. With the manual verification of the segmentation, 98.5% of connected characters were correctly segmented. Most of the segmentation errors are caused by the lack of BPs in the neighborhood of the connected strokes (see Figure 4c). The automatic evaluation, expressed in Table 1, shows the recognition results for both subsets and for each strategy of combination. For these both experimentations, each classification mode proceeded the same rejection threshold.

Considering the subset of isolated digits, the three combination modes improve the performances of the classifiers. The best result is performed by the

Table 1: Classifiers Performance

Classif Mode	*Isolated (%)*				*Connected (%)*			
	Rec.	*Err.*	*Rej.*	*Reab.*	*Rec.*	*Err.*	*Rej.*	*Reab.*
k_1	90.74	0.25	8.99	99.71	84.83	6.98	8.18	92.39
k_2	90.05	2.98	6.95	96.78	83.18	13.29	3.15	86.21
k_3	90.16	0.22	9.61	99.75	85.84	7.41	6.74	92.05
Cascade	98.52	0.09	1.38	99.90	95.24	2.14	2.61	97.80
Media	92.97	0.09	6.92	99.89	70.60	10.15	20.90	89.26
Product	91.68	0.09	8.21	99.89	68.74	8.40	24.20	90.70

Cascade with 98.52% of correct classification and 1.38% of rejected patterns. This is also the best approach for of connected digits with 95.24% of correct classification and 2.61% of rejection. For other both combinations, we can observe the strong influence of the segmentation output, when the classifiers ar not trained with segmented digits.

The performance analysis of the connected characters subset lead us to define two categories of errors : segmentation errors and classification errors. We define a segmentation error as the best score configuration where one (or more) character is mis-segmented and provokes a mis-classification (see Figure 4a). A classification error is the best score configuration where the segmentation is correct but one (or more) character is mis-classified (see Figure 4b). In both cases, we hope soon improve the reliability of the classification module by adding segmented characters in the training database.

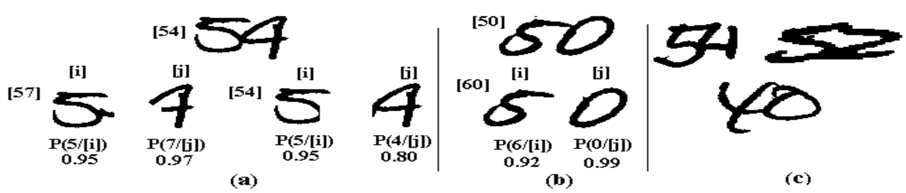

Figure 4: Cases of errors (a) Segmentation error (b) Recognition error (c) Cases not solved

This analysis shows the encouraging performances of our segmentation algorithm. With a limited number of rules, this technique is able to perform a correct segmentation in most cases of touching characters, even if the digits are strongly overlapped or skewed.If we compare with algorithms only based on contour and profile features[7], our method is more accurate in case of strong connections. Figure 5 shows examples of particular configurations where segmentation is not obvious with classical methods.

Figure 5: Correct segmentation in complex cases

5 Conclusion and Perspectives

We have presented a new segmentation-based recognition approach applied to unconstrained handwritten digits. The segmentation technique uses two sets of structural features to provide the possible segmentation paths, with any contextual information of the input image. Due to the small number of rules used in the cut path generation, the system is able to provide the correct segmentation in most cases of connected characters. The first results show that we can easily improve the recognition rate by considering the segmentation outputs in the training step of the recognizer. The cascade strategy will be used to yield this new database.

References

1. M.Shridhar and A.Badreldin. Recognition of isolated and simply connected handwritten numerals. *Pattern Recognition*, 19(1):1–12, 1986.
2. O. Matan and C.J.C.Burges. Recognizing overlapping hand-printed characters by centered-objects integrated segmentation and recognition. In *IJCNN*, pages 504–511, 1991.
3. S.Choi and I. Oh. A segmentation-free recognition of two touching numerals using neural networks. In *5⁰ ICDAR*, pages 253–256, 1999.
4. L.Heutte, J.Moreau, B.Plessis, J. Plagmaud, and Y.Lecourtier. Handwritten numeral recognition based on multiple feature extractors. In *2⁰ ICDAR*, pages 167–170, 1993.
5. Y. C. Chim, A.A. Kassim, and Y.Ibrahim. Dual classifier system for handprinted alphanumeric character recognition. *PAA*, 1:155–162, 1998.
6. J.Kittler abd M.Hatef, R.Duin, and J.Matas. On combining classifiers. *IEEE Trans on PAMI*, 20(3):226–239, 1998.
7. E.Lethelier, M.Leroux, and M.Gilloux. An automatic reading system for handwritten numeral amounts on french checks. In *3⁰ ICDAR*, pages 92–97, 1995.