Utilisation des points d'intérêt pour rechercher des mots imprimés ou manuscrits dans des documents anciens

Jean Camillerapp

Université Européenne de Bretagne, France
INSA, IRISA, UMR6074
Campus de Beaulieu
F-35042 Rennes cedex

jean.camillerapp@irisa.fr

RÉSUMÉ. *Dans certains types de documents anciens, lorsque le vocabulaire n'est pas trop important, une reconnaissance des mots à partir d'indices visuels peut s'avérer une solution plus efficace que la reconnaissance de chaque lettre. Cette approche est généralement appelée word spotting. Dans cet article nous proposons une nouvelle forme pour cette approche adaptée au repérage des en-têtes imprimés de champs dans les formulaires, à la détection de filet en pointillés ou à l'aide à la transcription de noms propres manuscrits. Elle utilise une représentation dense des images par des descripteurs locaux et une représentation parcimonieuse pour les modèles. L'article donne des évaluations des performances.*

ABSTRACT. *In certain document types, when the vocabulary is not too important, a recognition of the words by visual index can be a solution more effective than using an analytical approach based on the recognition of each letter. This approach is known as word spotting. In this paper we propose a new form for this approach adapted to the identifying headers printed form fields, detection of dashed lines, or transcription of handwritten proper nouns. We suggest to use a dense description by local descriptors for images and a parse for models. This article gives performances evaluations.*

MOTS-CLÉS : *document ancien, word spotting, point d'intérêt, SIFT, transcription assistée.*

KEYWORDS: *ancient document, word spotting, interest point, SIFT, assisted transcription.*

1. Introduction

Dans le traitement de documents, on peut chercher à reconnaître la structure, c'est le cas pour le traitement des partitions musicales, des plans architecturaux ou des schémas électriques. On peut aussi s'intéresser à la lecture des textes imprimés ou manuscrits qu'ils contiennent. Dans les formulaires ou dans les schémas annotés, on trouve un mélange de ces deux aspects.

Pour la lecture du contenu, deux grandes familles d'approche sont envisageables. L'une établit une segmentation en caractères et en réalise la reconnaissance (OCR), l'autre recherche des indices visuels dans l'image et utilise des comparaisons à partir de ces indices (*word spotting*). La première approche est surtout utilisée pour des documents de bonne qualité. La seconde est envisageable lorsque le vocabulaire est restreint.

Rath et Manmatha (Rath *et al.,* 2007) ont été les pionniers du *word spotting*. Ils utilisent les enveloppes supérieures et inférieures du tracé manuscrit comme indices visuels.

D'autre part il existe des travaux sur la localisation d'objets dans les images naturelles, qui s'appuient sur des indices visuels constitués par de petites zones de l'image, dans lesquelles la répartition de la luminosité est très hétérogène. Ces petites zones sont appelées coins, points clés ou point d'intérêt (le terme que nous utiliserons) selon les auteurs. On commence à trouver des travaux utilisant les points d'intérêt dans le traitement d'images de documents. La méthode présentée dans cet article utilise les points d'intérêt comme indice visuel.

Dans la section suivante, nous présentons certains travaux de la littérature. La section 3 décrit les points principaux de la méthode que nous proposons et les choix que nous avons faits pour tenir compte des particularités des images de documents. La section 4 donnera des exemples d'application dans des domaines différents ; elle fournira également des évaluations de performances.

2. Travaux dans le domaine

Les premiers travaux sur les comparaisons d'images à partir des points d'intérêt ont porté sur des images naturelles, mais certains résultats peuvent être repris pour des images de documents.

Schmid (Schmid *et al.,* 2000) compare six détecteurs de points d'intérêt dont celui de Harris (Harris *et al.,* 1988). L'article (Mikolajczyk *et al.,* 2005b) en compare six autres plus spécialement adaptés aux déformations d'images d'objets 3D. Seul le détecteur de Harris semble actuellement utilisé dans les images de documents.

Une fois les points d'intérêt détectés, il faut représenter les variations de la luminosité dans un petit voisinage, c'est le rôle du descripteur local. L'article (Mikolajczyk *et al.,* 2005a) en évalue plusieurs, là encore sur des images naturelles.

Le descripteur SIFT, *Scale Invariant Feature Transform*, proposé par (Lowe, 2004) et quelques unes de ses variantes sont souvent utilisés dans le traitement d'images de documents. Bay propose dans (Bay *et al.,* 2008) un autre descripteur donnant de bons résultats et qui serait plus rapide à calculer : SURF (*Speeded Up Robust Features*).

Les deux articles (Lowe, 2004) et (Bay *et al.,* 2008) proposent une description complète d'une chaîne qui réalise la localisation d'objets définis par un ensemble de points d'intérêt.

Leydier (Leydier *et al.,* 2009) propose de rechercher des mots définis par une requête textuelle, mais utilisant des indices visuels. Dans l'introduction et dans la section 2 il donne une bonne description des problèmes qui se posent autour du *word spotting* et un classement intéressant des différentes approches actuellement utilisées.

(Rodriguez *et al.,* 2008) présente un projet particulièrement ambitieux ; il s'agit de classer le courrier manuscrit à partir d'une dizaine de mots clés. Dans ce contexte il y a une très grande variabilité de l'aspect visuel d'un mot. Pour arriver à résoudre ce problème il utilise plusieurs centaines d'exemples de chacun des mots. La comparaison entre l'image et les modèles se fait au moyen de HMM.

(Rusinol *et al.,* 2011) propose une méthode d'exploration de pages de documents sans segmentation préalable. Il s'appuie pour cela sur une représentation de chaque page par un ensemble de SIFT calculés sur une grille 5x5. Les descripteurs sont requantifiés au moyen d'un dictionnaire, et les mots sont retrouvés grâce à un découpage systématique en fonction de la taille approximative des mots dans la collection.

Song (Song *et al.,* 2011) propose d'utiliser les SIFT pour caractériser des parties de caractères manuscrits.

3. Description générale de la méthode

Le travail exposé ici a pour objectif de construire et d'évaluer une nouvelle approche du *word spotting* à base de points d'intérêt. Cette construction devrait permettre par la suite de tester différentes options. Nous allons examiner les points suivants :

– la détection des points d'intérêt dans une image,

– le choix du descripteur local,

– la localisation d'un mot dans une image.

Dans les images de documents que nous avons à traiter, l'écriture est approximativement de la même taille dans une collection, il n'y a donc pas besoin d'assurer une grande invariance à l'échelle ; par contre les mots, surtout manuscrits, ne peuvent pas être considérés comme des objets solides, ils comportent de petites déformations. L'écriture est globalement horizontale, il peut y avoir une petite rotation du document au moment de la numérisation, mais elle n'est pas sensible à l'échelle d'un descripteur local. Comme les conditions de qualité d'encre, de conservation du papier ou de

numérisation ne sont pas stables, il est important d'assurer l'invariance au contraste des images.

3.1. *Définition des points d'intérêt*

L'objectif des points d'intérêt est de sélectionner sur des critères de variations locales de luminosité une fraction assez faible de points dans l'image. Cette sélection doit être stable, il faut choisir les mêmes points pour représenter un objet présent dans des images différentes. Ces points doivent être discriminants : dans une image il doit y avoir peu de points dont les descripteurs locaux se ressemblent.

Dans le cadre du *word spotting*, quelques essais nous ont montré que la détection proposée par Lowe au moyen des différences de gaussienne (DOG) ne conduisait pas à une stabilité suffisante. La figure 1 illustre cette affirmation. Il s'agit du même mot écrit dans trois cases consécutives d'un formulaire. Les points d'intérêts situés dans les régions blanches sont les plus instables ; au moment de la détection, il est facile de ne pas les prendre en compte. Les points dans le tracé sont assez stables, mais le bas du « h » ou la sélection dans le « a » montrent des différences importantes. Cette instabilité semble due à la finesse de l'écriture vis-à-vis de la méthode de détection qui s'appuie sur des différences du second ordre.

Figure 1. *Instabilité des points d'intérêts détectés par des* DOG

Ce problème de détection semble délicat dans les images de documents car dans (Rodriguez *et al.,* 2008) on prend un échantillon par pixel le long d'une ligne située dans le corps du mot tandis que dans (Rusinol *et al.,* 2011) on utilise un échantillonnage systématique par une grille 5x5.

Si, lors de la localisation d'un objet dans une image naturelle, il est légitime de ne pas vouloir retrouver dans l'image tous les points qui décrivent l'objet. Inversement, dans les images de documents l'absence d'une mise en correspondance peut être la marque d'une différence significative entre le modèle et l'image, car les différences entre les mots peuvent être faibles. Nous imposons donc que tous les points du modèle soient appariés, ce qui implique que la détection des points dans l'image soit très stable, quitte à avoir une plus grande densité de points.

Nous proposons de binariser l'image et d'utiliser les points contours car ils se situent dans des zones de fort gradient. Nous avons choisi arbitrairement, les points des contours gauches comme points d'intérêt (cf. figure 2). Nous reviendrons dans la section 3.3 sur l'impact de la binarisation.

(a) image (b) binarisation (c) points d'intérêt

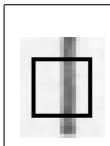
Figure 2. *Choix des points d'intérêt dans l'image*

Cette méthode de sélection a pour effet d'accroître le nombre de descripteurs et donc le temps pour les calculer, mais la sélection à partir des DOG a un coût supérieur à la sélection par les points contours. Ce choix accroît également la durée de la localisation car il augmente le nombre de comparaisons.

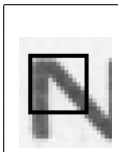
3.2. *Descripteur local, représentation de la luminosité*

Nous avons choisi le descripteur de Lowe souvent cité dans la littérature et clairement expliqué dans (Lowe, 2004). Nous en utilisons la version simple, car il n'est pas nécessaire d'avoir une invariance à la rotation (cf. introduction de la section 3).

Pour construire un descripteur local, Lowe propose de faire des statistiques sur la direction du gradient dans un petit voisinage. Nous avons suivi ses propositions en utilisant une fenêtre de 15x15 pixels, une quantification selon 8 directions et un cumul dans une matrice 3x3. Le descripteur est alors un vecteur de 72 éléments. Le vecteur est normalisé, pour que sa norme euclidienne soit égale à 100 ; c'est cette normalisation qui assure une invariance au contraste de l'image. Ainsi que le préconise Lowe, le calcul du descripteur est fait sur une image filtrée de l'image initiale en niveaux de gris.



19,9	3,0	26,8	63,2
19,7	1,8	26,7	63,2
19,5		26,7	63,2
19,3	1,5	27,0	63,5
19,3	2,9	27,4	63,9



34,5	40,3	62,7	85,9
23,7	19,6	45,4	73,6
25,2		32,2	62,3
37,1	19,1	30,4	56,3
53,5	38,0	39,8	57,5

Figure 3. *Variation de la distance entre le point central et ses voisins ; la partie de gauche montre un point d'intérêt situé au milieu d'un tracé rectiligne, la partie de droite montre un point situé dans l'angle d'un N*

Pour comparer deux descripteurs, en accord avec Lowe, nous utilisons la distance euclidienne. La figure 3 permet d'apprécier la variation de la distance entre un point et ses voisins dans une fenêtre 4x5. On notera que, dans le cas du tracé rectiligne, la distance croît dans les déplacements horizontaux, perpendiculaires au tracé, par contre la distance varie peu pour les déplacements verticaux. Ces valeurs, inférieures à 3, donnent une idée de l'influence du bruit sur le détecteur local. Dans l'image de droite, le point a été choisi dans une région qui comporte plusieurs directions pour le

gradient. Dans ce cas la variation de la distance est sensible dans toutes les directions, le point sera mieux localisable. Ce sont de tels points qu'il faudra utiliser dans les modèles.

3.3. *Localisation d'un modèle*

Un modèle est représenté par un ensemble de points avec leurs coordonnées et les descripteurs associés. On verra, dans les exemples décrits dans la section 4, comment sélectionner les points d'un modèle.

On dira qu'un point d'intérêt d'un modèle est apparié avec un point d'intérêt de l'image lorsque la distance entre leurs descripteurs est inférieure à un seuil. D'après nos expérimentations, ce seuil se situe autour de 60, alors qu'en raison de la normalisation la distance maximum est de $100\sqrt{2}$. Comme on a pu le voir dans la figure 3, la distance ne varie pas beaucoup dans un petit voisinage. Un point du modèle pourra donc être apparié avec un certain nombre de points d'intérêt dans l'image, proches géométriquement les uns des autres qui, par construction, sont les points des contours gauches.

Nous proposons de chaîner les points qui sont 8-voisins. Formellement cela revient à réaliser un étiquetage en composantes 8-connexes des points d'intérêt. Cette opération peut se faire lors de la détection des points dans l'image.

Si un point du modèle est apparié avec plusieurs points d'une même composante connexe, on ne retiendra que la position ayant la plus faible distance.

Dans les images de documents, la binarisation du tracé ne produit, en général, que des erreurs de l'ordre du pixel ; cette faible instabilité est absorbée par le seuil de distance lors de l'appariement (cf. figure 3). Cette instabilité du contour peut aussi provoquer une instabilité dans les composantes connexes, mais celle-ci n'a qu'une très faible influence sur la localisation d'un modèle. Il faut donc comprendre la binarisation comme un moyen de détecter les points d'intérêt, au même titre que les extrema des DOG et non comme une opération de segmentation de l'image.

L'hypothèse forte que nous avons faite dans la section 3.1 - *tous les points d'un modèle doivent être mis en correspondance* - simplifie l'algorithme de localisation d'un modèle dans une image, en particulier le choix du premier point n'a pas d'influence sur la localisation.

Dans un premier temps, on recherche toutes les localisations du premier point du modèle. Ces positions seront les germes d'une localisation à valider. Comme le descripteur local est assez discriminant, le nombre de germes dans une image est faible, quelques dizaines (on trouvera des valeurs numériques dans la section 4).

À partir de chacun des germes, la position relative du point suivant dans le modèle définit une petite zone de recherche dans l'image. La taille de cette zone est définie à priori en fonction de la connaissance que l'on a sur la déformation des mots (manus-

crits, imprimés avec des polices différentes, influence d'une petite rotation, ...). Si la zone ne contient aucune localisation du point, le germe est abandonné (cf. figure 4 c). Si elle en contient une seule, la validation de la localisation du modèle est poursuivie avec le point suivant du modèle.

Si la zone de recherche contient plusieurs localisations (cas du «t» de figure 4 b), on débute une recherche combinatoire. L'expérience montre qu'il n'y a pratiquement pas d'explosion combinatoire et que la localisation du modèle est très rapide. La qualité d'une localisation est évaluée par la somme des distances de chaque point du modèle avec son correspondant dans l'image divisée par le nombre de points. Cette évaluation permet de ne garder que la meilleure localisation lors de l'exploration combinatoire et de l'associer au germe en tant qu'indice de dissimilarité.

(a) modèle avec 4 points (b) zones de recherche (c) échec de la localisation

Figure 4. *Localisation, la zone du t génère 3 localisations, et donc 3 zones par la suite*

En conclusion, notre méthode se caractérise par une description relativement dense de l'image par les points d'intérêt, mais avec peu de points pour représenter le modèle. Ceux-ci devront être tous appariés avec des points de l'image.

4. Utilisation dans différents cadres applicatifs

Nous avons appliqué la méthode décrite ci-dessus pour résoudre trois types de localisation de mots. Il a fallu pour cela introduire dans chacune d'elles quelques adaptations par rapport aux principes généraux décrits ci-dessus. L'évaluation des temps de calcul a été faite sur un MacBookPro avec un processeurs Intel Core i7 à 2.66 Ghz.

4.1. *En-tête de champs*

Un certain nombre de formulaires administratifs se composent de lignes. Ces lignes comportent en général un en-tête imprimé et de la place pour insérer une mention manuscrite. Pour récupérer l'information manuscrite, il faut être capable de retrouver la bonne ligne et parfois d'utiliser un reconnaisseur adapté (chiffre, date, ...).

Les formulaires que nous avons traités sont des demandes de cartes d'identité datant des années 40. Un des objectifs du traitement est d'extraire des photos selon certains critères comme, par exemple, l'année de naissance. Cette collection comporte au moins six variantes (cf. figure 5) qui diffèrent aussi bien par la structure que par la typographie. Les formulaires remplis ont été archivés selon l'ordre alphabétique du

nom. Ils ont donc été numérisés dans cet ordre, ce qui rend totalement aléatoire l'ordre d'apparition des fonds imprimés.

C'est dans ce type de collections qu'il est particulièrement utile de pouvoir identifier l'en-tête de chaque ligne.

Figure 5. *Quelques variantes du formulaire*

Ces formulaires sont imprimés sur des papiers de couleurs différentes : des beiges, du rouge, du vert. Les images ont été numérisées en couleur à 300 dpi, et font approximativement 1200x2000 pixels. Elles ont été sauvegardées en JPEG.

4.1.1. *Définition des points d'intérêt*

La zone décrite par un descripteur local (ici une matrice 15x15) doit être adaptée à la taille de l'écriture, c'est-à-dire de la taille d'un caractère, éventuellement un peu plus petit. Afin de réaliser cette adaptation, on applique un zoom pour obtenir des images à 150 dpi. Les images sont donc de 600x1000 pixels et l'on utilise la traduction en niveaux de gris fournie par la bibliothèque JPEG.

La définition des points d'intérêt est celle donnée dans la section 3.1, on rajoute simplement une suppression simple des contours verticaux trop longs, qui correspondent aux filets qui encadrent le formulaire. Cette suppression permet un gain de 20% sur le nombre de points. On obtient environ 10 000 points d'intérêt par image, ce qui est nettement plus que les 800 points dont parle Lowe dans son article.

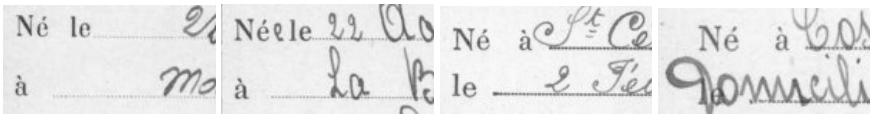
4.1.2. *Construction des modèles*

Dans ce type d'application, comme il y a peu d'en-têtes différents, nous avons voulu tester les performances d'identification des champs avec un seul modèle par champ. Pour créer un modèle, l'opérateur choisit une image dans laquelle l'identifiant n'est perturbé ni par des défauts d'impression ni par les tracés manuscrits. Ensuite une interface lui permet de choisir parmi les points d'intérêt d'une image, les points qu'il considère comme discriminants. Ce logiciel indique au fur et à mesure de la construction du modèle toutes les localisations possibles dans l'image choisie. L'opé-

rateur n'apprécie donc ses choix qu'au travers d'une seule image. La construction d'un modèle est assez rapide, il suffit de choisir 4 à 6 points.

La modélisation des champs comme *Nom*, *Prénom*, *Nationalité* ne pose pas de problème particulier, car le mot est suffisamment long et différe des autres en-têtes ; de plus le fait que ces identifiants soient présents une fois et une seule dans l'image, limite les erreurs de localisation.

Pour la détection de la date et du lieu de naissance, il y a deux variantes *Né le* et *Né à* qui sont assez proches (voir figure 6 a et c). D'autre part la localisation de la date lorsque l'en-tête se limite à *le* n'est pas fiable. En définissant le modèle au moyen de deux lignes *Né-le-2li*, (voir figure 7) on retrouve une plus grande fiabilité. C'est dans ce type de configurations que la localisation à partir des points d'intérêt apporte un avantage indéniable. On retrouverait le même avantage pour des en-têtes de colonne qui sont parfois sur plusieurs lignes.



(a) Né le (b) insertion d'un e (c) Né à (d) Domicile

Figure 6. *Exemples de variantes et de surcharges pour la naissance*

Comme il y a peu de modèles à construire, cette approche interactive a été jugée suffisante pour valider la phase de localisation, elle sera améliorée.

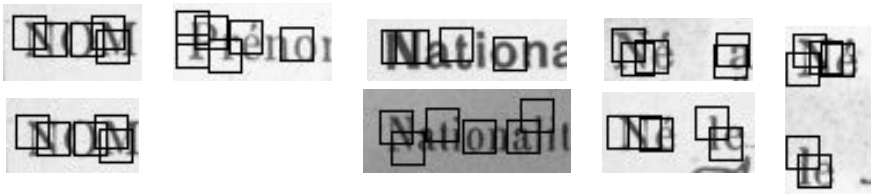


Figure 7. *Les 8 modèles utilisés, à la même échelle ; les carrés noirs indiquent la zone couverte par le descripteur.*

4.1.3. *Localisation des modèles*

On travaille sur toute l'image, il n'y a aucune segmentation en mots et l'on tente de localiser chacun des modèles associés à la collection.

Il s'avère que le descripteur local SIFT est assez discriminant car, pour une mise en correspondance, il sélectionne en général moins de 100 points d'intérêt (sur 10 000 dans l'image) et que ceux-ci forment une petite vingtaine de germes à valider (*cf.* section 3.3).

Lorsqu'un modèle donne lieu à une localisation en plusieurs endroits dans l'image, on ne garde que la meilleure.

4.1.4. *Résultats*

Les tests ont porté sur 793 images et il n'y a au départ qu'un modèle par champ. Dans le tableau 1, on constate globalement qu'il y a une bonne détection, peu de non détections et peu d'erreurs de localisation.

Pour la localisation avec un seul modèle de *Nom* (au moins 2 typographies avec une variation de longueur de 20%) et de *Nationalité* (au moins 5 typographies, police, gras, italique, taille) il y a très peu d'erreur, Il existe cependant un pourcentage important de non détection. En examinant les images sans détection on constate qu'elles appartiennent pratiquement toutes à la même variante du formulaire. Nous avons alors décidé d'ajouter un second modèle (voir figure 7 ligne du bas) en appliquant le processus de construction sur la liste des images en erreur ; les cas de non détections ont très largement diminué (moins de 2%).

Ident	Prés	Mod	Détecté		Non détecté		Erreur local.		Précis.
Nom	792	1	748	94,4%	44	5,6%	0	0,0%	100,0%
		2	786	99,2%	6	0,8%	0	0,0%	100,0%
Prénom	793	1	788	99,4%	5	0,6%	0	0,0%	100,0%
Nationa.	793	1	713	89,9%	80	10,1%	0	0,0%	100,0%
		2	778	98,1%	15	1,9%	0	0,0%	100,0%
Né-le	268	1	265	98,9%	3	1,1%	47	17,5%	84,9%
Né-a	525	1	498	94,9%	27	5,2%	23	4,4%	95,6%
Né-le+	268	1	265	98,9%	3	1,1%	1	0,4%	99,6%
Né-a+	525	1	498	94,9%	27	5,1%	20	3,8%	96,1%
Né-le-2li	527	1	496	94,1%	31	5,9%	51	9,7%	90,7%

Tableau 1. *Performances avec un ou deux modèles*

Les identifiants *Né-le*, *Né-a* et *Né-le-2li* sont moins caractéristiques. Ils obtiennent des performances un peu moins bonnes. L'existence de l'insertion d'un *e* (voir figure 6 b) ne perturbe pas trop la localisation, tandis que la surcharge avec *Domicile* (voir figure 6 d) explique quelques erreurs de localisation de *Né-le-2li*.

On note aussi un pourcentage important d'erreur de localisation. Nous pensons qu'une bonne partie de ces erreurs peut être détectée au niveau de l'analyse de la structure du document. Il suffirait d'introduire des vérifications de cohérences, comme des choix exclusifs ou l'alignement vertical des identifiants. Pour valider cette idée nous avons fait des tests en introduisant le fait que les deux identifiants *Né-le* et *Né-a* ne peuvent pas être simultanément présents dans une image. Les résultats donnés dans les lignes *Né-le+* et *Né-a+* montrent qu'effectivement les fausses localisations de *Né-le* à la place de *Né-a* ont pratiquement disparu.

Le temps de traitement d'une image avec un dictionnaire de 6 à 8 modèles est d'environ 300 ms, le calcul des descripteurs occupe environ 90% du temps.

Afin de valider notre choix de travailler à 150 dpi, nous avons fait des essais à 300 dpi. Les performances sont légèrement moins bonnes et, comme il fallait s'y attendre, le temps de calcul est doublé, car le nombre de points contours est doublé et non multiplié par 4.

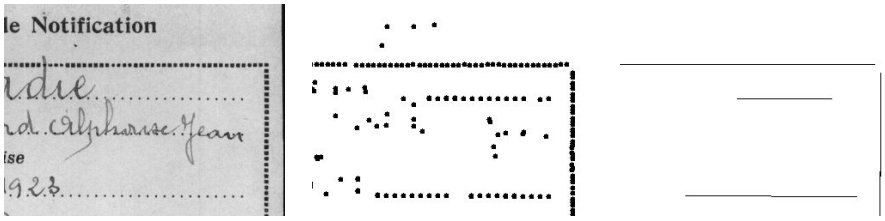
Ces résultats montrent que le choix d'une densité importante de points décrivant l'image, d'un descripteur local fondé sur les SIFT et de modèles construits très simplement fournit une chaîne très efficace pour l'identification des en-tête de champs. Cette chaîne ne nécessite aucune segmentation préalable et peut venir en complément d'une détection de la structure.

4.2. *Détection de lignes pointillées*

L'utilisation des points d'intérêt peut fournir une méthode assez générique pour détecter les lignes pointillées.

Dans la collection présentée dans l'exemple ci-dessus, une partie de la structure du formulaire est, dans certaines impressions, définie par des lignes en pointillé plutôt que par des filets.

Nous proposons alors de constituer deux modèles ne comportant que deux points d'intérêt : un modèle pour les lignes horizontales et un pour les lignes verticales. Ensuite lors de la localisation on cherche à concaténer les localisations : le point d'intérêt terminant une localisation devient le point de départ de la localisation suivante. En fin de localisation, on vérifie qu'il y a un nombre suffisant de concaténations pour former une ligne. Il appartiendra à la partie structurelle de l'analyse de ces images de définir les lignes pointillées qui correspondent au cadre du formulaire.



(a) image (b) points germes (c) lignes détectées

Figure 8. *Détection de lignes en pointillé dans des images à 150 dpi*

L'implémentation actuelle tolère la perte d'un point, ce qui peut arriver en raison soit de défauts d'impression, soit de surcharges avec du tracé manuscrit ou des cachets.

Il n'est pas facile de donner des résultats quantitatifs pour cette méthode, mais elle a été utilisée avec succès sur un millier d'images par l'application chargée de détecter la structure de l'image pour en extraire les photos.

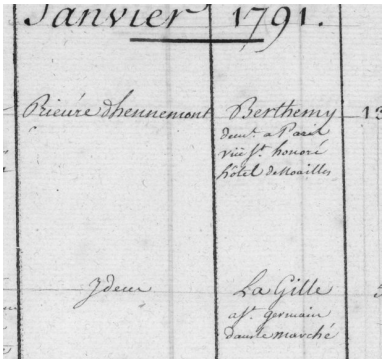
Cette méthode s'étend facilement à des pointillés plus compliqués ; il suffit d'augmenter le nombre de points du modèle pour couvrir le motif engendrant le pointillé. Elle s'étend également à n'importe quelle direction connue a priori, ce qui est fréquent dans les formulaires, mais ce qui n'est pas adapté aux dessins de plans industriels.

4.3. *Indexation de noms propres manuscrits*

L'indexation automatique des documents est un objectif important pour faciliter l'accès aux documents anciens numérisés. Malheureusement les performances des reconnaissances de mots manuscrits (HWR) ne sont pas très bonnes dès lors qu'il s'agit de documents anciens comportant des noms propres. En utilisant les capacités de rejet de l'HWR, on peut compenser ces manques par une saisie manuelle des mots mal reconnus, c'est la transcription assistée.

Pour diminuer les saisies, on peut regrouper les régions qui se ressemblent visuellement. Dans un premier temps nous décrivons l'adaptation de notre méthode pour réaliser ces regroupements, puis nous la comparons avec les résultats obtenus dans l'équipe (Guichard *et al.,* 2011) lors de l'implémentation de la méthode DTW proposée dans (Rath *et al.,* 2007).

Les documents traités sont des actes de ventes des biens confisqués lors de la Révolution française. Ceux-ci se présentent sous la forme de formulaires, dont il s'agit de transcrire certains champs ; la figure 9 en donne un exemple. Ces formulaires ont vraisemblablement été écrits, au moins sur quelques pages consécutives, par le même scripteur.



(a) image (b) régions à indexer

Figure 9. *Partie d'une page d'un acte de vente*

Les images ont été numérisées à 300 dpi, en couleur et en double page. Ce qui conduit à des images faisant approximativement 7000x5000 pixels.

Les régions à transcrire sont détectées par un processus indépendant qui ne rentre pas dans le cadre de cet article. Il fournit les coordonnées de chaque région.

Nous procédons en deux passes. La première construit automatiquement un modèle pour chacune des régions à transcrire et archive chaque modèle dans un fichier. La seconde reprend chaque région, en recalcule les points d'intérêt et essaie d'y localiser un sous-ensemble des modèles obtenus lors de la première passe. Cette limitation à un sous-ensemble, également présente dans l'autre approche de l'équipe, vise à utiliser le fait que le vocabulaire des mots varie d'un bout à l'autre de la collection et que ce sont plutôt les mêmes noms qui se retrouvent dans des pages consécutives. Dans les deux approches, la taille du sous-ensemble a été fixée à 200 mots, 100 avant et 100 après la région que l'on cherche à transcrire.

4.3.1. *Définition des points d'intérêt*

Afin d'adapter la taille de l'écriture à la taille du descripteur local, on réalise un zoom pour obtenir des images à 150 dpi. Les points d'intérêt sont calculés région par région. Il y a environ 300 à 400 points d'intérêt dans une région.

4.3.2. *Construction d'un modèle*

Il n'est plus envisageable d'utiliser une sélection manuelle des points d'intérêt, il faut automatiser cette sélection. On utilise pour cela une détection classique du corps du mot pour déterminer les hampes ascendantes ou descendantes et les extrémités des contours près du haut ou du bas du corps. Ces critères permettent de sélectionner parmi les points d'intérêt ceux qui semblent les plus significatifs. L'algorithme n'a plus qu'à en choisir quelques uns (ici 6 au maximum) bien répartis sur la largeur du mot.

On notera que cette procédure de sélection n'a pas besoin d'être stable entre deux images du même mot, par contre il faut que les points choisis pour représenter un modèle trouvent leur correspondant dans les autres images de ce mot. C'est là que la représentation dense des images par les points contours prend toute sa valeur.

4.3.3. *Comparaison entre une région et un modèle*

Le principe de la comparaison entre une région et un modèle est celui qui a été décrit dans la section 3.3.

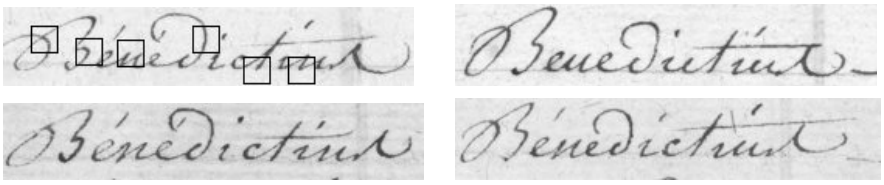


Figure 10. *Un modèle défini par 6 points et 3 régions appariées*

En sortie de la deuxième passe, on obtient une table qui associe à chaque région R_A son indice de dissimilarité *d(R_A, M_B)*, avec le modèle M_B. Cette table est très creuse, car il y a de très nombreux cas où la localisation a échoué. D'autre part, rien,

dans la construction de cette table, ne permet d'assurer qu'elle est symétrique. Supposons par exemple que la région R_A contienne un mot assez long, tandis que la région R_B contient un mot court, qui est un sous-mot de R_A, alors R_B pourra être comparée à M_A tandis que la comparaison de R_A avec M_B échouera. Cette absence de symétrie nous a semblé un indice important de mauvais appariement. Nous avons donc supprimé de la table toutes les entrées qui n'avaient pas leur pendant dans l'autre sens : *d(R_A, M_B)* et *d(R_B, M_A)*, doivent exister tous les deux.

Afin d'adapter les paramètres, une première évaluation a été faite sur 76 images qui fournissent 1203 régions. Il y a 367 identifiants qui ne sont utilisés qu'une seule fois, on a donc au maximum 836 appariements.

La construction d'un modèle prend 45 ms par région, tandis que l'appariement d'une région avec les 200 modèles prend 83 ms. Là encore c'est le temps de calcul des descripteurs qui est le plus important.

Le tableau 2 montre d'une part les performances en reconnaissance et d'autre part l'influence du seuil sur ces performances. La colonne *Correct* compte les régions qui ne sont reconnues que par d'autres modèles du même mot. La colonne *Correct + erreur* compte les régions qui sont reconnues à la fois par un modèle du bon mot et par des modèles de mots différents. Enfin la colonne *Erreur* compte les régions qui sont reconnues par un modèle d'un mot différent.

Seuil	Correct	Correct + erreur	Erreur	Absent
58	453	2	8	273
60	470	2	12	256
62	473	4	16	251

Tableau 2. *Evolution des performances en fonction du seuil*

On constate que d'une part le nombre d'erreurs reste faible et que d'autre part il manque une bonne proportion de mise en correspondance.

Lorsque le seuil croît, le mécanisme de mise en correspondance est plus tolérant ; comme il fallait s'y attendre le nombre de régions correctes augmente, sans que le taux d'erreur augmente beaucoup.

4.3.4. *Comparaison avec la méthode DTW*

Pour évaluer le gain d'une méthode de transcription, on calcule le pourcentage de régions qui nécessitent une saisie manuelle. Ce taux, qui doit être le plus bas possible, varie en fonction du taux d'erreur résiduel que l'on tolère. Nous présentons les résultats dans le tableau 3 avec trois réglages des paramètres adaptés à trois objectifs pour les taux résiduels. Ils ont été obtenus sur plus de 11 000 régions.

On peut utiliser un HWR avec de bonnes propriétés de rejet sur chaque région. On voit, par exemple, dans la colonne HWR *seul* du tableau, que si l'on tolère 5% d'erreur, il faut saisir en moyenne une région sur deux.

On peut également regrouper les régions qui se ressemblent visuellement par une technique classique de *clustering*. Ce regroupement a deux avantages : il permet d'une part de fusionner les résultats de l'HWR sur tous les éléments d'un cluster, ce qui fiabilise sa réponse, et d'autre part en cas de rejet, la saisie pour une région se propage à toutes les autres.

Nous avons donc testé la ressemblance visuelle en utilisant la méthode DTW et la méthode avec les points d'intérêt que nous venons de décrire.

	Taux de saisie manuelle		
Taux d'erreur	HWR seul	HWR + clustering DTW	HWR + clustering POI
1%	75,8%	58,0%	55,6%
5%	50,8%	39,4%	36,5%
20%	20,2%	12,1%	10,7%

Tableau 3. *Variation du taux de saisie en fonction de la méthode et du taux d'erreur résiduelles visé. Le taux doit être le plus bas possible*

L'introduction du clustering diminue de manière importante le taux de saisie. La méthode avec les points d'intérêt donne des résultats légèrement meilleurs que la méthode DTW, et elle est environ 10 fois plus rapide. Ce gain en vitesse est dû au fait que la comparaison entre une région et un modèle élimine très rapidement les modèles qui ne sont pas ressemblants. Ces regroupements ont mis en évidence un cluster avec plus de 1500 régions. Il correspond au mot *Idem*.

5. Conclusion

Les résultats montrent que le descripteur local SIFT est adapté aux traits fins, comme ceux de l'écriture ; il fournit une représentation locale discriminante. Cet avis très positif ne doit pas empêcher de faire des essais avec d'autres descripteurs.

L'utilisation d'une représentation dense des images et d'une représentation parcimonieuse des modèles apporte une réponse à la difficulté de détection des points d'intérêt d'une image de documents, sans trop alourdir les temps de calcul.

La méthode construit des modèles spécifiques à partir de peu d'exemples (1 ou 2) par opposition à des méthodes qui construisent des modèles génériques à partir d'un apprentissage statistiques sur un grand nombre d'exemples, comme celle de (Rodriguez *et al.,* 2008) ou celle de détection des visages de (Viola *et al.,* 2004).

Enfin il faudrait approfondir les méthodes de construction des modèles, en particulier il faudrait définir ce qu'est un point discriminant pour l'écriture manuscrite. Cette réflexion doit être liée à la définition du nombre de points d'un modèle, nécessaire pour affirmer qu'il y a une mise en correspondance entre un modèle et l'image.

Les images de demandes de cartes d'identité ont été fournies dans le cadre d'un contrat avec les archives départementales de la Mayenne. Celles des ventes révolutionnaires l'ont été dans le cadre d'une coopération avec les archives départementales des Yvelines, soutenue par le conseil général des Yvelines.

6. Bibliographie

Bay H., Ess A., Tuytelarrs T., Van Gool L., « Speeded-Up Robust Features (SURF) », *Computer Vision and Image Understanding*, vol. 110, n° 3, p. 346-359, 2008.

Guichard L., Chazalon J., Coüsanon B., « Exploiting Collection Level for Improving Assisted Handwritten Words Transcription of Historical Documents », *11e International Conference on Document Analysis and Recognition ICDAR 2011*, Beijing, p. 875-879, 2011.

Harris C., Stephens M., « A Combined Corner and Edge Detection », *4e Alvey Vision Conférence*, Manchester, p. 147-151, 1988.

Leydier D., Ouji A., Lebourgeois F., Emptoz H., « Towards an omnilingual word retrieval system for ancient manuscripts. », *Pattern Recognition*, vol. 42, n° 9, p. 2089-2105, 2009.

Lowe D. G., « Distinctive Image Features from Scale-Invariant Keypoints », *International Journal of Computer Vision*, vol. 60, n° 2, p. 91-110, 2004.

Mikolajczyk K., Schmid C., « A Performance Evaluation of Local Descriptors », *Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, n° 10, p. 1615-1630, 2005a.

Mikolajczyk K., Tuytelaars T., Schmid C., Zisserman A., Matas J., Schaffalitzky F., Kadir T., Van Gool L., « A Comparison of Affine Region Detectors », *International Journal of Computer Vision*, vol. 65, n° 1, p. 43-72, 2005b.

Rath T. M., Manmatha R., « Word Spotting for Historical Documents », *International Journal on Document Analysis and Recognition*, vol. 9, n° 2-4, p. 139-152, 2007.

Rodriguez J. A., Peronnin F., « Local gradient histogram features for word spotting in unconstrained handwritten documents », *1st International Conference on Handwriting Recognition ICFHR'08*, Québec, 2008.

Rusinol M., Aldavert D., Toledo R., Llados J., « Browsing Heterogeneous Document Collections by a Segmentation-free Word Spotting Method », *11e International Conference on Document Analysis and Recognition ICDAR 2011*, Beijing, p. 63-67, 2011.

Schmid C., Mohr R., Bauckhage C., « Evaluation of Interest Point Detectors », *International Journal of Computer Vision*, vol. 37, n° 2, p. 151-172, 2000.

Song W., Liwicki M., « Look Inside the World of Parts of Handwriten Characters », *11e International Conference on Document Analysis and Recognition ICDAR 2011*, Beijing, p. 784-788, 2011.

Viola P., Jones M., « Robust Real-Time Face Detection », *International Journal of Computer Vision*, vol. 57, n° 2, p. 137-154, 2004.