

**POLYGONALISATION DE TRACÉ ET EXTRACTION DE
PRIMITIVES POUR LA RECONNAISSANCE EN LIGNE DE
MOTS MANUSCRITS**

T. VALLÉE, R. MULLOT, J. LABICHE

La3i-LACIS, Université de Rouen
Place Emile Blondel, 76821 Mont Saint Aignan Cedex, FRANCE
Tél: 35.14.60.00 fax: 35.14.63.49 Mail: vthierry@la3i.univ-rouen.fr

RÉSUMÉ

ABSTRACT

Les contraintes économiques et matérielles (portabilité) mettent en évidence les avantages de l'acquisition d'information sur tablette à digitaliser par rapport à une "lourde" saisie sur clavier.

Lors de la saisie de textes sur tablettes à digitaliser, l'information récupérée est généralement très dense et très bruitée.

L'article que nous proposons se situe dans le cadre de la reconnaissance de l'écriture manuscrite en ligne. L'objectif de notre étude vise à mettre un système de reconnaissance omniscriteur. Nous proposons un prétraitement de l'information issue d'une table à digitaliser qui permet à la fois d'éliminer les bruits d'acquisition, de polygonaliser le tracé et d'extraire simultanément des primitives stables et robustes en vue de la reconnaissance des mots.

Economic and material constraints (portability) put in a prominent position the advantages of the information acquisition on digital tablets compared with classical keyboard acquisition mode.

This article deals with on line handwritten recognition problem. This objective of our study is to provide an omniscriteur recognition device. While writing texts on a digital tablet, the acquired information is generally very noisy. In this paper, we propose a pre-treatment of the information issued from a digital tablet, pre-treatment which allows to filter acquisition noises, to polygonize strokes and to extract strong primitives for the following text recognition stage.

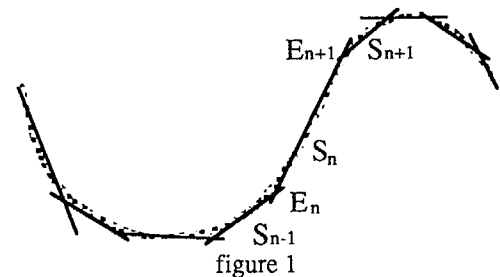
INTRODUCTION

Afin de faciliter l'utilisation des ordinateurs, une nouvelle interface est apparue [1]. Ce nouveau périphérique qu'est la table à digitaliser nécessite l'utilisation d'un système de reconnaissance de l'écriture cursive omniscriteur. Lors de l'utilisation d'une tablette à digitaliser, un prétraitement du signal obtenu est nécessaire. En effet par rapport aux systèmes hors ligne où le scripteur utilise le support naturel qu'est le papier, les systèmes en ligne utilisent un système de numérisation qui introduit un bruit sur le signal provoqué par le dispositif de numérisation et par le scripteur.

De plus, lors de l'acquisition, suivant la résolution de la tablette (de 100 à 1000 lpi) le nombre de points peut être important. Pour réduire ces points, une compression est nécessaire. Des méthodes de compression comme celle de Berthod [2] ou Bercu [3] nécessitent une étape de filtrage afin d'éliminer les bruits. D'autre méthode comme celle de Belaid [4] ou Ito et Chin [5] propose une méthode ayant un filtrage intégré mais perdent des points. L'article de Tappert et Suen [6] permet de faire une synthèse sur l'ensemble des techniques de prétraitement, en particulier sur les problèmes de réduction de bruit.

COMPRESSION DES POINTS

La méthode de polygonalisation que nous proposons, est basée sur une estimation de segments de droite (S_n) par double critère des Moindres Carrés présenté par Mullot [7] (figure 1).



Pour chaque nouveau point acquis par la tablette, deux droites D1 et D2 (figure 2) sont déterminées au sens des Moindres Carrés. Afin de rendre le traitement le plus rapide possible, seules 8 sommes sont calculées à chaque nouvelle acquisition. La première droite D1 est calculée suivant le critère des Moindres Carrés sur l'ensemble des points acquis depuis le dernier changement de modèle. Quant à la droite D2, elle sera estimée suivant le critère des Moindres Carrés sur les N derniers points acquis. Si l'angle entre D1 et D2 est supérieur à α , il y a alors détection d'un changement de modèle. Le renvoi du point d'intersection des droites D1 et D2 est retenu comme point de polygonalisation.

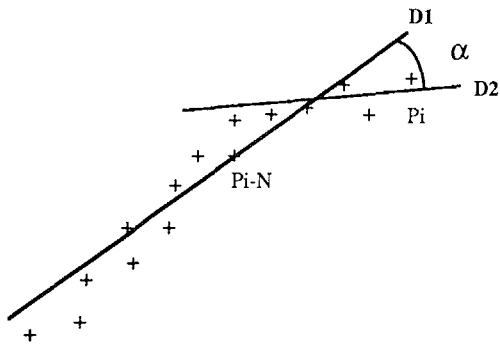
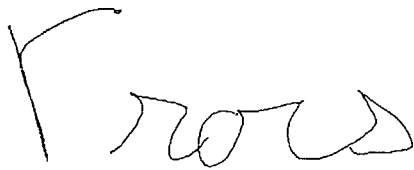


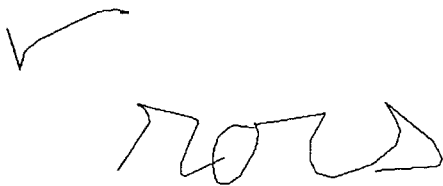
figure 2

Les 2 paramètres N et α sont respectivement le coefficient de filtrage et le degré de polygonalisation et permettent de régler la polygonalisation.



tracé original

- plus N est grand plus le tracé initial est filtré.

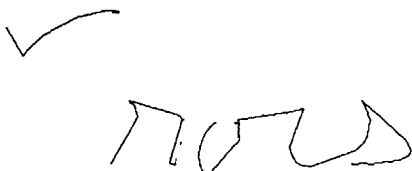


$n = 5, \alpha = 10^\circ$



$n = 15, \alpha = 10^\circ$

- plus α est petit, plus les points retenus sont nombreux, permettant d'approcher au plus près le tracé initial.



$n = 10, \alpha = 5^\circ$



$n = 10, \alpha = 15^\circ$

DÉTECTION DES RUPTURES

Mais comme la plus part des algorithmes classiques notre méthode de polygonalisation a un pouvoir trop filtrant. En effet, à l'issue des traitements, les points anguleux et les points de rebroussement sont atténués, voire éliminés. Pour éviter ce phénomène, nous avons développé un deuxième algorithme permettant la détection de ces points singuliers.

Pour chaque point acquis, une estimation d'une troisième droite $D3$ (figure 3) est effectuée. Cette droite est calculée suivant le critère des Moindres Carrés, sur les M points suivants le point courant.

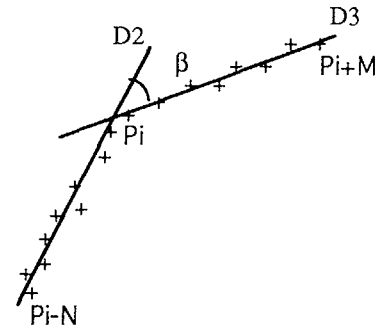


figure 3

Le contrôle de l'évolution de l'angle β entre $D2$ et $D3$ permet de détecter les points singuliers.

En effet, l'angle β évolue faiblement sur les portions courbe du tracé (figure 4),

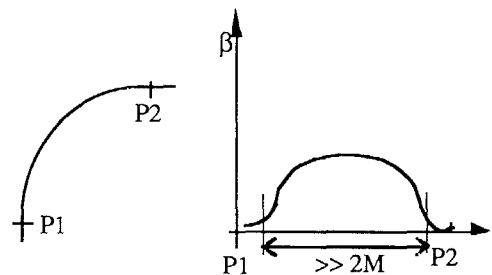


figure 4

alors qu'un point anguleux ou un point de rebroussement se traduit par une évolution de la courbe des évolutions de β (figure 5). Le maximum de l'angle β se trouve sur le point anguleux ou le point de rebroussement lui même. Pour que ce point singulier soit validé, il faut que la largeur de la courbe d'évolution de β soit proche de $2M$.

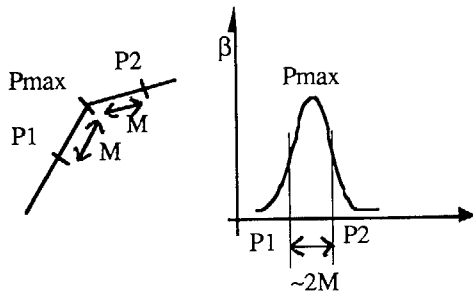
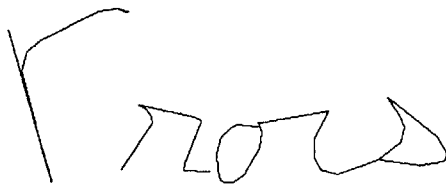


figure 5

Pour différencier les points anguleux des points de rebroussement, il suffit d'analyser la valeur maximum de β . Si ce maximum est proche de 90° et bien inférieur à 180° , un point anguleux est alors caractérisé. Si ce maximum est proche de 180° , il s'agit alors d'un point de rebroussement. Ces 2 points singuliers sont déterminés comme points d'intersection entre D2 et D3.



$n = 5, \alpha = 10^\circ, m = 5$



$n = 15, \alpha = 10^\circ, m = 15$



$n = 10, \alpha = 5^\circ, m = 10$



$n = 10, \alpha = 15^\circ, m = 10$

Les points obtenus grâce à ces 2 méthodes ne sont en aucun cas les points réels, mais les points optimum permettant d'obtenir un résultat de polygonalisation le plus proche possible du tracé initial global.

ANALYSE DES RÉSULTATS

On remarque que la première méthode est très efficace si le tracé comporte des lignes droites et des courbes, mais en revanche, "oublie" les points de rebroussement, et "arrondit" les points anguleux. Cependant la mise en concurrence des 2 méthodes permet à la fois de restituer les points de rebroussement, mais également de conserver le graphisme et la spécificité des points anguleux.

Cependant, il arrive que l'algorithme "surproduise" des points anguleux. ceci a lieu, en particulier sur des tracés courbes pour lequel le rayon est très faible. Il convient alors, soit d'augmenter l'angle minima de détection des points anguleux, soit d'augmenter la taille de la fenêtre de détermination des droites D2 et D3. Par contre, ce phénomène ne se retrouve en aucun cas pour les points de rebroussement.

Le choix des valeurs de N, α et M est primordiale afin d'obtenir une compression correcte (voir l'annexe). Dans notre application, le meilleur compromis compression - ressemblance du tracé a été trouvé pour les paramètres N, α et M respectivement à 8 points, 5° et 8 points.

CONSTRUCTION DU MOT

Au fur et à mesure de cette acquisition, une liste de points est créée comportant les primitives suivantes:

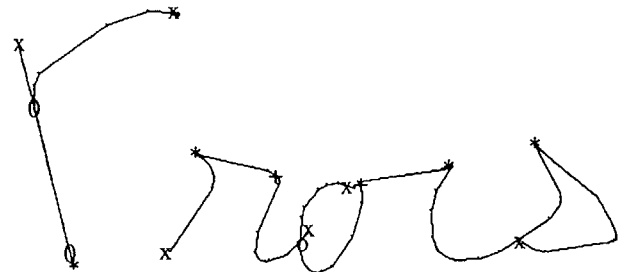
- points de départ et de fin de chaque trait,
- points de polygonalisation,
- points anguleux et de rebroussement,
- estimation du degré du rayon de courbure courant,
- historique du tracé.
- les points de croisement,
- les points de chevauchement.

La détection des points de croisement et des points de chevauchement est réalisée à partir d'une image du tracé que l'on construit lors de l'acquisition de chaque nouveau point polygonalisé. Cette image nous permet de caractériser la connexité et la topologie des mots.

Un point de croisement est détecté si le segment en cours de tracé en croise un autre déjà tracé.

Les points de chevauchement sont une concaténation de plusieurs points de croisement connexes constitués par 2 segments de tracé. Dans ce cas, nous mémoriserons le début et la fin du chevauchement.

L'extraction des points de chevauchement, renforce le poids des points de rebroussement, puisqu'après chaque point de rebroussement se trouve un chevauchement.



- x Point de départ et de fin de trait
- + Point anguleux
- * Point de rebroussement
- o Point de croisement
- 0 Point de chevauchement

Lorsque le scripteur a terminé l'acquisition de chacun des mots, on dispose donc d'une liste de points et d'une image comportant toutes deux des primitives pertinentes pour la reconnaissance.

CONCLUSION

Notre premier objectif qui était une compression temps réel du tracé initial sans perte d'information est atteint. Dans notre application nous obtenons un taux de compression moyen de 78%. De plus l'extraction de primitives pertinentes en même temps que l'acquisition nous permet d'envisager différentes méthodes de reconnaissance. Une liste de points permettant des traitements de type en ligne, et une image du mot permettant



d'utiliser des traitements de type hors ligne sont disponibles sans coût de traitement supplémentaire.

L'étape ultérieure consiste à combiner et à arranger ces primitives afin de dégager des caractéristiques directement exploitable par notre système de reconnaissance. Enfin, il est utile de remarquer que ce module de polygonalisation peut être utilisé dans d'autres domaines que celui de la reconnaissance optique de texte manuscrit en ligne. En particulier une application liée à l'interprétation autour du cadastre français [8] a utilisé ces algorithmes. Les résultats sont tout à fait convaincants prouvant la fiabilité du module.

BIBLIOGRAPHIE

- [1] ...C. Higgins, M. Ford "stylus driven interfaces - the electronic paper concept." *First International Conference on Document Analysis and Recognition*, Saint-Malo, octobre 1991.
- [2] ...M. Berthod *On-line Analysis of Cursive Writing*. In C.Y. Suen and R. De Mori Ed. "Computer Analysis and Perception", Vol 1: Visual Signals, 55-81, Boca Raton CRC Press, 1982.
- [3] ...S. Bercu, B. Delyon et G. Lorette "Segmentation pour une méthode de reconnaissance d'écriture cursive en ligne", *Bigre n°80 - CNED'92*, Nancy, juillet 1992.
- [4] ...A. Belaid, G. Masini "Segmentation de tracés sur tablette graphics en vue de leur reconnaissance" *Revue Technique et Science Informatiques*, Vol 1, n°2, 1982.
- [5] ...M.R. Ito, T.L. Chin, "On-line computer recognition of proposed standart ANSI (USASI) handprinted characters", *Pattern Recognition*, Vol 10, N°5/6, 1978.
- [6] ...C.C. Tappert, C.Y. Suen, T. Wakahara "On-line Handwriting Recognition - A survey. Proc. of the 9th Int. Conf. on Pattern Recognition 2, Rome Nov 1988.
- [7] ...R. Mullo, "Segmentation d'images et extraction de primitives pour la reconnaissance optique de texte", thèse de doctorat de l'université de Rouen.
- [8] ...J.M. Ogier, R. Mullo, J. Labiche, Y. Lecourtier, "French map interpretation: a sound approach" accepté à *IEEE/SMC'93 conférence*, 17-20 Octobre 1993. Letouquet, France

ANNEXE



tracé original

tracé comprimé



tracé original

tracé comprimé