Analyse de fontes anciennes : de la génération de données synthétiques à la reconnaissance

Nicholas Journet, Anne Vialard, Jean-Philippe Domenger

Analyse de fontes anciennes : de la génération de données synthétiques à la reconnaissance

Nicholas Journet — Anne Vialard — Jean-Philippe Domenger

Laboratoire Bordelais de Recherche en Informatique
Unité Mixte de Recherche CNRS (UMR 5800)
351, cours de la Libération
F-33405 Talence cedex
{journet,vialard,domenger}@labri.fr

RÉSUMÉ. *Cet article présente la première version d'un logiciel d'analyse et de reconnaissance de fontes anciennes. La principale originalité de ce travail est l'intégration d'un générateur d'images synthétiques de textes anciens dans la chaîne d'analyse. Ces images sont générées selon des critères spécifiés par l'utilisateur en termes de fontes utilisées, de nombre de lignes par image et de type de fond sur lequel sont intégrés les caractères. Au travers de plusieurs expérimentations, nous montrons que la génération d'images de documents anciens, en mettant à disposition une base d'images conséquente et adaptable, permet de tester avec précision la qualité de la chaîne d'analyse de fontes anciennes.*

ABSTRACT. *This paper presents the first version of a software dedicated to the analysis and recognition of old fonts. The first part of the analysis process, and the main contribution of this work, consists in a module for generating synthetic images of old documents. The images are generated according to criteria which are specified by the user (font, number of lines per image, type of background). We show that the generation of old documents allows to test precisely the quality of the old font recognition algorithms.*

MOTS-CLÉS : *Analyse d'images de documents anciens, Reconnaissance de fontes, Génération de vérité terrain*

KEYWORDS: *Old document images analysis, Optical font recognition, Ground truth generation*

1. Introduction

Dans le cadre d'un projet lié à l'indexation de documents patrimoniaux, nos travaux de recherche se sont focalisés sur la reconnaissance de fontes utilisées dans les images de documents anciens. A ce sujet, la littérature scientifique propose de nombreuses approches permettant de caractériser finement la très grande diversité des fontes contemporaines. Cependant, nous ne savons pas à ce jour si ce qui a été proposé pour l'analyse de documents contemporains est directement transposable à un corpus d'images de documents anciens.

Notre ambition est de proposer un logiciel, construit sous forme de briques logicielles, qui permette de caractériser et d'évaluer la qualité d'algorithmes de reconnaissance de fontes anciennes. La principale originalité de ce travail est l'intégration d'une phase de génération de documents anciens dans la chaîne d'analyse. Cette première phase permet de générer des images de documents comportant des caractéristiques précises (*eg :* type de fonte, fond bruité ou non, taille des caractères, ...). Il est ainsi possible d'évaluer la qualité de la chaîne d'analyse selon divers contextes.

Après un état de l'art des travaux relatifs à l'analyse et la reconnaissance de fontes contemporaines, cet article décrit la première version de ce logiciel dédié à l'analyse de fontes anciennes. Nous présentons tout d'abord le module de génération d'images de documents anciens (version image et XML). Il permet de constituer une base de vérité terrain à la fois conséquente et représentative des documents dont sont issues les fontes. La suite de l'article décrit les premières caractéristiques embarquées dans le logiciel. Elles regroupent une partie des caractéristiques citées dans l'état de l'art ainsi que de nouveaux descripteurs basés sur des estimateurs géométriques discrets. Enfin nous expliquons comment, sur la base de la vérité terrain générée et des algorithmes d'extraction de caractéristiques, un module de classification permet d'évaluer la qualité de la reconnaissance des fontes anciennes.

2. Reconnaissance de fontes contemporaines

L'objectif affiché des méthodes de reconnaissance de fontes (Optical Font Recognition) est avant tout de faciliter l'étape de reconnaissance de caractères. Elles permettent également de produire du texte rééditable à partir de documents papiers. Nous présentons ici un ensemble représentatif de ces méthodes dédiées, pour la majorité d'entre elles, à l'analyse de fontes contemporaines. Nous ne nous intéressons pas aux méthodes qui permettent d'identifier la langue du document ou le script du document (roman, chinois, arabe, ...). En effet, ces dernières tendent à être les plus indépendantes possibles de la fonte utilisée.

Les méthodes présentées sont regroupées en trois catégories, suivant le type de caractéristiques extraites du document avant l'étape de classification. Dans la première catégorie de méthodes, l'analyse du document s'appuie sur des connaissances typographiques comme la structure des caractères (corps /jambages) ou les empattements spécifiques à une fonte donnée. On peut distinguer les caractéristiques ty-

pographiques globales, calculées sur l'image d'une ligne ou sur l'image d'un mot, et les caractéristiques typographiques locales, calculées sur des composantes connexes correspondants aux caractères ou à des parties de caractères. L'idée commune aux méthodes de la deuxième catégorie est de considèrer un bloc de texte d'une fonte donnée comme une image de texture de façon à extraire des descripteurs de texture. Enfin les autres approches, parfois spécifiques à une langue, sont regroupées dans la troisième catégorie.

2.1. *Caractéristiques typographiques*

L'article de Zramdini et Ingold (Zramdini *et al.,* 1993) propose 5 descripteurs permettant de différencier la police, la graisse, la pente et la taille des fontes analysées. A partir de l'image binaire d'une ligne de texte sont calculés un profil vertical (nombre de pixels noirs de chaque ligne) et un profil horizontal (nombre de pixels noirs de chaque colonne). Le profil vertical permet d'obtenir la hauteur des différentes parties des caractères (corps, ascendant, descendant). Le calcul de la densité des pixels noirs à partir du profil horizontal donne une indication sur la graisse de la fonte. Enfin la variance de la dérivée du profil horizontal renseigne sur l'inclinaison du texte : elle est moins élevée pour du texte en italique. La classification est ensuite réalisée par un classifieur bayésien. La base d'apprentissage utilisée comporte 112 fontes différentes. Pour chaque fonte, 100 lignes de texte anglais de 6 cm ont été générées à partir de documents existants imprimés puis scannés à 400 dpi. Les résultats annoncés sont très satisfaisants : en particulier le taux de reconnaissance global de la famille de fonte est de plus de 90%. Notons que les images de document sont prétraitées : en plus d'un éventuel filtrage du bruit, les espaces variables entre mots sont remplacés par un espace fixe.

Eynard et Emptoz (Eynard *et al.,* 2009) ont proposé une méthode de reconnaissance du style d'un mot extrait d'un document ancien binarisé. Plus précisément, il s'agit de décider si le mot est écrit en style italique ou en style roman. Leur analyse s'appuie sur le profil horizontal de l'image du mot. Plusieurs indicateurs sont combinés pour prendre la décision finale : comparaison entre la valeur maximum du profil du mot initial et du mot incliné, taille des espaces entre les lettres,variation de la pente du profil. Les résultats sont très bons, proches de 100% pour les mots de plus de 3 lettres extraits d'un document du 18ème siècle.

Les auteurs de (Jung *et al.,* 1999) analysent le profil vertical d'une image de mot pour obtenir les hauteurs de l'ascendant et du descendant ainsi que la taille de la fonte. Ils effectuent également une analyse locale des extrémités de caractères contenant les empattements, ceci après suppression de la partie centrale des caractères. La classification, réalisée par un réseau de neurones, a été testée sur 2520 images de mots scannées à 200dpi, comportant 42 fontes différentes (6 polices croisées avec 7 tailles). Le taux de reconnaissance global annoncé est de 95% avec de moins bons résultats pour les petites tailles de caractère.

2.2. *Descripteurs de texture*

Pour pouvoir considérer l'image de document comme une image de texture, il faut prétraiter celle-ci de façon à créer une image de texture homogène comportant un espacement constant entre les mots et entre les lignes.

Dans (Zhu *et al.,* 2001), les auteurs présentent une approche texture recherchant les fréquences spatiales et les orientations d'une image de document afin d'en caractériser la fonte. Le prétraitement de l'image binaire initiale consiste tout d'abord à normaliser la hauteur des lignes et l'espacement entre deux caractères, la hauteur d'une ligne de texte et la distance entre deux caractères étant mesurées sur les profils verticaux et horizontaux de l'image. Ce prétraitement est finalisé par le remplissage de toutes les parties restées vides avec des morceaux de texte normalisés choisis aléatoirement. Dans un deuxième temps, l'image de texture est analysée par un banc de 16 filtres de Gabor (4 orientations combinées à 4 fréquences). Le vecteur caractéristique est composé de la moyenne et l'écart type de chacune des 16 réponses. Finalement, la classification se fait simplement par un calcul de distance euclidienne pondérée entre le vecteur caractéristique calculé et ceux de la base d'apprentissage. Cette technique permet de discriminer plus d'une cinquantaine de fontes anglaises et chinoises avec un taux de reconnaissance moyen de 99%. D'autres tests réalisés sur la base de données utilisée dans (Zramdini *et al.,* 1993) donnent des résultats plus nuancés : le taux de reconnaissance moyen de la police de caractère est de 83%, cette moindre efficacité s'expliquant par des confusions entre polices serif et sans-serif de la même famille.

Ma et Doermann (Ma *et al.,* 2005) ont proposé une variante plus performante de cette approche. La classification y est effectuée par un réseau de neurones et les filtres de Gabor sont définis de façon à favoriser les réponses à la répétition d'un trait d'une direction donnée.

On trouve une approche statistique de l'analyse de texture dans (Avilés-Cruz *et al.,* 2005). L'image initiale est prétraitée pour obtenir une image de texte uniforme par une méthode similaire à celle de (Zhu *et al.,* 2001). Sont alors calculés les moments d'ordre 3 et 4 de cette image puis le nombre des caractéristiques obtenues est réduit par une ACP. La classification est réalisée par un classifieur bayésien. En ce qui concerne les expérimentations, une image de test de 640x640 pixels est générée pour chacune des 32 fontes considérées (8 polices croisées avec 4 styles à taille fixe de 12pts). Sur chaque image 100 échantillons de taille 32x32 pixels sont prélevés à des positions aléatoires. La moitié des échantillons est utilisée comme base d'apprentissage, l'autre moitié pour les tests. Les taux de reconnaissance annoncés pour des images non bruitées et à une résolution de 300dpi sont de 100%.

Remarquons que dans les trois méthodes ci-dessus la normalisation des images ne permet pas de différencier plusieurs tailles de fontes. Il se trouve que les espacements entre caractères sont aussi caractéristiques de l'aspect visuel d'un texte. Le fait de normaliser supprime complètement cette information.

2.3. *Autres approches*

Dans la méthode de Khoubyrabi and Hull (Khoubyari *et al.,* 1996), on ne trouve pas d'étape d'extraction de caractéristiques proprement dite dans la mesure où la reconnaissance de fonte se fait par "matching". Ces auteurs proposent de construire un prototype pour tous les *mots fonctions* du texte analysé. Il s'agit des mots courts et fréquents comme *the*, *of* ou *a* en anglais. Pour cela tous les mots du texte sont regroupés en clusters et le prototype d'un mot fonction est calculé comme la moyenne des images de mots regroupées dans le cluster correspondant. La reconnaissance de fonte se fait finalement par calcul d'une distance entre les prototypes et une base de données d'images de mots fonctions. Les taux de reconnaissance obtenus sont de l'ordre de 85%.

Yang et al. (Yang *et al.,* 2006) ont proposé une méthode originale mais spécifique à l'analyse de documents en chinois. L'idée centrale de cet article est d'avoir mis évidence 5 tracés de base composant les caractères chinois (trait horizontal, trait vertical, trait en biais,...). Ces traits peuvent être représentés par des motifs élémentaires de 8 pixels chacun. On calcule alors la correspondance entre un motif et un bloc de texte comme étant la moyenne des niveaux de gris sous le motif, ceci pour N positions aléatoires du motif. On obtient ainsi N valeurs pour chaque motif de base. Les séries obtenues sont analysées en utilisant la transformée de Hilbert-Huang de façon à obtenir deux descripteurs pour chaque motif : l'énergie haute fréquence et l'énergie basse fréquence. Enfin la classification se fait simplement par distance euclidienne pondérée. Les tests présentés s'appuient sur une base d'apprentissage de 960 blocs de texte (40 blocs pour chacune des 24 fontes considérées). La reconnaissance de fonte se fait ici sur des blocs de texte en niveau de gris mais une normalisation du texte similaire à celle de (Zhu *et al.,* 2001) reste nécessaire. Le taux de reconnaissance moyen obtenu est de plus de 97%.

Les auteurs de (Lee *et al.,* 2003) proposent d'appliquer la *factorisation en matrices non-négatives* à la reconnaissance de fontes. Cette technique permet de décomposer une matrice de données (ici l'image de taille 28x28 pixels d'un caractère où les valeurs sont normalisées entre 0 et 1) en une combinaison de bases. Les coefficients de la combinaison sont utilisés comme caractéristiques de l'image. Enfin la classification se fait simplement en cherchant le plus proche voisin dans la base de données composée de 4992 échantillons correspondant à 48 fontes différentes. Les tests ont été effectués sur 190830 images de caractères pour un taux de reconnaissance moyen de plus de 99%.

Dans notre travail actuel, nous avons repris le calcul de caractéristiques typographiques globales basée sur le profil horizontal et vertical. Nous avons également proposé de nouvelles caractéristiques locales (voir partie 4). Notre travail pourra être complété par le calcul de descripteurs de texture.

3. Génération de vérité terrain

3.1. *Génération de vérité terrain à base de fontes contemporaines*

Comme tout problème de reconnaissance des formes, la reconnaissance de fontes nécessite la constitution d'une base de tests. En raison du grand nombre de travaux scientifiques existants dans ce domaine, on peut dénombrer un grand nombre de bases sur lesquelles ont été testées les différentes méthodes de reconnaissance de fontes existantes. Cette quantité de bases s'explique principalement par la facilité que présente la création automatique d'une base d'images de documents composée de texte multi-fontes. En effet, il suffit d'utiliser les nombreux fichiers True Type Fonts (`TTF`) disponibles et de générer des images contenant des phrases. Une simple extraction des caractères de ce fichier permet de générer une grande diversité d'images tests (cf figure 1).

Figure 1. *Images de fontes générées sur la base de fichiers `TTF`. Le grand nombre de fichiers `TTF` disponible permet de générer des mots dans des fontes visuellement très variées.*

3.2. *Génération de vérité terrain à base de fontes anciennes*

Dans le cadre de nos travaux sur la reconnaissance de fontes anciennes, la génération d'une vérité terrain pose un double problème. Tout d'abord, il n'existe pas (ou très peu) de fichiers `TTF` de fontes anciennes. Il n'est donc pas envisageable d'aborder le problème sous le même angle que dans le cadre de travaux sur des fontes contemporaines. Dans le même temps, il faut pouvoir créer des images reproduisant fidèlement les spécificités visuelles des documents anciens. Ainsi, la fonte n'est pas l'unique information importante. Il faut pouvoir reproduire la dégradation des caractères au fil des pages, les différentes dégradations du papier (taches, trous, ...) ainsi que les déformations dues à l'étape de numérisation.

Sur la base de cette double constatation, nous avons mis en place un logiciel dédié à la génération d'images contenant non seulement des fontes anciennes mais respectant également les spécificités visuelles des documents anciens. La figure 2 détaille le fonctionnement de notre proposition.

Le logiciel met à disposition d'un utilisateur une IHM permettant de saisir, à la souris, des exemples de caractères de la fonte qu'il souhaite créer (point 1 de la figure 2). Pour chaque caractère extrait, l'utilisateur indique son label. On associe ainsi un label à chaque forme. Lors de cette phase de saisie, une étude de la position de la composante connexe du caractère et des composantes connexes voisines permet de

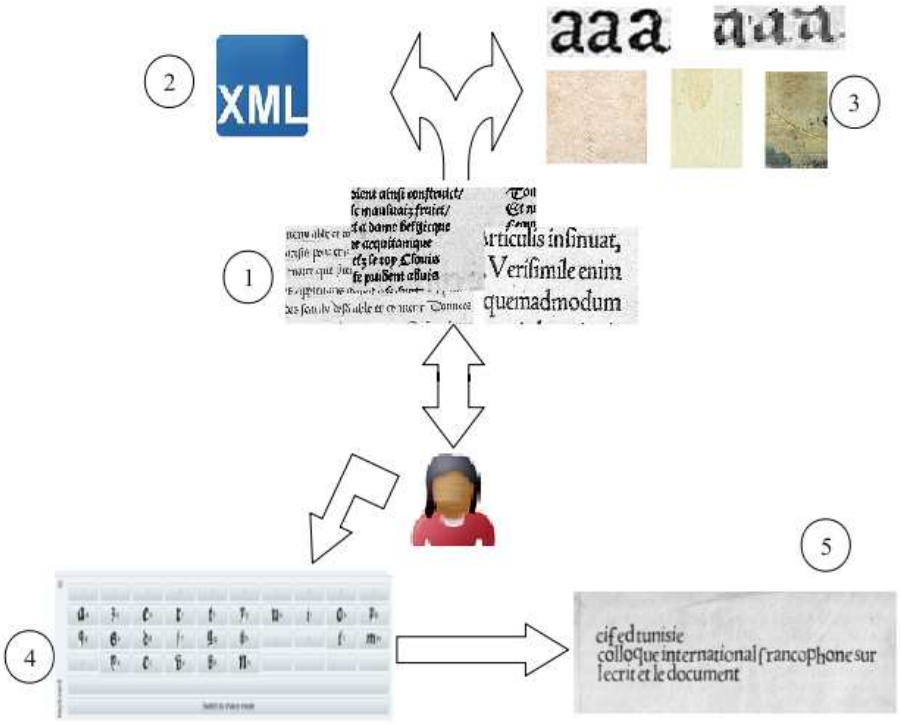

Figure 2. *Schéma récapitulant le fonctionnement du processus de génération de documents anciens. Une IHM permet de saisir à la souris des caractères et des fonds issus d'images de documents numérisés (points* 1,2,3*). L'utilisateur peut ensuite générer des documents synthétiques en jouant sur divers paramètres (points* 4 *et* 5*).*

déterminer les points d'accroches nécessaires au futur positionnement de chaque caractère (baseline, placement du prochain caractère, ...). Si les paramètres sont appris automatiquement, l'utilisateur peut néanmoins affiner ces différents réglages (cf figure 3). Sur le même modèle, l'utilisateur peut extraire divers types de fonds d'images de documents anciens. Cela lui permet, par exemple, d'extraire des fonds visuellement variés (fonds bruités, fonds très clairs ou très sombres, ...)

A la fin de l'étape 1 l'utilisateur a donc saisi pour chaque fonte ancienne qu'il souhaite créer plusieurs exemples de chaque lettre. Il a également extrait plusieurs types de fonds.

Après cette phase de saisie, l'ensemble des informations extraites est répertorié dans un fichier `XML`. On y retrouve, entre autres, le nom de la fonte et les caractères extraits auxquels sont associées des informations de position (points d'accroche). On

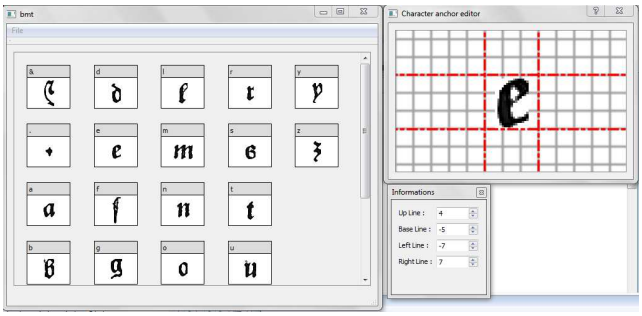

Figure 3. *Capture d'écran de l'éditeur de fontes anciennes. Sur la partie de gauche sont situés les caractères déjà appris par l'utilisateur. Sur la partie de droite se trouve la fenêtre s'ouvrant lors de la saisie d'un exemple de caractère. Les traits rouges symbolisent les* 4 *paramètres calculés automatiquement. Ces paramètres peuvent être modifiés en déplaçant les traits rouges ou en indiquant directement leurs valeurs dans les champs situés au dessous.*

trouve également dans ce fichier `XML` des informations sur les espaces inter-lignes, inter-mots et inter-caractères de la fonte extraite (point 2 de la figure 2).

Cette phase de saisie aboutit également à la création d'un ensemble d'images ; une image par caractère et une image par fond (point 3 de la figure 2).

Le point 4 de la figure 2 illustre quant à lui, la phase de génération d'un document ancien. Au travers d'une interface graphique, l'utilisateur renseigne plusieurs paramètres selon le rendu visuel qu'il souhaite obtenir :

1) Le texte qui s'affichera dans l'image finale. Ce texte peut être saisi sur un clavier virtuel (cf figure 4) ou directement être issu d'un fichier texte.

2) La fonte ancienne à utiliser.

3) Le type de fond sur lequel seront intégrés les caractères.

4) La taille des caractères (re-dimensionnement selon le facteur d'échelle indiqué).

5) La taille de la page.

Lors de la phase de création d'une police, l'utilisateur a saisi plusieurs exemplaires de chaque lettre. Lors de la génération de l'image finale, chaque caractère est choisi aléatoirement parmi l'ensemble des caractères extraits en phase 1. Ce choix permet d'obtenir un résultat final plus probant avec des caractères qui ne sont pas tous identiques entre eux. La figure 4 illustre la variété de lettres "e" pouvant exister dans une police donnée.

La figure 5 illustre le résultat de la génération d'un texte selon la fonte utilisée ou bien encore selon le fond choisi. Ces images ont été générées sans utiliser l'interface utilisateur. Le résultat de la figure 5 est donc obtenu entièrement automatiquement. Si

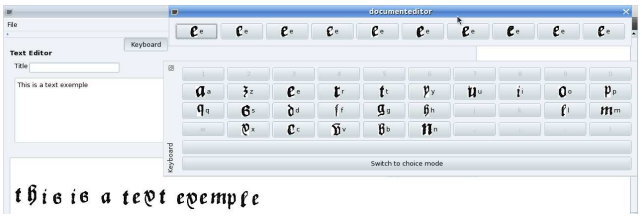

Figure 4. *Capture d'écran du clavier virtuel mis à disposition de l'utilisateur. Celui-ci peut choisir la fonte du texte. Un écran lui permet de prévisualiser la disposition des caractères en fonction des paramètres de position. L'utilisateur a la possibilité de choisir entre plusieurs exemplaires d'un même caractère (ici le dernier "e" de "exemple").*

le visuel est correct, on peut néanmoins remarquer que les caractères "q", et "h" ont mal été positionnés. Il faudrait donc qu'un utilisateur règle le positionnement qui a été initialement calculé.

Figure 5. *Images de documents anciens générés automatiquement. Les deux images du haut ont été générées avec deux fontes différentes. L'image du bas a été générée avec un fond bruité.*

4. Extraction de caractéristiques

Afin de valider l'intérêt d'une phase de génération de documents dans une chaîne d'analyse de fonte, nous avons déjà intégré plusieurs algorithmes d'extraction de caractéristiques dans notre logiciel. Le deuxième module du logiciel que nous présentons dans cette partie gère le processus d'extraction de caractéristiques sur les images

générées par le premier module, puis binarisées. Après analyse, chaque image est décrite par un vecteur de réels issus de caractéristiques typographiques calculées sur chaque ligne de caractères et de caractéristiques locales calculées sur chaque caractère.

Nous avons tout d'abord intégré au logiciel l'ensemble des caractéristiques typographiques présentées dans (Zramdini *et al.,* 1993). Ces cinq caractéristiques sont calculées à partir des profils verticaux et horizontaux de l'image binarisée d'une ligne de texte. Il s'agit de :

– la hauteur de ligne,

– la hauteur de la partie supérieure des caractères (sans compter le descendant),

– la hauteur de la partie centrale des caractères (sans compter ni ascendant ni descendant),

– la densité des pixels noirs calculée sur la partie centrale de la ligne,

– la variance de la dérivée du profil horizontal.

Nous avons également intégré des caractéristiques plus locales calculées sur la boîte englobante de chaque caractère composant l'image :

– la densité des pixel noirs,

– l'aire de la boîte englobante,

– l'orientation de l'axe principal, c'est-à-dire l'orientation du premier vecteur propre de la matrice de covariance des pixels contenus dans la boîte englobante du caractère.

Les images de texte testées comportent toujours plusieurs caractères et très souvent plusieurs lignes. Les algorithmes d'analyse que nous venons de présenter fournissent une valeur par ligne pour les premiers et une valeur par caractère pour les seconds. C'est la valeur médiane qui est finalement sélectionnée pour caractériser l'image.

Les dernières caractéristiques intégrées proviennent de l'analyse des composantes connexes correspondant aux caractères binarisés. Nous utilisons ici des outils de géométrie discrète : estimateurs géométriques discrets et image de distance. On pourra consulter (Coeurjolly *et al.,* 2007) pour plus de détails à ce sujet. Les caractéristiques actuellement implémentées sont :

– le périmètre du contour extérieur de chaque caractère. L'estimateur de longueur utilisé est l'estimateur de Rosen-Profitt.

– l'aire en nombre de pixels de la composante connexe.

– l'orientation du contour.

En chaque point d'un contour discret 4-connexe, on peut définir la tangente comme étant un segment de droite discrète, le plus long possible, confondu avec le contour. On obtient ainsi une orientation de la tangente en chaque point de contour. La caractéristique retenue pour un caractère est l'orientation la plus présente. Elle est liée au style (normal, italique) de la fonte.

– le nombre de points dominants.
La courbure en chaque point de contour est calculée comme la dérivée de l'orientation de la tangente. Par seuillage, nous déterminons les points de forte courbure. Ces points correspondent aux points dominants du contour, c'est-à-dire aux points représentatifs de la forme. La caractéristique retenue est le nombre de points dominants du contour de chaque caractère.

– la distance au fond.
Les transformées en distance permettent d'associer à chaque point d'une région binaire sa distance au fond. La caractéristique retenue ici est la valeur maximum de distance au fond pour chaque caractère. Cette distance renseigne sur la graisse de la fonte.

Les caractéristiques issues de l'analyse du contour d'une composante connexe sont illustrées dans la figure 6. Au final, on ajoute au vecteur caractéristique de l'image la médiane des caractéristiques obtenues pour chaque caractère.

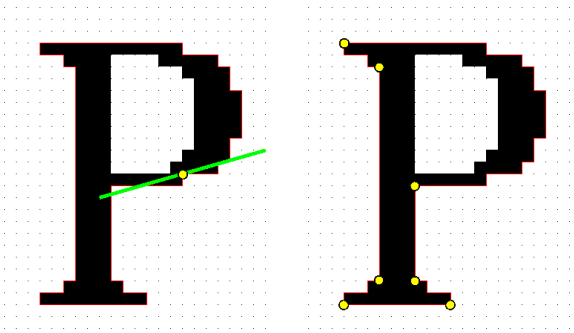

Figure 6. *Calcul de caractéristiques locales : on analyse le contour inter-pixels extérieur du caractère. Sur la partie gauche de la figure est représentée la tangente en un point de contour, sur la partie droite apparaissent les points dominants du contour.*

5. Evaluation

Le troisième et dernier module du logiciel que nous développons actuellement génère automatiquement de nombreux graphiques. Ces derniers permettent d'interpréter les résultats d'une étape de classification appliquée aux données générées par le premier module.

Lors de l'étape de génération des données et de l'étape de calcul des caractéristiques, il est possible de jouer sur plusieurs paramètres (taille de la base, type de fonds, tailles de caractères, nombre de caractéristiques extraites, ...). Cette multitude de cas de figure permet de tester très facilement de nombreuses hypothèses lors de l'étape d'évaluation. Cette section décrit un exemple de test que nous pouvons réaliser avec

cette première version de notre logiciel. Dans un premier temps nous montrons que les caractéristiques présentées dans la section précédente permettent effectivement de reconnaître des fontes contemporaines. Ensuite, au travers de deux tests réalisés sur des fontes anciennes, nous montrons que ce dernier module permet d'analyser rapidement la qualité des descripteurs.

5.1. *Protocole de test*

Suite à la phase d'extraction des caractéristiques, chaque caractère est décrit par un vecteur de réels. Nous avons choisi d'utiliser une méthode de classification avec phase d'apprentissage basée sur les SVM. Pour chaque test, la méthodologie suivante est appliquée :

1) Séparation de la base en deux. Pour chaque ensemble de fontes, un sous-ensemble d'exemples est tiré aléatoirement. Ce sous-ensemble est destiné à alimenter la phase d'apprentissage. L'autre est destiné à la phase de reconnaissance

2) Construction des vecteurs supports à l'aide de la librairie LIBSVM[1]. Etant donné que nous sommes dans un cas de classification à plusieurs classes, l'approche utilisée est "un contre un" avec un radial-kernel et un gamma de 0.052.

3) Reconnaissance sur la base n'ayant pas servi lors de l'apprentissage

Etant donné que les éléments inclus dans la base d'apprentissage sont tirés aléatoirement, il est nécessaire de répéter plusieurs fois ce protocole afin de mesurer l'importance des exemples choisis. Ainsi, pour chaque taille de base d'apprentissage le protocole décrit précédemment est répété 50 fois. Pour chaque taille de base d'apprentissage, nous sommes donc en mesure de fournir un ensemble de statistiques (min, max, moyenne, écart type,...). Ce choix permet une analyse plus fine des résultats.

5.2. *Classification de fontes contemporaines*

Dans un premier temps, nous avons expérimenté notre logiciel de reconnaissance de fontes sur une base d'images de textes contemporains.

Nous avons tout d'abord sélectionné 121 fontes différentes. Pour chacune d'entre elles nous avons choisi 41 phrases différentes. La base de fontes contemporaines que nous avons testée est donc composée de 4840 images.

La figure 7 illustre les résultats obtenus. Ils sont similaires à ceux obtenus par (Zramdini *et al.,* 1993) ou (Zhu *et al.,* 2001) avec des taux de reconnaissances proches de 100%. On observe par exemple que, pour une taille de base d'apprentissage correspondant à 30% de la taille de la base d'origine, le plus mauvais des 50 taux de reconnaissance a été de 60% et le meilleur de 100%. Le taux moyen de reconnaissance est de 93% avec un écart type d'environ 15 points. On observe également qu'il

1. http ://www.csie.ntu.edu.tw/ cjlin/libsvm/

suffit d'une taille de base d'apprentissage d'environ 50% de la base complète pour arriver à des taux de reconnaissance d'environ 100%.

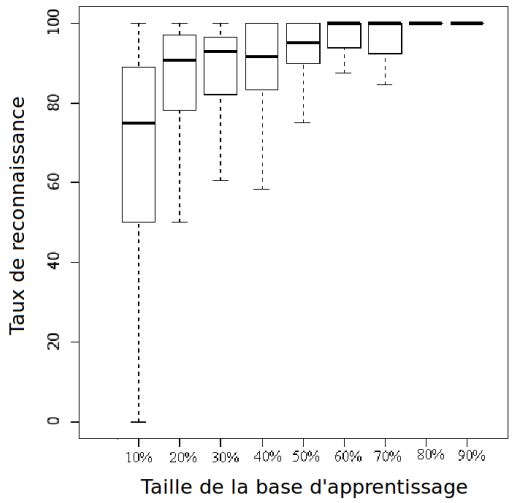

Figure 7. *Résultats obtenus sur la base de fontes contemporaines. Cette base est composée de* 4840 *images issues de* 121 *fontes différentes. Pour chaque taille de base d'apprentissage, ce graphique indique le minimum,le maximum, la moyenne et l'écart-type des taux de classification obtenus sur plusieurs jeux d'essais.*

5.3. *Classification de fontes anciennes*

Dans un premier temps, nous souhaitions savoir s'il était possible de transposer "directement" aux fontes anciennes, les algorithmes utilisés pour la reconnaissance de fontes contemporaines.

La principale différence que l'on peut observer entre les fontes anciennes et les fontes contemporaines se situe au niveau de la dégradation de l'image à analyser. La spécificité de l'analyse de fontes anciennes tient plus, selon nous, à la mise en place d'un protocole d'analyse et de reconnaissance permettant le traitement d'images fortement bruitées. Ainsi, les caractères peuvent être dégradés et une même lettre peut être présente sous différentes formes (caractères séparés en plusieurs morceaux, trous dans les lettres, lettres attachées entre elles,...). De même, la présence de l'encre du recto sur le verso ou encore la présence de taches près des caractères complique l'analyse.

Pour réaliser ces tests, nous avons donc généré un grand nombre d'images de textes issus de 12 fontes anciennes différentes. Cette génération a permis d'obtenir 7820 zones de textes composées chacune d'une unique fonte ancienne. Ces images générées l'ont été avec des fonds de qualités diverses, avec un nombre de lignes pouvant varier

de 1 à 15, ... Sur cette base, nous avons appliqué strictement le même protocole de test utilisé pour les fontes contemporaines. La figure 8 permet de conclure qu'une "simple" transposition des algorithmes d'analyse de fontes contemporaines ne permet pas de différencier correctement 7820 images réparties sur 12 fontes différentes. En effet, dans le meilleur des cas (90% de la base en apprentissage et 10% pour la reconnaissance) le taux de reconnaissance ne dépasse que de très peu les 60%.

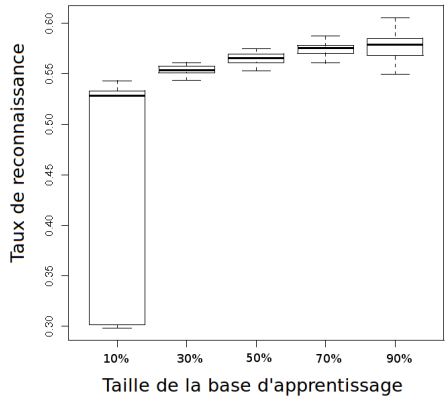

Figure 8. *Résultats obtenus sur la base de fontes anciennes. Cette base est composée de 7820 images générées avec différentes qualités de fonds et différentes polices anciennes. Le meilleur taux de reconnaissance ne dépasse que de très peu les* 60%.

Une deuxième expérimentation montre tout l'intérêt de générer, "à la carte" des images de documents anciens.
Nous avons retiré des 7820 images toutes les images contenant moins de 3 lignes. La figure 9 résume les tests réalisés sur les 6065 images restantes. Le graphique met en évidence un taux de reconnaissance qui s'est amélioré. En effet il s'approche désormais des 80%. Le fait d'avoir pu modifier très simplement la base d'image testées permet de conclure rapidement que les caractéristiques actuellement intégrées dans notre logiciel ne sont pas robustes aux blocs de texte contenant trop peu de caractères. Il faudrait certainement intégrer des descripteurs plus globaux (texture par exemple).

6. Conclusion et perspectives

Cet article présente la première version d'un logiciel dont l'ambition est de montrer qu'il est intéressant d'intégrer une phase de génération de données synthétiques dans une chaîne d'analyse et de reconnaissance de fontes anciennes.

Après une étape interactive où un utilisateur saisit plusieurs exemplaires de caractères d'une même fonte et référence plusieurs types de fonds d'images, nous pouvons générer des images de documents anciens visuellement proches des originales.

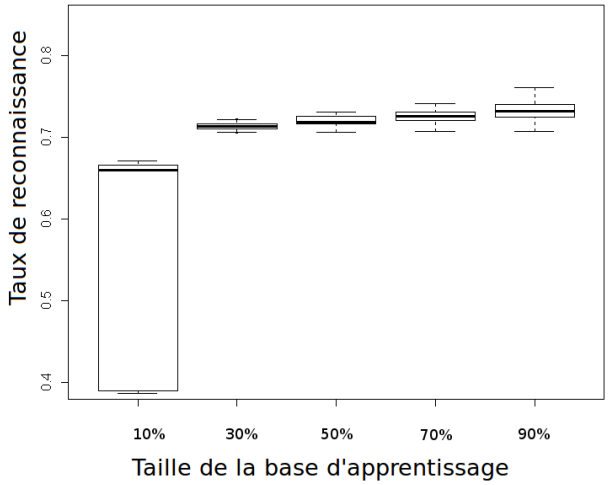

Figure 9. *Résultats obtenus sur une base de fontes anciennes avec des images contenant au moins 3 lignes de texte. Cette base est consituée de* 6065 *images. Le meilleur taux de reconnaissance atteint presque* 80%.

L'ensemble de ces images est ensuite, de manière classique, analysé par des programmes calculant des caractéristiques diverses (analyse des orientations des pixels, de caractéristiques géométriques locales aux caractères, ...) . Un dernier module d'évaluation permet de générer des graphiques illustrant la pertinence et la robustesse des algorithmes d'analyse de fontes sur les images générées.

A l'instar de ce que proposent les auteurs de (Strecker *et al.,* 2009), nous souhaitons, dans la prochaine version du générateur de documents anciens, permettre à l'utilisateur de choisir la mise en page qu'il souhaite obtenir. Cette option permettrait d'obtenir une plus grande variété d'images de documents anciens.

Toujours pour améliorer la qualité visuelle des images générées, nous travaillons actuellement sur une option permettant à l'utilisateur de choisir si la page générée contiendra des défauts supplémentaires (présences de taches, de trous, de pliures, ...). Une autre perspective relative à la phase d'apprentissage est de faire évoluer notre prototype vers un outil semi-automatique dans lequel le système rechercherait lui-même à partir d'un exemple, les caractères ressemblants dans le document modèle. Cela permettrait de diversifier les caractères et de simuler au moins partiellement les dégradations au fil des pages.

Nous souhaitons utiliser les résultats fournis par le module d'évaluation afin d'évaluer les caractéristiques actuelles et de déterminer quelles caractéristiques devraient être intégrées à la prochaine version du logiciel (texture par exemple). Pour cela il faudra tester la robustesse des descripteurs dans divers contextes (nombre de caractères dans un bloc de texte, quantité de bruit présent, déformation des caractères, ...).

Remerciements

Les auteurs souhaitent remercier Julian Roigt pour l'aide apportée tout au long de la réalisation de ce travail.

7. Bibliographie

Avilés-Cruz C., Rangel-Kuoppa R., Reyes-Ayala M., Andrade-Gonzalez A., Escarela-Perez R., « High-order statistical texture analysis : font recognition applied », *Pattern Recogn. Lett.*, vol. 26, n° 2, p. 135-145, 2005.

Coeurjolly D., Montanvert A., Chassery J.-M., *Géométrie discrète et images numériques*, Hermès, 2007.

Eynard L., Emptoz H., « Italic or Roman : Word Style Recognition Without A Priori Knowledge for Old Printed Documents », *ICDAR 2009*, p. 823-827, 2009.

Jung M., Shin Y., Srihari S., « Multifont Classification Using Typographical Attributes », *ICDAR '99 : Proceedings of the Fifth International Conference on Document Analysis and Recognition*, IEEE Computer Society, p. 353, 1999.

Khoubyari S., Hull J. J., « Font and function word identification in document recognition », *Comput. Vis. Image Underst.*, vol. 63, n° 1, p. 66-74, 1996.

Lee C., Kang H., Jung K., Kim H., « Font Classification Using NMF », *CAIP03*, p. 470-477, 2003.

Ma H., Doermann D. S., « Font identification using the grating cell texture operator », *DRR*, p. 148-156, 2005.

Strecker T., Beusekom J. v., Albayrak S., Breuel T. M., « Automated Ground Truth Data Generation for Newspaper Document Images », *ICDAR '09 : Proceedings of the 2009 10th International Conference on Document Analysis and Recognition*, IEEE Computer Society, Washington, DC, USA, p. 1275-1279, 2009.

Yang Z., Yang L., Qi D., Suen C. Y., « An EMD-based recognition method for Chinese fonts and styles », *Pattern Recogn. Lett.*, vol. 27, n° 14, p. 1692-1701, 2006.

Zhu Y., Tan T., Wang Y., « Font Recognition Based on Global Texture Analysis », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, n° 10, p. 1192-1200, 2001.

Zramdini A. W., Ingold R., « Optical Font Recognition from Projection Profiles », *Electronic Publishing*, vol. 6, n° 3, p. 249-260, 1993.