Alignement texte-image sans apprentissage pour les manuscrits médiévaux[1]

Yann LEYDIER[1] — Véronique ÉGLIN[1] — Stéphane BRÈS[1] — Dominique STUTZMANN[2]

[1] *Université de Lyon, CNRS, INSA-Lyon, LIRIS, UMR5205, F-69621, France*
Email: yann@leydier.info, veronique.eglin@insa-lyon.fr, stephane.bres@insa-lyon.fr
[2] *Institut de Recherche et d'Histoire des Textes, CNRS, UPR841, F-75016, France*
Email: dominique.stutzmann@irht.cnrs.fr

RÉSUMÉ. *Dans cet article, nous décrivons une nouvelle approche pour l'alignement texte-image des documents du moyen-âge. La méthode est dédiée à l'alignement au niveau mot, sans segmentation ni apprentissage. L'alignement mot à mot est basé sur une distance d'édition appliquée à des signatures extraites à la fois à partir des chaînes de caractères Unicode et à partir des images. Les résultats sont évalués sur la "Queste del saint Graal" (XIIIème s.) par des paléographes grâce à une interface de validation intuitive qui permet une correction interactive très rapide. Le gain de temps résultant de l'absence d'une phase d'apprentissage permet d'apporter plus d'attention à l'intégration des différentes spécificités et des variations des écritures du moyen-âge (abréviations, allographes…).*

ABSTRACT. *In this paper, we describe a new approach for text-image alignment of middle-age documents. The method is dedicated to word-to-word alignment in a segmentation-free and learning-free way. The best word-to-word matching lies on an adapted editing cost between signatures extracted on Unicode characters and on the images. The results are evaluated on the "Queste del saint Graal" (13th c.) by palaeographers through an intuitive validation interface that also offers a very fast interactive correction process. The gain of time resulting from the absence of learning stage offers the opportunity to pay more attention to the integration of different specificities and variations in middle-age handwritten documents (presence of typical abbreviations, allographs…).*

MOTS-CLÉS : *alignement texte-image, manuscrit médiéval, programmation dynamique*

KEYWORDS: *text-image alignment, medieval manuscript, dynamic programming*

1. Traduction d'un article publié en anglais à ICFHR 2014 (Leydier *et al.*, 2014)

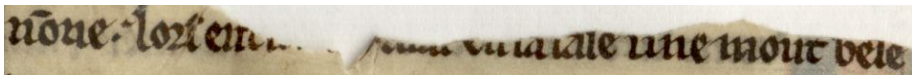
1. Introduction

Les manuscrits médiévaux sont la principale source pour notre compréhension de l'histoire sociale, économique, politique et religieuse du moyen-âge, et sont aussi de précieux témoins de phénomènes linguistiques. Ces manuscrits sont fragiles et ne doivent pas être manipulés fréquemment. Par conséquent, ils sont numérisés dès que possible.

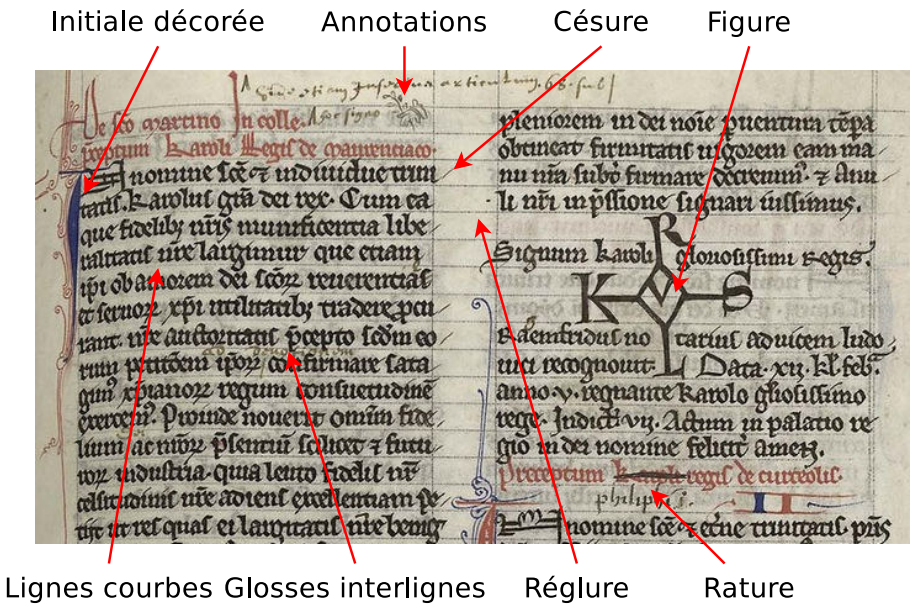
Les manuscrits médiévaux sont des objets très complexes. Le parchemin peut avoir un grain très visible et est souvent endommagé par des déchirures (voir fig. 1(a)), des trous ou des taches. La mise en page n'est pas normalisée et les copistes étaient libres de faire ce qu'ils voulaient : colonnes non-rectangulaires, textes entrelacés, petites ou grandes initiales ornées (voir fig. 1(b))... La couleur était souvent utilisée dans le texte et il était commun de rejeter les derniers mots d'un paragraphe sur une ligne suivante sur le côté droit. Les abréviations ont été largement utilisées mais très rarement normalisées (voir fig. 1(c)). Certaines lettres avaient plusieurs allographes tels que le 's' rond et le 's' long (voir fig. 1(d)) ou les 'r' droit ou rond (voir la troisième et l'avant-dernière lettre dans la figure 2(a)), qui sont rarement transcrites comme telles par les chercheurs (voir également fig. 1(e)). En outre, l'espacement n'était pas clairement défini et l'espace entre les mots est rarement plus grand que l'espace entre les caractères (voir fig. 2).

En 2007, nous avons proposé une méthode de *word-spotting* (Leydier *et al.*, 2007) qui permettait de collecter toutes les occurrences d'un mot dans un manuscrit. L'utilisateur avait juste à élaguer visuellement une courte liste de mots au lieu de parcourir et lire des livres entiers. En 2009, nous avons proposé une méthode de *word retrieval* (Leydier *et al.*, 2009) qui étendait la précédente et permettait à l'utilisateur de saisir le mot à rechercher au clavier, ce qui offrait un accès encore plus simple au contenu de ces documents. Ces méthodes de *word-spotting* étaient basées sur des comparaisons d'images flexibles et cohésives (basées sur des régions d'intérêt), sans aucune connaissance sur la transcription textuelle ou les styles d'écriture.

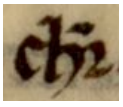
Les étapes suivantes dans l'indexation des manuscrits médiévaux sont la transcription assistée par ordinateur (TAO) puis la reconnaissance automatique (OCR). La TAO est un processus dans lequel l'utilisateur transcrit manuellement quelques pages (ou quelques mots choisis au hasard) pour apprendre à la machine à reconnaître le reste du document. Cela fonctionne bien sur les documents imprimés anciens (Ramel *et al.*, 2013) mais la création d'un système de TAO pour manuscrits médiévaux exige une connaissance fine des caractéristiques de l'écriture pour être en mesure d'intégrer les variations allographiques dans la conception des caractères. Aligner les images sur un texte déjà transcrit aidera grandement l'étude des écrits médiévaux.



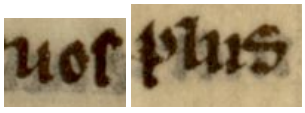
(a) Page déchirée



(b) Quelques caractéristiques des manuscrits médiévaux



(c) Abbreviation de 'chevalier'



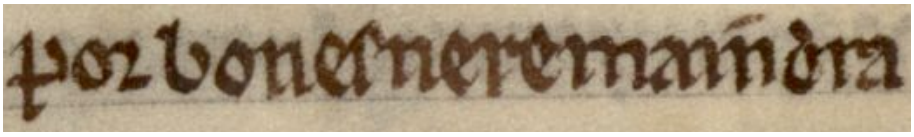
(d) s long et s rond

(e) 'Et' / 'et'

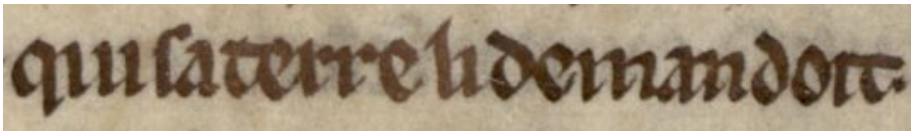
Figure 1. *Extraits d'un manuscrit médiéval*

2. Synthèse bibliographique

Hobby (Hobby, 1998) et Jawahar *et al.* (Jawahar et Kumar, 2007) ont proposé des méthodes d'alignement respectivement pour les documents en anglais et indiens imprimés. Elles sont basées sur des composantes connexes extraites de l'image numérisée et d'une image synthétique du texte. Comme le nombre de composantes connexes n'est pas le même dans les deux images, une *Dynamic Time Warping* est utilisé pour les jumeler.



(a) “por bones ne remaindra”



(b) “qui sa terre li demandoit .”

Figure 2. *La segmentation automatique en mots est impossible*

Hassner *et al.* (Hassner *et al.*, 2013) utilisent plusieurs polices pour créer des images de synthèse qui sont comparées en utilisant un mécanisme de flux de SIFT couplé avec un descripteur de texture locale basé sur des pixels binaires.

Nous ne pouvons pas exploiter ce type de méthodes puisque nous ne pouvons pas créer des images qui ressemblent à l’original en raison de la variabilité de manuscrits. Même s’il existe certaines polices imitant les écritures médiévales, il leur manque souvent des caractères et elles ne modélisent pas la grande variété d’écritures du moyen-âge. En outre, dans un seul manuscrit médiéval, les composantes connexes peuvent représenter des objets textuels à des échelles variables (du fragment de texte jusqu’à un groupe de mots).

Zinger *et al.* ont proposé une méthode d’alignement pour manuscrits du 20e siècle (Zinger *et al.*, 2009). La segmentation en mots est basée sur les espaces entre les portions de texte sur la ligne de base et fait usage d’un apprentissage sur un ensemble de pages alignées manuellement. L’alignement est réalisé en utilisant une fonction de coût appliquée sur les mots ou la largeur de caractères.

Nous ne voulons pas forcer nos utilisateurs finaux à effectuer une telle lourde étape d’apprentissage.

Beaucoup d’autres auteurs, comme Tomai *et al.*, Kornfield *et al.* et Stamatopoulos *et al.* (Stamatopoulos *et al.*, 2010) effectuent une segmentation en lignes et mots traditionnelle avant d’appliquer une programmation dynamique pour réaliser l’alignement. Tomai *et al.* utilisent plusieurs hypothèses de segmentation sur des manuscrits du 18e siècle (Tomai *et al.*, 2002). Kornfield *et al.* utilisent des caractéristiques simples et robustes telles que le nombre de hastes et de hampes sur le journal de Washington (Kornfield *et al.*, 2004).

Comme nous l’avons mentionné auparavant, nous ne pouvons pas effectuer de segmentation en mots, car il n’y a généralement pas d’espaces entre les mots.

Lorigo *et al.* ont proposé un algorithme d'alignement pour les manuscrits arabes (Lorigo et Govindaraju, 2007). Alors que les manuscrits latins n'ont pas tous des espaces entre les mots, les manuscrits arabes ont des espaces à l'intérieur des mots. Les auteurs segmentent les lignes en parties de mots qui sont caractérisées par le nombre de traits ascendants et descendants, le nombre de boucles et de signes diacritiques en dessus ou en dessous de la ligne de base. Ces caractéristiques sont très robustes et simples. Un caractère spécifique est reconnu (un seul trait vertical) et ajouté au vecteur de caractéristiques comme une valeur booléenne.

L'algorithme d'appariement est un peu compliqué en raison des erreurs de segmentation, ce qui nous réconforte dans notre décision d'éviter les méthodes reposant sur une segmentation. Les caractéristiques proposées par les auteurs pourraient convenir pour nos manuscrits latins moyennant des adaptations.

Fischer *et al.* ont travaillé sur l'alignement de manuscrits médiévaux (Fischer *et al.*, 2010)(Fischer *et al.*, 2011). Ils effectuent manuellement une segmentation en blocs de texte en utilisant GIMP et l'utilisateur doit exclure les enluminures et les gloses. Les blocs sont binarisés et les lignes sont séparées en suivant les espaces blancs. Un utilisateur expert doit alors corriger manuellement les erreurs de segmentation résiduelles. La segmentation en mots et l'alignement sont réalisés avec des HMM. D'autres algorithmes basés sur les HMM ont également été proposés dans la littérature (par exemple : Tosseli *et al.* (Toselli *et al.*, 2011)) en particulier sur des manuscrits modernes. Toutefois les HMM sont un outil extrêmement contraignant en termes d'apprentissage et nous avons déjà indiqué que notre méthode ne devrait pas obliger l'utilisateur à créer une base d'apprentissage. De même, nous rejetons toutes les autres méthodes d'alignement basées sur des HMM.

3. Présentation

Sur la base de ces observations, nous avons décidé de créer une méthode qui ne nécessite pas d'apprentissage. Nous avons également choisi de ne pas utiliser de caractéristiques traditionnelles qui nécessitent une étape de binarisation, qui ne peuvent pas donner de résultats suffisamment bons, et de privilégier une approche sans segmentation.

Comme exposé dans l'introduction, les manuscrits médiévaux ont de nombreuses caractéristiques qui les rendent très difficiles à traiter. Pour rendre la tâche plus réaliste, nous avons décidé de concevoir notre première itération avec quelques contraintes :

– les pages peuvent contenir plusieurs colonnes rectangulaires ;

– toutes les lignes de texte ont une hauteur similaire ;

– les enluminures sont au moins aussi hautes que deux lignes de texte ;

– la transcription XML indique les sauts de ligne et les sauts de colonne.

Notre méthode est appliquée directement sur les lignes de texte puisque la segmentation de mots est impossible. Nous avons privilégié une détection de la ligne médiane plutôt qu'une segmentation en lignes traditionnelle dont les résultats auraient été incertains. Le texte Unicode et le texte de l'image seront tous deux associés à des signatures basées sur un alphabet simple. Enfin, les signatures des lignes Unicode et des lignes correspondantes sur l'image seront alignées en utilisant une programmation dynamique.

4. Détection de lignes

Puisqu'il est impossible d'obtenir une segmentation en mots robuste sur les manuscrits du moyen-âge, notre détection de ligne repose sur l'emplacement des lignes médianes associé à une estimation de la hauteur des lignes. Comme notre alignement mettra l'accent sur les formes à proximité des lignes médianes, ce n'est pas grave si les hampes et les hastes de deux lignes successives se croisent ou se touchent.

4.1. *Colonnes*

Soit s_w la largeur moyenne des traits calculée par auto-corrélation horizontale sur l'image en niveaux de gris I_g. Nous définissons l'image I_s comme une version réduite de I_g avec largeur$(I_g)/s_w$ et la même hauteur que I_g. La mise à l'échelle est effectuée par un *min*. De cette façon, les blancs plus étroits que s_w sont supprimés.

Nous calculons la projection verticale de I_s. Cet histogramme contient autant de sommets qu'il y a de colonnes. Comme les colonnes peuvent ne pas avoir les mêmes hauteurs, nous calculons le nombre de colonnes avec plusieurs seuils et choisissons la valeur médiane.

Une telle détection laisse les frontières gauche et droite très imprécises et ne calcule pas les marges supérieure et inférieure, mais cela n'est pas grave puisque la détection de colonnes est juste une aide pour la détection des lignes. En fait, il serait très difficile de parvenir à une bonne segmentation en colonnes avec une méthode descendante puisque les pages sont presque toujours penchées, les lignes ne se terminent jamais vraiment à la même abscisse, et des ornements peuvent être inclus dans les marges des colonnes de texte.

4.2. *Lignes*

Soit l_h la hauteur moyenne des lignes calculée avec une auto-corrélation verticale sur I_s.

Nous calculons les lignes médianes sans binarisation avec une simple dérivée le long de l'axe Y. Cette dérivée (I_y) est obtenue par convolution de l'image avec la dérivée première d'un vecteur-colonne gaussien avec $\sigma = l_h/3$. Une telle valeur de σ

assure que le vecteur de convolution a une taille de l_h et peut efficacement extraire les lignes de texte. Quand un pixel de I_y est négatif et que son voisin du bas est positif, le point appartient à une ligne médiane (voir fig. 3).

Comme le bruit dans I_g a été lissé dans I_s, il y a des fausses détections. Les fausses lignes sont dues à de petites variations de l'arrière-plan et peuvent être facilement filtrées : la dérivée en Y autour d'elles est très faible.

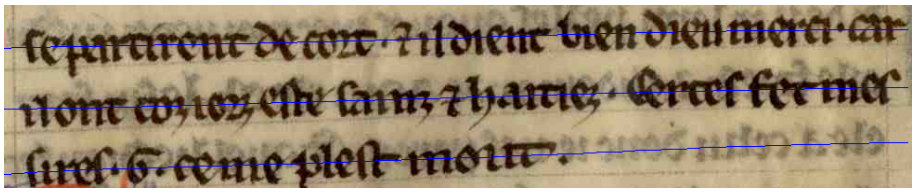


Figure 3. *Lignes médianes (en bleu)*

4.3. *Enluminures*

Dans cette étude, nous avons choisi d'exclure l'alignement des enluminures car elles doivent être traitées différemment du texte normal.

Les enluminures ne sont pas seulement de grandes zones colorées. Certaines encres étaient assez ternes et, comme indiqué précédemment, le parchemin n'est pas blanc. Nous définissons donc les enluminures comme étant soit des zones colorées (dans l'espace TSV, $S > 127$ et $V > 127$) ou des zones ternes qui diffèrent de la teinte du parchemin. La teinte du parchemin est considérée comme la teinte moyenne de l'image μ_T et est généralement très semblable à la teinte du texte régulier. σ_T est l'écart type de la teinte de l'image. Nous classons une zone terne comme enluminure si $|T - \mu_T| > 4 \times \sigma_T$.

Puisque les images sont très bruitées, nous ne gardons que les zones dont la hauteur est deux fois plus grand que l_h. Enfin, toutes les lignes calculées dans la section précédente sont coupées si elles franchissent une enluminure.

5. Alignement

Nous avons développé un alignement à l'intérieur des lignes basé sur la transcription Unicode et les lignes de texte sur l'image. Chaque mode de représentation est associé à des signatures qui peuvent être comparées et alignées.

5.1. *Signature*

Nous avons choisi des éléments de signature qui peuvent être calculés soit le long d'une ligne médiane sur l'image ou générés à partir d'une chaîne Unicode. Avec l'aide de paléographes, nous avons sélectionné six éléments visuels significatifs décrits dans le tableau 1.

`.`	un point près de la ligne médiane	fig. 4 élément 5
`(`	une courbe faisant face à la gauche	fig. 4 élément 4
`)`	une courbe faisant face à la droite	fig. 4 élément 1
`'`	un trait court au dessus de la ligne médiane	fig. 4 élément 2
`,`	un trait court en dessous de la ligne médiane	fig. 4 élément 6
`l`	un trait vertical qui travers la ligne médiane	fig. 4 élément 3

Tableau 1. *Éléments de signature*

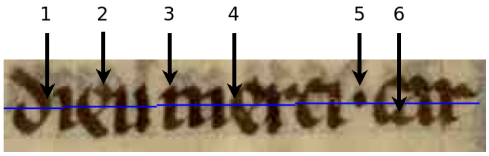


Figure 4. *Éléments de signature (ligne médiane en bleu)*

5.1.1. *À partir d'une chaîne Unicode*

Une table de conversion est fournie pour chaque famille de l'écriture. Chaque caractère est associé à une chaîne (voir tableau 2).

Caractère	Signature
a	`,l`
b	`l)`
c	`('`
f	`l'`
m	`lll`

Tableau 2. *Exemple de signatures de caractères pour une écrite proto-gothique.*

Le texte Unicode est extrait à partir de fichiers XML TEI contenant des transcriptions normalisées, diplomatiques et parfois facsimilé qui doivent être interprétées pour obtenir un texte aussi proche que possible de ce qui a été écrit sur le document (voir fig. 5).

```
<w type="NOMpro" xml:id="w106_000100">
  <choice>
    <me:norm>Lancelot</me:norm>
    <me:dipl>lanc<ex>elot</ex></me:dipl>
    <me:facs>lan<bfm:mdvAbbr>c&bar;.
              </bfm:mdvAbbr></me:facs>
  </choice>
</w>
```

Figure 5. *Extrait d'une fichier XML TEI. Le mot sur l'image est 'lanc̄.'*

5.1.2. *À partir de l'image*

Les signatures sont extraites en trois étapes à partir du champ de gradient de l'image $\mathcal{G} = \binom{L_x}{L_y}$ et la dérivée seconde le long de la tangente aux isophotes L_{vv} (voir fig. 6) :

$$L_{vv} = \frac{L_x^2 \cdot L_{yy} + L_y^2 \cdot L_{xx} - L_x \cdot L_y \cdot (L_{xy} + L_{yx})}{L_x^2 + L_y^2}$$

Le gradient d'un pixel est considéré comme significatif si $||\mathcal{G}(x)|| > m$, où m est calculé automatiquement sur la page entière en utilisant des moyennes de modules de gradient pondérées par les niveaux de gris ($m = \left(\frac{\sum L(x) \times ||\mathcal{G}(x)||}{\sum L(x)} + \frac{\sum (1-L(x)) \times ||\mathcal{G}(x)||}{\sum 1-L(x)}\right)/2$).

(a) Image RVB (b) Angles des gradients (c) L_{vv}

Figure 6. *Caractéristiques différentielles : angles de gradients significatifs et L_{vv}*

Les éléments de signature '`''`', '`,`' et '`l`' sont extraits lors d'une première étape. Une image binaire est créée pour représenter les lignes horizontales entre gradients significatifs pointant respectivement vers la gauche et la droite (voir fig. 7(a)). Pour chaque ligne, les pixels contenus dans la bande de hauteur l_h sont projetés verticalement au-dessus et en dessous de la ligne médiane (voir fig. 7(b)) et les modes des deux projections sont extraits (voir fig. 7(c)). Nous fusionnons les modes supérieurs et inférieurs quand ils sont reliés sur la ligne médiane pour obtenir les éléments de signature de type '`l`'. Les autres modes sont les éléments de signature '`''`' et '`,`'.

Les points sont calculés lors de la seconde étape. Nous créons une image binaire contenant les pixels pour lesquels $L_{vv} > 0$ et pour lesquels le gradient est significatif (voir fig. 7(d)). Nous extrayons ensuite les composantes connexes. Comme les points

dans les manuscrits médiévaux ont une forme semblable à un losange (en raison de la forme d'une plume), un point doit avoir une surface approximativement égale à la moitié de la superficie de sa boîte englobante rectangulaire. Pour veiller à ce que les composantes connexes sélectionnées soient vraiment des points, nous créons un histogramme de 8 angles de gradients. Si l'histogramme est plat ($\sqrt{\text{MSE}} < 0,05$), alors nous considérons la composante connexe comme un point (voir fig. 7(e)).

Dans la dernière étape, nous détectons les courbes. Une autre image binaire est créée à partir des gradients significatifs orientés vers la gauche (voir fig. 7(f)) ou à droite et nous extrayons les composantes connexes. Pour chaque composante connexe, nous comptons le nombre de pixels convexes ($L_{vv} > 0$) et de pixels concaves ($L_{vv} < 0$). Si la quantité de pixels convexes est supérieure à 75% de la surface, nous considérons que le composante connexe est une courbe (voir fig. 7(g)).

5.2. *Alignement*

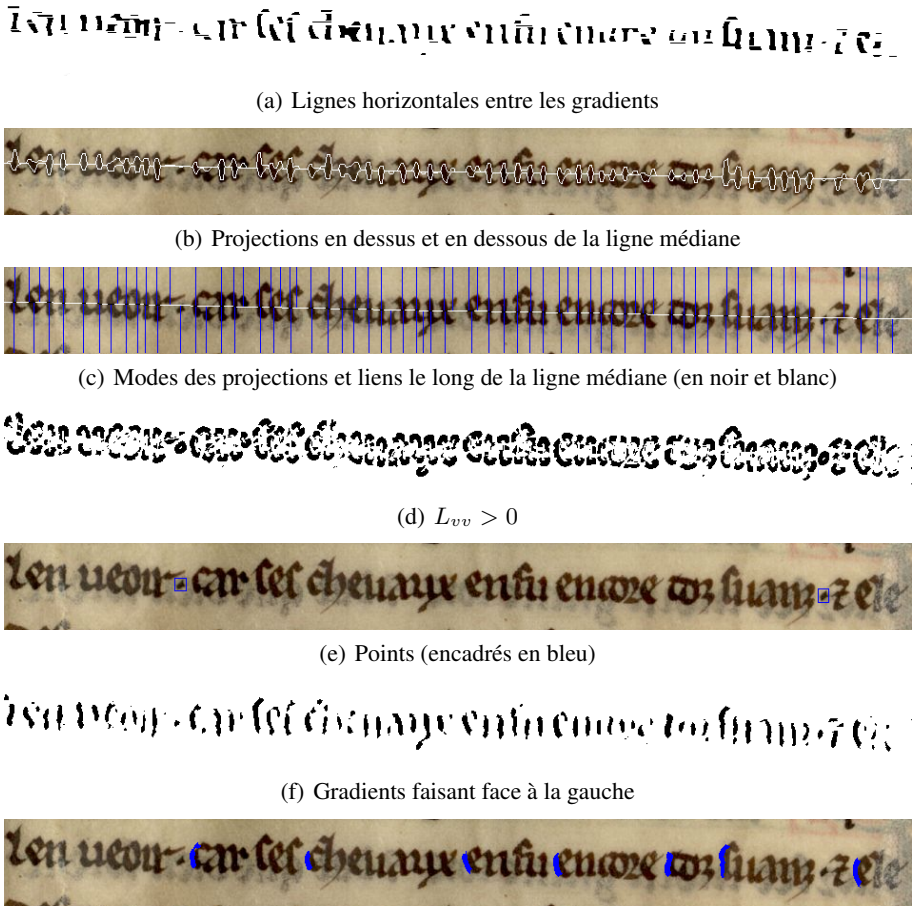
L'alignement est réalisé entre les chaînes de signature représentant une ligne de texte (ou une partie de ligne) sur l'image et dans la transcription. Nous calculons la matrice de transformation de la distance Levenshtein en utilisant les paramètres dans le tableau 3.

Puisque notre algorithme de détection de points a une bonne précision (c.à.d. très peu de fausses détections), nous les utilisons comme points d'ancrage : lorsqu'un point se trouve sur l'image, il est coûteux de supprimer ou substituer. Les éléments courbes sont généralement plus courts que les lignes droites, de sorte que leurs coûts d'insertion et de suppression sont aussi faibles que les coûts des petits traits en dessus ou en dessous de la ligne médiane. Les coûts de substitution de '`'' avec '`,' et '`l' avec '`.' sont plus élevés en raison de leur différence visuelle claire.

Symbole	Insertion	Suppression	Substitution
,	1	1	' :2, autre :1
'	1	1	, :2, autre :1
l	2	2	. :2, autre :1
.	1	3	l :3, autre :2
(	1	1	1
)	1	1	1

Tableau 3. *Paramètres pour la distance de Levenshtein.*

Un chemin optimal est ensuite calculé à partir de la matrice de transformation. Lorsque plusieurs trajets ont le même coût, on en choisit un de façon aléatoire. Dans les travaux futurs, nous allons utiliser les informations de l'image pour choisir le meilleur chemin.



(a) Lignes horizontales entre les gradients

(b) Projections en dessus et en dessous de la ligne médiane

(c) Modes des projections et liens le long de la ligne médiane (en noir et blanc)

(d) $L_{vv} > 0$

(e) Points (encadrés en bleu)

(f) Gradients faisant face à la gauche

(g) Courbes vers la gauche (en bleu)

Figure 7. *Les trois étapes de l'extraction de signatures.*

6. Résultats

6.1. *Protocole*

Nous avons testé notre méthode sur la *Queste del Saint Graal*, un corpus exceptionnel et le plus intéressant à être aligné. En tant que chef-d'œuvre de la littérature française, il a déjà été étudié par des philologues et des linguistes et publié en ligne, dans une édition conforme à la TEI, dans un effort d'avant-garde des sciences humaines numériques en France. Le manuscrit en cours d'étude (Lyon, Bibliothèque Municipale, MS. PA 77) date de la première moitié du 13e siècle et est un témoin précoce pour la littérature française ainsi que, sur le plan paléographique, du début

de l'écriture gothique *textualis*. On compte 120 pages contenant deux colonnes et on peut remarquer que la première feuille est déchirée et une partie du texte est manquante (voir fig. 1(a)).

L'alignement est évalué manuellement car nous n'avons pas de vérité terrain. Une interface graphique est fournie pour aider à la validation. Toutes les occurrences d'un mot sont affichées et un simple clic sur une image marque le mot comme rejeté. Un clic et glissement de souris permet de raturer plusieurs mots pour les rejeter. Les mots sont regroupés en fonction de leur signature si bien que les mots mal alignés sont regroupés et peuvent être rejetés rapidement, évitant ainsi des efforts aux utilisateurs experts (voir fig. 8.).

Figure 8. *Interface de validation : groupes de mots au centre, mots rejetés à droite.*

Une propagation automatique de la validation est effectuée. Un mot non-vérifié encadré par deux mots validés est automatiquement validé. Un mot non-vérifié qui suit un mot validé et un mot rejeté est automatiquement rejeté (de même pour un mot non-vérifié précédant un mot rejeté et un mot validé). Cette propagation nous permet de partir de 38% des mots vérifiés pour arriver à 74% du texte validé ou rejeté.

La validation manuelle a pris 35 heures. De plus, nous avons estimé que l'alignement manuel de chaque mot aurait pris plus de 137 heures. La précision sur les mots vérifiés est de 72,9%, ce qui est évidemment moins que les résultats des méthodes basées sur une segmentation de mots trouvées dans la littérature. Toutefois, compte tenu des problèmes que nous avions à traiter (enluminures, l'espacement aléatoire, allographes, abréviations...), ces résultats sont très bons pour une méthode générique applicable à des corpus représentant diverses écritures.

La plupart des erreurs proviennent de lettres majuscules qui ont une forte variation de formes. Les mots commençant (ou précédant un mot commençant) avec un 'r 'ou

un’ f ’ont une précision inférieure, probablement parce que les signatures de ces deux lettres sont trop courtes. Une analyse plus approfondie des statistiques est actuellement effectuée par les paléographes.

Une interface de correction est également fournie (voir fig. 9). Les extrémums des lignes médianes sont affichés (et reliés par une ligne droite) et peuvent être déplacés librement, la ligne est alors automatiquement réalignée. Les limites de chaque mot peuvent également être déplacées à la souris. Si les limites gauche et droite d’un mot sont corrigées manuellement, il est automatiquement validé. Lorsqu’on réaligne une ligne contenant des mots validés, l’alignement est effectué autour des mots validés afin qu’ils puissent agir comme points d’ancrage pour limiter le nombre d’alignements possibles. Chaque correction manuelle des limites des mots est stockée afin de calculer des statistiques. L’état d’un mot (non-vérifié, validé, rejeté) peut également être changé par un clic droit de la souris.

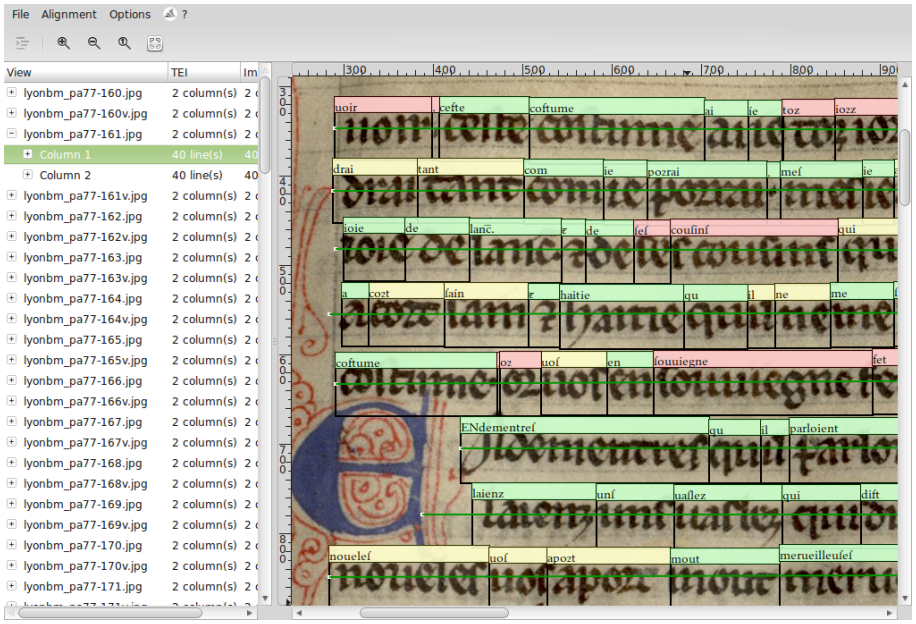


Figure 9. *Interface de correction : toutes les boîtes peuvent être modifiées, validées ou rejetées. Les alignements validés sont en vert, les alignements rejetés sont en rouge et ceux qui n’ont pas encore été vérifiés en jaune.*

7. Conclusion et travaux futurs

Nous avons présenté une nouvelle méthode pour l’alignement texte-image dédiée aux manuscrits médiévaux. Afin de faire face aux difficultés inhérentes de ces documents, nous avons opté pour une approche sans segmentation et sans binarisation.

Nous avons également choisi une approche sans apprentissage afin de faciliter la tâche pour les utilisateurs experts.

L'alignement est basé sur des caractéristiques robustes qui peuvent être extraites à la fois du texte Unicode et des images par calcul différentiel, ce qui assure une réponse fiable même sur des images bruitées. Les résultats sont présentés dans une interface graphique conviviale qui facilite le processus de validation et de correction.

Dans de futurs travaux, nous allons proposer une segmentation en blocs de texte améliorée afin de permettre l'alignement de documents avec des structures physiques complexes et des travaux supplémentaires sur la couleur seront effectués pour permettre un bon alignement des enluminures. Nous mettrons également l'accent sur les allographes et le traitement des abréviations qui sont rarement transcrites en tant que telles.

Remerciements

Ces travaux ont été financés dans le cadre du projet de recherche ORIFLAMMS (ANR 12-CORP-0010 / Cap Digital).

La figure 1(b) est extraite du *Cartulaire blanc de Saint-Denis*, Paris, Archives nationales, LL 1 157 (f. 357r). Toutes les autres figures sont extraites de la *Queste del Saint Graal*, Bibliothèque Municipale de Lyon, ms. PA 77.

8. Bibliographie

Fischer A., Frinken V., Fornés A., Bunke H., « Transcription alignment of Latin manuscripts using hidden Markov models », *Historical Document Imaging and Processing*, ACM, p. 29-36, 2011.

Fischer A., Indermühle E., Bunke H., Viehhauser G., Stolz M., « Ground truth creation for handwriting recognition in historical documents », *Document Analysis Systems*, ACM, p. 3-10, 2010.

Hassner T., Wolf L., Dershowitz N., « OCR-Free Transcript Alignment », *International Conference on Document Analysis and Recognition*, p. 1342-1346, Aug., 2013.

Hobby J., « Matching document images with ground truth », *International Journal on Document Analysis and Recognition*, vol. 1, n° 1, p. 52-61, 1998.

Jawahar C., Kumar A., « Content-level annotation of large collection of printed document images », *International Conference on Document Analysis and Recognition*, vol. 2, IEEE, p. 799-803, 2007.

Kornfield E., Manmatha R., Allan J., « Text alignment with handwritten documents », *International Workshop onDocument Image Analysis for Libraries*, IEEE, p. 195-209, 2004.

Leydier Y., Eglin V., Bres S., Stutzmann D., « Learning-free text-image alignment for medieval manuscripts », *International Conference on Frontiers in Handwriting Recognition (ICFHR)*, p. 363-368, Sep., 2014.

Leydier Y., LeBourgeois F., Emptoz H., ≪ Text search for medieval manuscript images ≫, *Pattern Recognition*, vol. 40, n° 12, p. 3552-3567, 2007.

Leydier Y., Ouji A., LeBourgeois F., Emptoz H., ≪ Towards an omnilingual word retrieval system for ancient manuscripts ≫, *Pattern Recognition*, vol. 42, p. 2089-2105, 2009.

Lorigo L., Govindaraju V., ≪ Transcript mapping for handwritten Arabic documents ≫, *Electronic Imaging*, International Society for Optics and Photonics, p. 65000W-65000W, 2007.

Ramel J.-Y., Rayar F., Sidere N., ≪ Interactive layout analysis, content extraction and transcription of historical printed books using Pattern Redundancy Analysis ≫, *Literary and Linguistic Computing, The Journal of Digital Scholarship in the Humanities*, 2013.

Stamatopoulos N., Louloudis G., Gatos B., ≪ Efficient transcript mapping to ease the creation of document image segmentation ground truth with text-image alignment ≫, *International Conference on Frontiers in Handwriting Recognition (ICFHR)*, IEEE, p. 226-231, 2010.

Tomai C., Zhang B., Govindaraju V., ≪ Transcript mapping for historic handwritten document images ≫, *International Workshop on Frontiers in Handwriting Recognition*, IEEE, p. 413-418, 2002.

Toselli A., Romero V., Vidal E., ≪ Alignment between Text Images and their Transcripts for Handwritten Documents ≫, *Language Technology for Cultural Heritage, from the LaTeCH Workshop Series*, p. 23-37, 2011.

Zinger S., Nerbonne J., Schomaker L., ≪ Text-image alignment for historical handwritten documents ≫, *Document Recognition and Retrieval XVI*, vol. 7247, p. 724703-724703, 2009.