

## Détection de l'inclinaison des documents arabes imprimés

Abdenour Sehad, Lamia Mezai, Mohamed Tayeb Laskri, Mohamed Cheriet

► **To cite this version:**

Abdenour Sehad, Lamia Mezai, Mohamed Tayeb Laskri, Mohamed Cheriet. Détection de l'inclinaison des documents arabes imprimés. Conférence Internationale Francophone sur l'Écrit et le Document (CIFED 04), 2004. <sic\_00001218>

**HAL Id: sic\_00001218**

**[https://archivesic.ccsd.cnrs.fr/sic\\_00001218](https://archivesic.ccsd.cnrs.fr/sic_00001218)**

Submitted on 7 Dec 2004

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Détection de l'inclinaison des documents arabes imprimés

A. Sehad<sup>1</sup>, L. Mezai<sup>1</sup>, M.T. Laskri<sup>2</sup>, M. Cheriet<sup>3</sup>

<sup>1</sup> Division Architecture des Systèmes et MultiMédia  
Centre de Développement des Technologies Avancées Cité 20 Aout, BP 11, Baba Hassen, Alger

<sup>2</sup> Département d'Informatique Université Badji Mokhtar Annaba  
B.P. 12 Annaba 23000 Algérie

<sup>3</sup> Laboratoire d'Imagerie, de Vision et d'Intelligence Artificielle  
École de Technologie Supérieure, Canada

Sehad@cdta.dz, lmezai@cdta.dz, mtlaskri@wissal.dz, mohamed.cheriet@gpa.etsmtl.ca

**Résumé :** Dans cet article, nous présentons une nouvelle méthode de détection de l'inclinaison de documents arabes imprimés. La transformée de Hough est utilisée et adaptée en tolérant un angle d'inclinaison de  $0^\circ$  à  $180^\circ$ . Nous avons considéré les connaissances à priori de l'écriture arabe imprimée qui est cursive par nature. En effet, en exploitant des tracés stables et homogènes, et en les modélisant par le centre de leur boîte englobante, notre approche permet une meilleure approximation de l'inclinaison du texte imprimé, tout en réduisant le temps de calcul. Des expérimentations sur cents documents imprimés et variés montrent des résultats prometteurs avec des temps d'exécutions tolérables dans le contexte d'applications réelles.

**Mots-clés :** Détection de l'inclinaison, documents arabes imprimés, transformée de Hough, détection de liaisons.

## 1 Introduction

L'analyse et la reconnaissance de documents a pour but de convertir un document sous format papier vers un format électronique compréhensible et réutilisable. Le document papier, une fois converti sous forme électronique, permet une recherche par le contenu, un transfert très rapide, un archivage et une gestion beaucoup plus aisée. Par contre, le document sous forme papier permet à l'utilisateur de disposer d'une plus grande autonomie, d'une possibilité de lecture détaillée et plus confortable.

L'un des problèmes le plus fréquent dans un système d'analyse et de reconnaissance de documents est la détermination de l'inclinaison du document. En général, l'inclinaison des images est provoquée essentiellement soit par un mauvais positionnement des pages lors de la saisie optique, soit par une mise en page fantaisiste et irrégulière de l'auteur.

Parmi les techniques de détection de l'angle d'inclinaison les plus utilisées, nous pouvons citer la méthode Trincklin [BEL 92], [ING 99], la méthode de projection [BAG 97], [BEL 92], [BUN 97], [DAN 99], [KAV 02], la transformée de Hough [AMI 96], [BER

98], [HUL 98], [JAI 96], [LE 94], [PER 99], [PAR 96], [YIN 01], [YU 99] et la méthode des k-plus proches voisins [ANT 97], [KWA 02], [OGO 93], [SAF 00], [SEH 00].

La méthode de Trincklin [BEL 92], [ING 99] utilise la méthode des moindres carrés pour évaluer l'inclinaison du texte. Cette méthode est rapide, peu sensible au bruit et appropriée pour de nombreux types de documents incluant des graphiques mais elle nécessite d'avoir des lignes relativement bien justifiées sur la gauche et elle n'est pas appropriée pour les documents multi-colonnes. L'angle d'inclinaison est compris entre  $-10^\circ$  et  $+10^\circ$ .

La méthode de projection [BAG 97], [BEL 92], [BUN 97], [DAN 99], [KAV 02] est basée sur le calcul de l'histogramme horizontal de l'image du document pour chaque angle appartenant à l'intervalle de détection. Cette méthode est facile à implémenter, appropriée pour des documents à structure simple, mais elle n'est pas appropriée pour des documents complexes contenant des graphiques ou des photographies. L'angle d'inclinaison est compris entre  $-10^\circ$  et  $+10^\circ$ .

La transformée de Hough [AMI 01], [BER 98], [HUL 98], [JAI 96], [LE 94], [PER 99], [PAR 96], [YIN 01], [YU 99] est une technique de détection des lignes et des courbes dans une image. Elle est utilisée aussi pour détecter l'angle d'inclinaison avec un intervalle de détection compris entre  $0^\circ$  et  $180^\circ$ . Cette méthode est exacte, robuste et appropriée pour des documents multi-colonnes, mais elle nécessite un espace mémoire important et un temps de traitement prohibitif.

La méthode des k-plus proches voisins [ANT 97], [KWA 02], [OGO 93], [SAF 00], [SEH 00] consiste à calculer l'orientation de la ligne qui relie chaque deux composantes connexes voisines, les orientations sont représentées dans un histogramme. L'angle d'inclinaison du document est déterminé à partir du pic de l'histogramme. Cette méthode est appropriée pour des documents multi-colonnes, elle détecte plusieurs angle d'inclinaison, l'intervalle de détection est illimité par contre le temps de traitement est très important et elle est sensible au bruit et à l'écriture cursive particulièrement les documents arabes..

La transformée de Hough a été testée sur des documents arabes imprimés. Mais, le problème majeur était la lenteur des calculs. Ainsi et afin de palier à ce problème, nous pensons nécessaire d'aviser une méthode de réduction de points à traiter, et ce sans altérer la précision dans la détection de l'angle d'inclinaison ainsi recherché pour la totalité des points. L'utilisation de la méthode de Jain et Yu [JAI 96] ou de Kasturi et Fletcher [TOM 02] donne de bons résultats pour les documents Latins, mais elle n'est pas précise pour les documents arabes, car l'écriture arabe est cursive ainsi les points centraux des composantes connexes ne sont pas alignés sur la même ligne (exemple : pour le sous-mot "لك" et le sous-mot "في", les centres de leurs composantes connexes respectives n'appartiennent pas à la même ligne), cela est dû au fait que les composantes connexes ne sont pas de tailles homogènes. Ce problème ne se pose pas pour les documents Latins lorsque les caractères ne sont pas attachés.

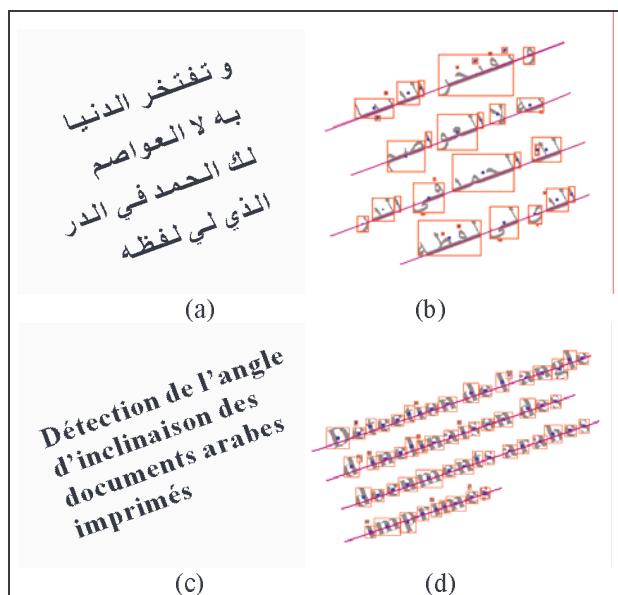


FIG. 1 - (a) Texte arabe incliné, (b, d) Les points centraux des composantes connexes, (c) Texte latin incliné

Dans cet article, nous proposons une nouvelle méthode de réduction de nombre de points à appliquer sur la transformée de Hough et cela afin de diminuer le temps de traitement et de garantir une exactitude dans la détection de l'angle d'inclinaison. Pour cela, nous avons utilisé une caractéristique de l'écriture arabe : les liaisons reliant les caractères arabes car elles ont presque la même hauteur et la même largeur et elles sont alignées sur la même ligne.

## 2 Caractéristiques de l'écriture arabe

L'écriture arabe est une écriture manuscrite, qui s'écrit de droit à gauche. En général les caractères arabes évoluent de façon ascendante ou descendante en prenant des allures minces ou épaisses. Chaque lettre arabe peut s'écrire de 2 à 4 formes différentes :

1. En début du mot (D),
2. Au milieu du mot (M),
3. A la fin du mot mais liée à une lettre (Fl),
4. A la fin du mot sans être liée à la lettre (Fs).

| D | M | Fl | Fs | D  | M  | Fl | Fs |
|---|---|----|----|----|----|----|----|
| أ |   | أ  | أ  | ضد | ضد | ض  | ض  |
| ب | ب | ب  | ب  | ط  | ط  | ط  | ط  |
| ت | ت | ت  | ت  | ظ  | ظ  | ظ  | ظ  |
| ث | ث | ث  | ث  | ع  | ع  | ع  | ع  |
| ج | ج | ج  | ج  | غ  | غ  | غ  | غ  |
| ح | ح | ح  | ح  | ف  | ف  | ف  | ف  |
| خ | خ | خ  | خ  | ق  | ق  | ق  | ق  |
| د |   | د  | د  | ك  | ك  | ك  | ك  |
| ذ |   | ذ  | ذ  | ل  | ل  | ل  | ل  |
| ر |   | ر  | ر  | م  | م  | م  | م  |
| ز |   | ز  | ز  | ن  | ن  | ن  | ن  |
| س | س | س  | س  | هـ | هـ | هـ | هـ |
| ش | ش | ش  | ش  |    |    | و  | و  |
| ص | ص | ص  | ص  | يـ | يـ | يـ | يـ |

TAB. 1 - L'apparence des caractères de l'alphabet arabe.

Les caractères arabes sont toujours liés par des éléments d'épaisseur uniforme, le long d'un texte d'une même police et dont la position correspond au niveau de naissance de tout caractère; c'est le niveau de la ligne de jonction.



FIG. 2 - Représentation de la ligne de jonction

## 3 Détection de l'inclinaison en réduisant le nombre de points

La transformée de Hough permet de détecter avec exactitude l'angle d'inclinaison par contre le temps d'exécution est très important. La réduction du temps de traitement est indispensable pour cela le nombre de points doit être réduit. Pour cela, nous avons utilisé la connaissance à priori de l'écriture arabe cursive, et cela en exploitant des tracés stables et homogènes, à savoir les liaisons qui relient les caractères imprimés, et en les modélisant par le centre de leur boîte englobante, ce choix est justifié par le fait que ces liaisons :

- reposent sur la même ligne
- possèdent approximativement la même hauteur et la même largeur

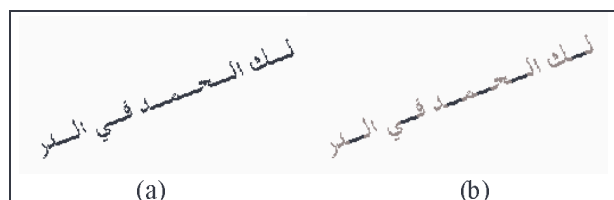


FIG. 3 - (a) Texte incliné, (b) Extraction de liaisons

La détection de liaisons est effectuée suivant les étapes suivantes :

1. Extraction de composantes connexes
2. Filtrage de composantes connexes
3. Extraction de liaisons
4. Filtrage de liaisons
5. Application de la transformée de Hough sur les points centraux des liaisons.

### 3.1 Calcul de composantes connexes

L'extraction des composantes connexes est réalisée au moyen d'un algorithme qui, sélectionne le premier pixel noir et recherche les pixels noirs qui lui sont voisins. Lorsqu'un pixel voisin est trouvé, les coordonnées du rectangle qui englobe ces pixels sont calculées. Pour se déplacer d'un pixel à un pixel voisin, nous avons utilisé un voisinage 8-connectivité. L'image sera toute balayée afin de trouver toutes les composantes connexes. Chaque composante connexe est identifiée par les coordonnées du coin supérieur gauche (x1,y1) et inférieur droit (x2,y2).

### 3.2 Filtrage de composantes connexes

Cette étape consiste à sélectionner les composantes connexes contenant des sous-mots; pour cela la hauteur et la largeur moyenne des composantes connexes sont calculées. Les composantes dont la hauteur (largeur) est supérieure à la hauteur (largeur) moyenne sont retenues les autres composantes sont supprimées (Fig. 4(b)).

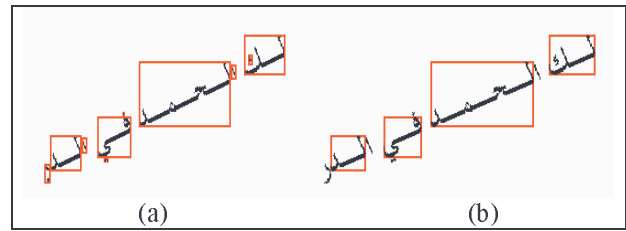


FIG. 4 - (a) Composantes connexes, (b) Filtrage de composantes connexes

### 3.3 Détection de liaisons

C'est l'étape la plus importante dans notre approche, elle consiste à détecter les liaisons qui relient les caractères arabes imprimés. Pour cela, les histogrammes horizontal et vertical sont calculés pour chaque composante connexe. En générale les liaisons sont caractérisées par une hauteur inférieure à la hauteur des caractères et leur fréquence d'apparition est très grande dans un sous-mot. Pour cela, la hauteur (largeur) la plus fréquente est extraite à partir des histogrammes. Ensuite, les liaisons sont extraites, elles correspondent aux zones dont la hauteur (largeur) est inférieure à la hauteur (largeur) la plus fréquente dans l'histogramme vertical (horizontal).

$$HistoV [x] = \sum_{y=c.y1}^{c.y2} Pixel [x][y] \quad [1]$$

$$HistoH [y] = \sum_{x=c.x1}^{c.x2} Pixel [x][y] \quad [2]$$

– 'c' est la composante connexe d'un sous mot.

$$Pixel[x][y] = \begin{cases} 0 & \text{si le pixel est noir} \\ 1 & \text{si le pixel est blanc} \end{cases} \quad [3]$$

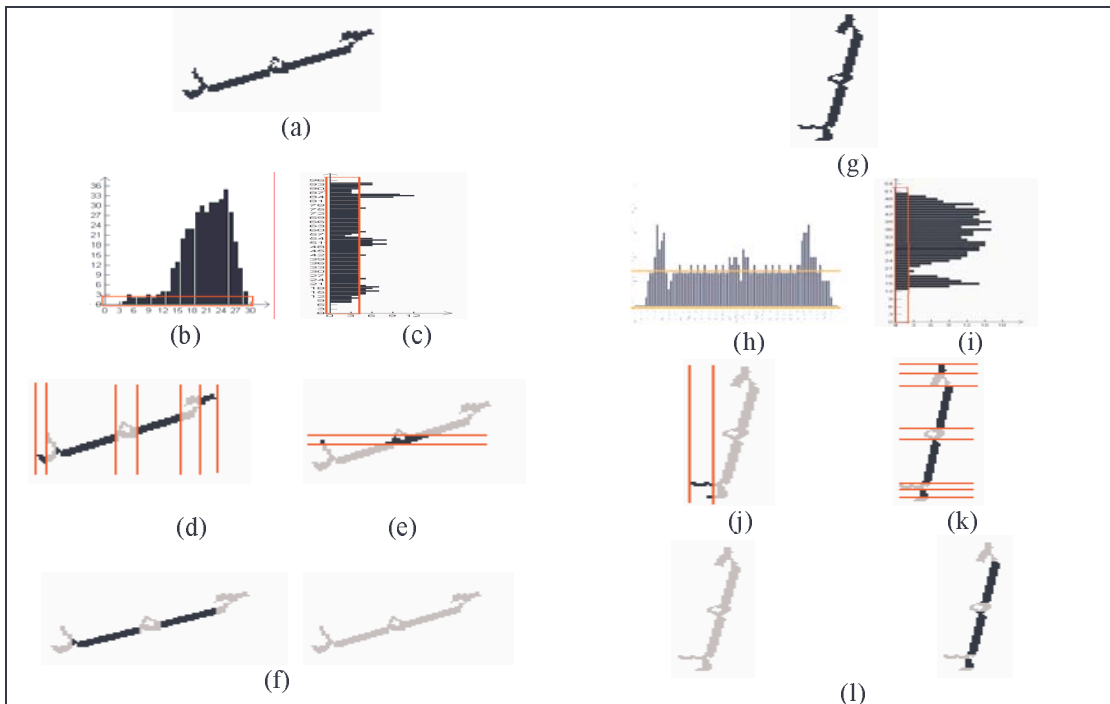


FIG. 5 - (a,g) Sous mot, (b,h) Histogramme vertical, (c,i) Histogramme horizontal, (d,e,j,k) Détection de liaisons, (f,l) Filtrage de liaisons

Etant donné que l'utilisation de l'histogramme vertical ou horizontal dépend de l'orientation du document (Fig. 5) qui doit être retrouvée. Nous avons calculé les deux histogrammes pour détecter les liaisons, ensuite, l'histogramme qui détecte plus de liaisons par rapport à l'autre sera gardé et l'autre sera éliminé.

### 3.4 Filtrage de liaisons

Afin de sélectionner les parties de caractères qui peuvent être considérées comme liaisons, le pré-traitement suivant est alors effectué :

- Elimination des parties détectées au début ou à la fin d'une composante connexe car les liaisons se situent à l'intérieur des sous-mots (voir Fig. 5(d) et Fig. 5(k)).
- Elimination des parties qui se chevauchent verticalement ou horizontalement (voir Fig. 5(e) et Fig. 5(j)).

### 3.5 Détection de l'angle d'inclinaison

Une fois que les liaisons seront filtrées, la transformée de Hough sera appliquée sur les points centraux de liaisons en utilisant la formule suivante :  $\rho = x * \cos\theta + y * \sin\theta$ , avec :  $0^\circ \leq \theta \leq 180^\circ$ . Le vecteur accumulateur est examiné afin d'extraire la valeur maximale d'indices ( $\rho_m, \theta_m$ ),  $\theta_m$  correspond à l'angle d'inclinaison. Finalement, l'algorithme de détection de l'angle d'inclinaison se résume comme suit :

```

Début
  Calculer les composantes connexes
  Filtrer les composantes connexes
  Détecter les liaisons
  Pour chaque liaison faire
  {
    Soit (xc, yc) le point central de la liaison
    Pour 0 <= θ <= 180°
    {
      Calculer ρ = xc cosθ + yc sinθ
      Incrémenter H(ρ, θ)
    }
  }
  Trouver les coordonnées (ρm, θm) de la valeur maximale dans le tableau H(ρ, θ)
  θm Correspond à l'angle d'inclinaison
Fin
  
```

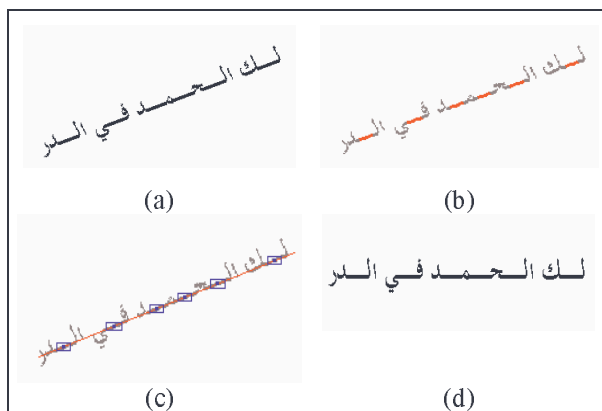


FIG. 6 - (a) Texte incliné, (b) Détection de liaisons, (c) Les points centraux des liaisons, (d) Texte redressé

## 4 Résultats expérimentaux

Afin de tester l'exactitude de l'algorithme de détection proposé dans cet article, nous l'avons appliqué sur cent images de documents arabes imprimés. Ces dernières ont été obtenues en scannant des livres, des lettres et des revues avec un scanner «EPSON EXPRESSION 1600 Pro»; la résolution de numérisation était de 100, 150 et 200 dpi. Les images sont sauvegardées sous format BMP (noir et blanc). Quelques images sont présentées dans la figure Fig. 7.

Les tests ont été établis en effectuant des rotations sur les documents scannés ; l'angle de rotation testé lors des expériences varie entre  $0^\circ$  et  $180^\circ$ . Le déplacement dans l'espace de Hough était de un pour les axes  $\rho$  et  $\theta$  ( $\Delta\rho=1, \Delta\theta=1$ ).

Le temps de calcul est mesuré depuis le début de calcul des composantes connexes jusqu'à la fin de la détection de l'angle d'inclinaison. Les tests ont été effectués sur un PC doté d'un microprocesseur Pentium III 1GHz et une mémoire de 128 Mo sous Windows XP. L'application a été développée avec le C++ Builder 5.

Le tableau Tab. 2 illustre l'efficacité de notre algorithme sur un échantillon de 10 images choisies arbitrairement sur l'ensemble des 100 images traitées. Afin de comparer notre méthode avec la méthode originale de Hough, nous mentionnons dans ce tableau la taille de l'image, la résolution, l'angle estimé par chaque méthode, l'angle réel, le nombre de points traités, ainsi que le temps de calcul. Comme nous pouvons le constater sur ces exemples, que notre algorithme permet de réduire le temps d'exécution d'une manière drastique; ainsi, le temps moyen d'exécution de l'ensemble de 100 images est égal à 12 secondes si nous utilisons les points centraux des liaisons comparé à 30 minutes et 18 secondes dans la cas d'utilisation de tous les points de l'image. En outre, un autre constat de valeur est l'angle détecté par notre méthode qui est égal dans la plupart des cas à l'angle détecté par la transformée de Hough appliquée sur tous les points de l'image; il existe d'autres cas et pour certains types de documents où l'angle détecté par notre méthode est meilleur à celui obtenu par la transformée de Hough.

Nous avons obtenu un taux de précision de 89% pour la détection exacte de l'angle d'inclinaison, ce taux augmente jusqu'à 96% si nous considérons une marge d'erreur qui varie entre  $[-1^\circ, +1^\circ]$  (le taux de précision de l'angle d'inclinaison est égale au nombre de document dont l'angle d'inclinaison a été correctement détecté divisé par le nombre total de documents). Les imprécisions sont dues principalement aux liaisons détectées dans les caractères descendants (ى, ر, ل, ن, ق, ص) situés à la fin des sous mots (Fig. 8) et à la mauvaise qualité de certains documents.

| Image | Taille    | Résolution | Angle estimé |     | Vrai angle | Nombre de points Utilisés par la transformée de Hough |        | Temps d'exécution |                |
|-------|-----------|------------|--------------|-----|------------|---|--------|-------------------|----------------|
|       |           |            | (1)          | (2) |            | (1)   | (2)    | (1)               | (2)            |
| 1     | 714*781   | 100        | 25           | 25  | 25         | 445   | 38627  | 8s                | 4 mn 57 s      |
| 2     | 1237*1429 | 150        | 101          | 99  | 99         | 809   | 124168 | 7 s               | 35 mn 12 s     |
| 3     | 1801*2045 | 200        | 16           | 17  | 16         | 862   | 189136 | 11 s              | 48 mn 40 s     |
| 4     | 1577*1617 | 100        | 127          | 128 | 128        | 586   | 87130  | 6 s               | 22 mn 39 s     |
| 5     | 2149*2149 | 200        | 45           | 45  | 45         | 884   | 158332 | 12 s              | 44 mn 13 s     |
| 6     | 945*909   | 100        | 142          | 143 | 142        | 551   | 43859  | 5 s               | 7 mn 16 s      |
| 7     | 567*590   | 100        | 58           | 58  | 58         | 458   | 28895  | 3 s               | 3 mn 45 s      |
| 8     | 2709*2673 | 200        | 156          | 156 | 156        | 2025  | 344065 | 37 s              | 58 mn 16 s     |
| 9     | 2595*2286 | 200        | 72           | 72  | 72         | 4229  | 372100 | 58 s              | 1 h 30 mn 45 s |
| 10    | 2014*1575 | 150        | 164          | 163 | 164        | 3220  | 279376 | 48 s              | 1 h 22 mn 47 s |

TAB. 2 – Tableau illustrant le temps de traitement

- (1) la transformée de Hough appliqué sur les points centraux des liaisons
- (2) la transformée de Hough appliqué sur tous les points de l'image

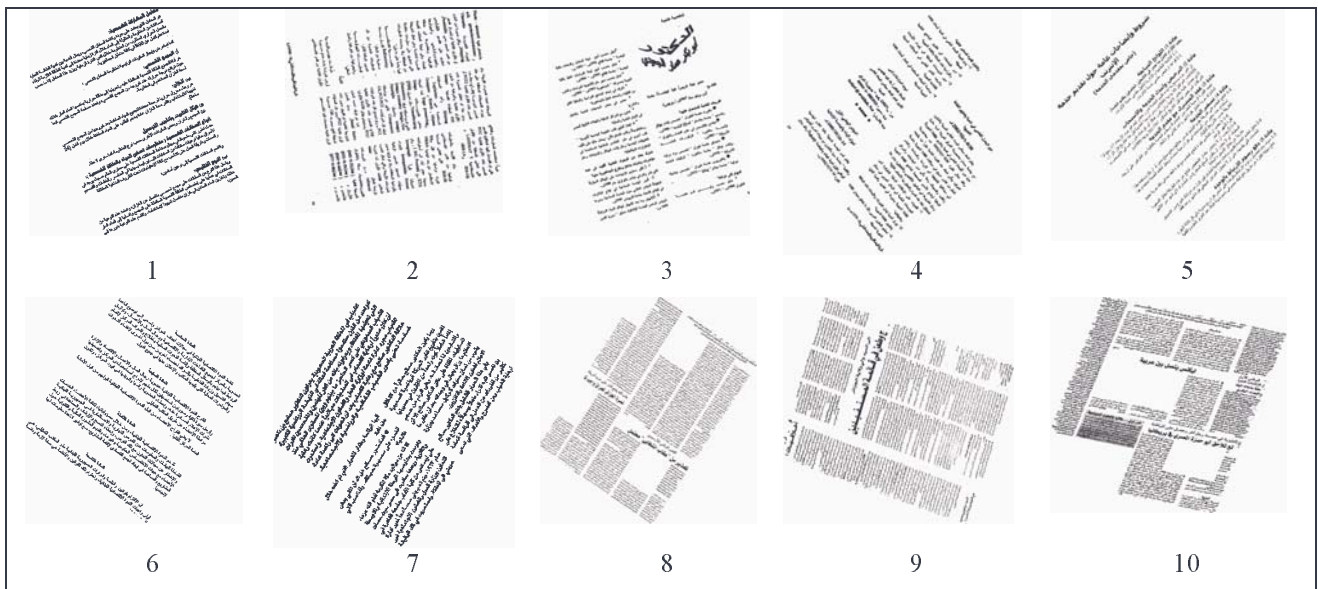


FIG. 7 - Exemple de quelques documents inclinés

## 5 Conclusion

Dans cet article, nous avons présenté une nouvelle méthode pour la détection de l'inclinaison des documents arabes imprimés. Notre algorithme permet de diminuer le temps de traitement en réduisant le nombre de points à appliquer sur la transformée de Hough.

Cette réduction consiste à extraire les éléments reliant les caractères arabes et les modéliser par les centres de leur boîtes englobantes. Ce choix est justifié par le fait que ces éléments possèdent généralement la même largeur et la même hauteur et elles sont alignées sur la même droite. L'extraction des liaisons est effectuée selon plusieurs étapes : (1) calcul des composantes connexes, (2) sélection des composantes connexes qui contiennent des sous-mots, (3) calcul de l'histogramme horizontal et vertical de chaque composante connexe afin d'extraire les liaisons, (4) filtrage des liaisons. Une fois que les

liaisons sont extraites, la transformée de Hough est appliquée sur les points centraux des liaisons afin de trouver l'angle d'inclinaison de l'image du document. Les tests ont montré une bonne performance de l'algorithme. La précision dans l'estimation de l'angle d'inclinaison est très élevée et le temps d'exécution est acceptable. Comme perspective, nous proposons de résoudre les problèmes rencontrés avec les lettres de type "ص ي ر ل ن ق" afin d'obtenir une meilleure exactitude. Pour cela, nous envisageons de rajouter une étape de vérification qui consiste à calculer l'orientation locale des sous-mots qui contiennent plus de 4 liaisons en utilisant la méthode des k-plus proches voisins ainsi les liaisons qui n'appartiennent pas à la direction trouvée seront éliminées (par exemple dans la figure Fig. 8(a) la dernière liaison (situé la fin de la lettre 'س') est éliminée car elle n'appartient pas à la direction retrouvée).

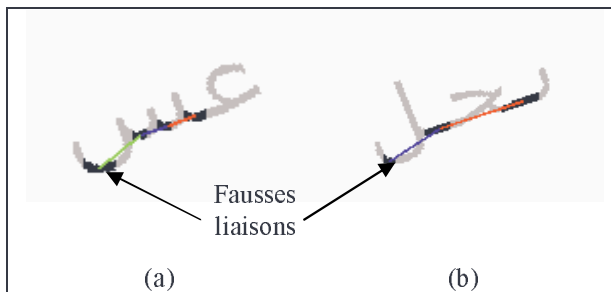


FIG. 8 – Exemple de fausses liaisons

## 6 Références

[AMI 01] AMIN A., SHIU R., “Page Segmentation and Classification Utilizing Bottom-Up Approach”, *International Journal of Image and Graphics*, Vol. 1, No. 2, 2001, pp. 345-361.

[AMI 96] AMIN A., FISCHER S., PARKINSON T., SHIU R., “Fast algorithm for skew detection”, *SPIE Proceedings*, Vol. 2661, 29-30 Janvier 1996.

[ANT 97] ANTONACOPOULOS A., “Local Skew Angle Estimation from Background Space in Text Regions”, *Proceedings of the 4th International Conference on Document Analysis and Recognition*, Germany, August 18-20, 1997, vol. 2, pp. 684-688.

[BAG 97] BAGDANOV A., KANAI J., “Projection Profile Based Skew Estimation Algorithm for JBIG Compressed Images”, *In: Proceedings of the 4th International Conference on Document Analysis and Recognition*, Germany, 1997, p. 401-405.

[BEL 92] BELAID A., BELAID Y., “Reconnaissance des formes : méthodes et application”, InterEdition 1992.

[BER 98] BERGLER S., KHOURY S., SUEN B. C. Y., WAKED B., “Skew Detection, Page Segmentation and Script Classification of Printed Document Images”, *IEEE International Conference on Systems, Man, and Cybernetics*, October 1998, pp. 4470-4475.

[BUN 97] BUNKE H., WANG P. S. P., “*Handbook of character recognition and document analysis*”, Edition World Scientific Publishing, 1997.

[DAN 99] DANCE C., NEWMAN W., TAYLOR A., TAYLOR S., TAYLOR M., ALDHOUS T., “CamWorks: A Video-based Tool for Efficient Capture from Paper Source Documents”, *IEEE International Conference on Multimedia Computing and Systems*, Volume 2, June 07-11, 1999.

[HUL 98] HULL J. J., TAYLOR S. L., “*Document Analysis System IP*”, World Scientific Edition, 1998.

[ING 99] INGLIS S. J., “*Lossless Document Image Compression*”, these de doctorat, Université de Waikato, New Zealand, Mars 1999.

[JAI 96] JAIN A., YU B., “A Robust and Fast Skew Detection Algorithm for Generic Documents”. *Pattern Recognition*, Volume 29 No. 10, October 1996, p. 1599-1629.

[KAV 02] KAVALLIERATOU E., FAKOTAKIS N., KOKKINAKIS G., “Skew angle estimation for printed and handwritten documents using the Wigner-Ville distribution”, *Image and Vision Computing*, 2002, pp. 813-824.

[KWA 02] KWAG H. K., KIM S. H., JEONG S. H., LEE G. S., “Efficient skew estimation and correction algorithm for document images”, *Image and Vision Computing*, 2002, pp. 25-35.

[LE 94] LE D. X., THOMA G., WESCHLE H., “Automated Page Orientation and Skew Angle Detection for Binary Document Images”, *Pattern Recognition*, Volume 27, Number 10, October 1994, pp. 1325-1344.

[OGO 93] O’GORMAN L., “The document Spectrum for page layout analysis”, *IEEE Transactions on Page Layout Analysis and Machine Intelligent*, 1993, p. 1162-1173.

[PAR 96] PARKER J. R., “*Algorithms for image processing and computer vision*”, Wiley, John & Sons, Incorporated, 1996.

[PER 99] PERANTONIS S. J., GATOS B., PAPAMARKOS N., “Block decomposition and segmentation for fast Hough transform evaluation”, *Pattern Recognition*, 1999, p. 811-824.

[SAF 00] SAFABAKHSH R., KHADIVI S. “Document Skew Detection Using Minimum-Area Bounding Rectangle”, *The International Conference on Information Technology: Coding and Computing*, March 27 - 29, 2000, pp. 253-258.

[SEH 00] SEHAD A., HOCINI H., SEHAD M., AMEUR S., “Analyse des documents par la méthode des k plus proches voisins”, *CVA*, 18-20 Novembre 2000, TIZI-OUZOU, Algérie.

[TOM 02] TOMBRE K., TABBONE S., PÉLISSIER L., LAMIROY B., DOSCH P., “Text/Graphics Separation Revisited”, *Proceedings of the 5th International Workshop on Document Analysis Systems*, 2002, pp. 200-211.

[YIN 01] YIN P. Y., “Skew Detection and Block Classification of Printed Documents”, *Image and Vision Computing*, 2001, pp. 567-579.

[YU 99] YU Z., MA H., “An Enhanced Skew angle Estimation Technique for Binary Document Images”, *Fifth International Conference on Document Analysis and Recognition*, September 20-22, 1999, pp. 165-168.