Évaluation sur la création automatique de classes de signatures manuscrites pour l'authentification en ligne

Evaluation of automatic classification of handwritten signatures for online authentication

Nicolas Ragot [1], Julie Fortune [1,2], Paul M'Bongo [1], Nicole Vincent [2], Hubert Cardot [1]

[1] Université François Rabelais Tours, Laboratoire Informatique, 64 av. Jean Portalis, 37200, Tours, France
{nicolas.ragot, julie.fortune, hubert.cardot}@univ-tours.fr
[2] Atos Worldline, 19 rue de la Valle Maillard, 41000 Blois, France
[3] Laboratoire LIPADE, Université Paris Descartes, 45 rue des Saints-Pères, 75270 Paris Cedex 06, France
nicole.vincent@parisdescartes.fr

Manuscrit reçu le 12/02/2009

Résumé et mots clés

Dans ce papier, nous avons cherché à évaluer l'intérêt de la création de classes de signatures manuscrites pour un système d'authentification en ligne. Notre objectif est d'étudier : comment créer automatiquement des classes de signatures (ou styles de signature) ; comment prendre en compte ces classes pendant l'authentification afin de spécialiser le système lors de l'enrôlement d'un utilisateur ; quelles améliorations nous pouvons en attendre. Dans notre étude, la création des classes s'effectue grâce à deux algorithmes de *clustering* et en se basant sur différents sous-espaces de description des signatures. La spécialisation du système consiste à déterminer non plus un seuil de décision (acceptation ou rejet) global au système (*i.e.* le même pour toutes les personnes qui s'enrôleront) mais un seuil adapté à chacune des classes. En termes d'évaluation, nous nous sommes plus particulièrement attachés à étudier l'impact de la classification (en fonction de l'algorithme de classification, du nombre de classes, de l'espace de description) sur les performances d'un système d'authentification basé sur une approche *Coarse To Fine* et l'algorithme *Dynamic Time Warping*. Les résultats expérimentaux sur la base SVC montrent que l'on peut améliorer les performances en diminuant le taux d'erreurs égales de 14,4 %. Cependant la sensibilité de la classification est très grande et la notion de classe unique pour un signataire semble trop restrictive.

Biométrie, signature manuscrite, authentification en ligne, classification non supervisée, DTW.

Abstract and key words

In online handwritten signature verification systems [PLAMONDON 89, LECLERC 94, JAIN 02], adaptation to a specific user or a group of users is a key point since the parameters that determine the acceptation or the rejection of a user are quite sensible. Indeed, biometric data are behavioral and not physical which makes them more variable. Moreover, few learning data are available and the biometric profile of a user can only be determined from a limited number of signature acquisitions. This is why, in the same way as it is done in handwriting recognition, we wanted to study both how classes of handwritten signatures can be automatically determined and what is the impact of these classes on a verification system.

The verification system, on which this study is based, is not new itself. It is a Coarse To Fine approach that uses the Dynamic Time Warping algorithm (DTW) [WIROTIUS 05a]. This is a ligth system that was specifically designed to be

embedded on smartphones or PDAs. During the enrolment step, 5 reference signatures are acquired from the user. One should notice here that the system forces the user to provide 5 signatures that are roughly of the same style (with similar total length and total duration). This is coherent with the database we used for our experiments and with the classification methods employed in our study. After the preprocessing (normalization – translation, rotation and scale – and removal of the unnecessary points: only the points with minimal local speed are kept [WIROTIUS 05b]), these signatures represent the biometric profile of the user. The authentication itself is based on the Coarse To Fine approach. In the coarse step, signatures that are too dissimilar from the profile are rejected. This assessment is peformed from a simple comparison based on the total length and the total duration of the signatures. For the signatures that are not rejected at this first step, a finer comparison is performed using the Dynamic Time Warping algorithm. The final acceptation or rejection of a signature depends on a threshold μ that is learnt from an experimental database to reach the equal error rate (EER).

The main drawback of the previous system, considering its parameters, is the threshold μ. This is the reason why we tried to find an automatic mechanism to determine classes of signatures and to adapt the system (μ) to these classes. In fact, we used clustering algorithms and we studied the impact on the performances of the system of: the number of classes; the clustering feature space; the algorithm used.

The process to determine the classes consists in using a clustering algorithm on a learning dataset that contains the biometric profiles of several signers. Then, for each of the k classes obtained, we find a local threshold μ_k that enables to obtain the EER on the class k. Next, during the enrolment of a new signer, the class to which the signer belongs is considered to be the one in which most of the signatures of his profile are (this is why it is better that all signatures of the profile should be of the same style). The threshold that will be used during the authentication step is the one corresponding to that class.

The experiments were performed using the SVC database using the leave-one-out protocol to learn the thresholds and to test the system on different data. Without using the classes of signatures, the system can achieve 1.94% of EER on this dataset.[1] We next conducted several experiments. Firstly, we used the K-means algorithm with several values for K and several feature subspaces based on the global characteristics of the signatures (static and dynamic). The results show that the EER can be decreased to 1.84% for some small values of K. For most of the feature subspaces used, over $K = 3$, the results are not interesting (equal or worse to those obtained with the original system). Moreover, the results do not seem to be stable depending on the value of K. In fact, observing the result of the clustering, it seems that for too many signers, the different signatures in their profile end up in different classes, not in a unique one, which indicates that the clustering is not stable, despite our constraint on the signatures of the profile. The best results – considering the compromise between the EER performance and the stability of the clustering, i.e. the standard deviation of FAR and FRR – are obtained using the two principal axes (using a principal component analysis) and 2 classes. The second experiment was equivalent but using the fuzzy C-means instead of the K-means. The results show a great improvement in terms of stability of the clustering and a general improvement of the EER for small values of K. The best result was obtained using two classes and again the two principal axis. In this case, the EER fall to 1.66% which represents an improvement of 14.4%. Nevertheless, deeper studies of the results of the clustering show that we are still unable to find a classification that is valid for all signers: there are still signatures of a same signer that belong to different classes. This phenomenom of course increases with higher values of K. Then our perspectives are either to consider that a signer could belong to several classes (even if he always signs with the same style) or to operate several classifications, using different feature spaces, and to choose for a signer the most pertinent classification.

Biometrics, handwritten signature, online authentication, clustering, Dynamic Time Warping.

Remerciements

Les auteurs souhaitent remercier la société ATOS Worldline et plus partiulièrement Jean-Claude Barbezange, pour le soutien de ces travaux.

1. In fact, an improved version of the DTW enables this system to reach 1.46% of EER. It is not used in this study since this system uses much more ressources, which makes it more difficult to embed it onto small devices.

1. Introduction

La problématique de l'authentification par signature manuscrite [PLAMONDON 89, LECLERC 94, JAIN 02] occupe une place tout à fait particulière dans le domaine de la reconnaissance de formes. En effet, au contraire de l'identification, il s'agit d'un problème de reconnaissance « à une classe », pour lequel on dispose de peu de données d'apprentissage. De plus, les données biométriques utilisées, les signatures, sont qualifiées de comportementales et sont donc par essence extrêmement sujettes à la variabilité, à la fois entre différentes personnes mais également pour une même personne [RAGOT 08]. Pour l'ensemble de ces raisons, les paramètres du système qui déterminent l'acceptation ou le rejet d'une personne sont particulièrement sensibles et dépendants de la personne qui cherche à s'authentifier. Dans un cadre applicatif réel, où les données d'apprentissage sont peu nombreuses et ne représentent qu'un faible échantillon de la variété des signatures manuscrites, les systèmes d'authentification peuvent donc souffrir d'un manque d'adaptation à l'utilisateur ou à un groupe d'utilisateurs ayant des spécificités communes.

Dans ce contexte, il semble intéressant de se rapprocher des recherches effectuées dans la communauté de l'écrit, notamment autour de l'écriture manuscrite, et qui ont montré qu'il existe différents types de scripteurs [CRETTEZ 95, NOSARY 99, SEROPIAN 02, BENSEFIA 05, SIDDIQI 07]. Nous pouvons alors faire l'hypothèse raisonnable qu'il existe également différents types – ou classes – de signataires (ou de signatures) et que l'on va pouvoir adapter les paramètres du système d'authentification, pour un signataire, en fonction de la classe de signatures à laquelle il appartient et ainsi améliorer les performances du système.

Il existe des travaux, notamment en graphologie [SEDEYN 02] qui devraient permettre de déterminer les spécificités des signatures manuscrites. Nous devrions donc également pouvoir en déduire des classes de signatures, partageant certaines de ces spécificités. Cependant, cette méthode reposant sur les connaissances d'un expert n'est pas sans poser de problèmes. En effet, elle contraint à l'utilisation de descripteurs les plus proches possibles de ceux utilisés par l'expert, alors même que ces descripteurs ne sont pas nécessairement les plus adaptés pour le classificateur employé (problème courant en sélection de caractéristiques). De plus, cette connaissance peut être coûteuse à modéliser et à extraire du signal, ce qui n'est pas forcément compatible avec l'usage de terminaux d'authentification légers – souvent utilisés en biométrie. Enfin, ces experts en graphologie sont souvent spécialisés sur l'étude de signatures d'une origine géographique bien précise (française, anglo-saxone, asiatique, etc.) rendant l'expertise inutilisable en dehors de son domaine de compétence. Il semble donc bien plus avantageux d'essayer de déterminer automatiquement les classes de signatures à partir des échantillons biométriques disponibles. Rien n'empêchera par la suite d'enrichir ou de modifier cette classification si de nouvelles données sont disponibles.

Dans cette étude, nous nous sommes donc penchés sur l'utilisation d'une classification non supervisée pour créer automatiquement des classes de signatures. Nous avons plus particulièrement étudié l'impact de la classification résultante sur les performances d'un prototype d'authentification léger – *i.e.* conçu pour être embarqué sur un smartphone ou un PDA et nécessitant peu de ressources – basé sur l'algorithme *Dynamic Time Warping* (DTW) [JAIN 02, OHISHI 00]. L'impact de la classification a notamment été étudié en fonction du nombre de classes obtenu, de l'espace de représentation utilisé, ainsi que de la méthode de classification employée.

Dans la suite, la section 2 décrit le système d'authentification sur lequel nous nous sommes appuyés pour conduire notre étude. La section 3 présente le processus utilisé pour déterminer automatiquement des classes de signatures et comment celles-ci sont employées pour améliorer les performances du système d'authentification. Enfin, les résultats expérimentaux sont analysés dans la section 4.

2. Système d'authentification

Notre étude se base sur l'utilisation d'un système d'authentification léger conçu lors de précédents travaux [WIROTIUS 05a, WIROTIUS 05b]. Celui-ci fonctionne en deux étapes. Lors de la première, la phase d'enrôlement (*cf.* section 2.2), la personne à authentifier s'enregistre en fournissant quelques signatures pour que le système crée son profil (ensemble de signatures de référence). La deuxième étape (*cf.* section 2.3) est l'authentification à proprement parler : une personne – la personne préalablement enregistrée ou une autre qui attaque le système – soumet une signature qui est authentifiée ou rejetée. Dans tous les cas, le mécanisme d'acquisition des signatures et les prétraitements effectués à chaque étape sont identiques (*cf.* section 2.1).

2.1 Acquisition des signatures et prétraitements

L'acquisition des signatures s'effectue sur un TabletPC, une tablette graphique ou un PDA. Afin de conserver le maximum de compatibilité, seules les coordonnées de chacun des points de la signature ainsi que les posés et levés de stylet sont conservés. Ensuite, nous procédons à une normalisation des signatures. Cette normalisation consiste en :

- une rotation pour rendre l'axe d'inertie horizontal [LEJTMAN 01, WIROTIUS 05][2] ;

2. Le choix de ce prétraitement est discutable dans la mesure où l'on peut considérer que certains signataires ont justement comme spécificité de ne pas signer horizontalement. Cependant, cela ne peut être réellement pris en compte que lorsque le signataire est parfaitement à son aise pour signer et que la zone de signature est parfaitement repérée et orientée (par un rectangle par exemple). En réalité, les signatures s'effectuent très souvent dans

- une homothétie pour que toutes les signatures aient la même taille tout en conservant leurs proportions ;
- une translation pour centrer la signature par rapport au repère.

La figure 2 illustre le résultat de cette normalisation sur la signature de la figure 1.

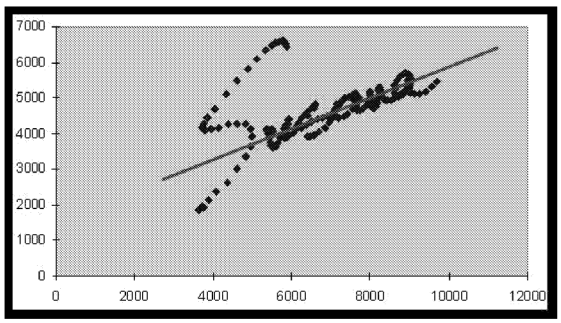

Figure 1. Signature originale avec son axe d'inertie.

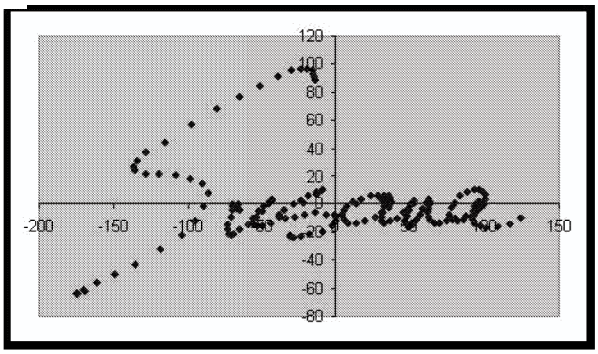

Figure 2. Signature normalisée.

Le dernier prétraitement effectué consiste à ne conserver que certains points significatifs de la signature. L'intérêt est ici multiple. Cela permet de réduire la taille de la signature et donc d'accélérer les traitements ultérieurs. Cela permet également de diminuer l'espace de stockage nécessaire pour le profil. Ces deux propriétés sont particulièrement importantes pour une application devant fonctionner dans un cadre réel d'utilisation. En effet, la CNIL interdit les bases de données biométriques. Il faut donc stocker les données biométriques de l'utilisateur sur une carte à puce personnelle et, pour assurer la sécurité du système, aucune information ne doit pouvoir être récupérée de cette carte. L'authentification s'effectue donc également sur cette carte. Ainsi le système est plus difficilement attaquable, la contrepartie étant l'utilisation de ressources système limitées[3]. Un deuxième avantage, et non des moindres, à la sélection de points est que si ceux-ci sont bien choisis, les performances du système peuvent être accrues car on fait ainsi disparaître le bruit qui pouvait se trouver dans les données brutes. Dans notre cas, après des études comparatives sur plusieurs méthodes de sélection de points, nous avons choisi de ne conserver que les points de vitesse minimale [WIROTIUS 05b], la vitesse en chaque point Pt étant calculée par :

$$vi = \frac{dist(Pt_i, Pt_{i+1})}{t_{i+1} - t_i},$$

où $dist(Pt_i, Pt_{i+1})$ est la distance euclidienne entre deux points successifs effectués aux instants t_i et t_{i+1}.

2.2 Enrôlement

Cette étape a pour objectif de créer le profil d'un utilisateur. Après une phase d'entraînement, nécessaire à la prise en main du périphérique d'acquisition, l'utilisateur fournit 5 signatures au système. En cas de problème lors de l'acquisition, l'utilisateur peut annuler une saisie et la recommencer. Le système effectue également une vérification pour éviter tout problème lors de la création du profil. Cette vérification consiste à demander 5 nouvelles signatures. Si parmi ces 5 signatures de vérification l'une d'entre elles est « trop » dissemblable des 5 signatures de référence, l'utilisateur doit recommencer la procédure d'enrôlement. La dissemblance entre 2 signatures est évaluée en considérant uniquement la durée totale et la longueur totale de la signature, exactement de la même façon que lors de la première étape de notre processus d'authentification (*cf.* section 2.3.1). Ce processus permet ainsi de constituer le profil du signataire avec seulement 5 signatures pour lesquelles nous sommes assurés d'avoir un minimum de similarité.

Ce choix peut paraître discutable car nous limitons ainsi la possibilité que pourrait avoir un signataire de signer de plusieurs façons différentes. Cependant, il faut bien voir que cette vérification ne contraint en fait que l'utilisation de signatures très différentes en terme de temps d'exécution ou de longueur. Les variations légères sur ces critères ou importantes selon d'autres caractéristiques restent tout à fait possibles. De plus, ce mode de fonctionnement parait assez logique étant donné l'usage. En effet, lors d'un processus d'authentification, un signataire effectue ses 5 signatures les unes à la suite des autres, presque sans pose, et naturellement il devrait donc être assez peu enclin à utiliser des signatures radicalement différentes. Autre point, si l'on considère que 5 signatures sont nécessaires pour appréhender la variablité d'une signature (et 5 signatures reste une quantité assez faible), il serait plus judicieux, si l'on souhaite prendre en compte plusieurs styles de signatures pour une même personne, de demander 5 exemplaires de chacun de ses styles de signatures. Finalement, cette particularité de notre méthode d'enrôlement semble également être en adéquation avec la façon dont les bases d'évaluations sont constituées (bien que ce ne soit pas toujours explicite) : souvent des faux sont établis et ils ne le sont vraisemblablement qu'à partir de l'observation d'une seule signature de référence. Nous insistons ici particulièrement sur ce point car il ne sera bien entendu pas sans importance pour la création des classes de signatures (*cf.* section 3.1).

des conditions variables – position debout, signature effectuée « à la vavite », support pas tout à fait bien positionné, etc. – et l'utilisation d'un repère rectangulaire pour la saisie de la signature reste assez spécifique aux signatures occidentales. Il semble donc plus judicieux de ne pas prendre en compte cette orientation de la signature dans le processus d'authentification.

3. Il s'agit là d'un point qui distingue notre système d'un certain nombre d'autres systèmes commerciaux.

Une fois ces 5 signatures valides acquises, elles subissent les prétraitements décrits ci-dessus (*cf.* section 2.1) puis sont conservées (sur une carte à puce par exemple) comme signatures de référence en attente d'une future authentification.

2.3 Authentification

Lors de cette phase, un utilisateur souhaitant s'authentifier signe sur le périphérique d'acquisition. La signature test ainsi obtenue, après avoir subi les prétraitements décrits section 2.1, va être comparée aux signatures de référence du profil biométrique de la personne qu'il prétend être. Pour cela, la comparaison s'effectue en deux étapes, selon une approche *Coarse To Fine* [WIROTIUS 05a, WIROTIUS 05b]. L'approche *Coarse* (*cf.* section 2.3.1) permet d'éliminer directement les signatures très différentes des signatures de référence. Ces attaques supposées seront donc rejetées sans passer par l'étape *Fine* (*cf.* section 2.3.2) qui effectue une comparaison plus précise mais aussi plus coûteuse en temps de calcul. Ce principe permet donc d'accélérer les traitements lors de l'authentification, l'étape *Fine* n'étant utilisée que si nécessaire.

2.3.1 Étape *Coarse*

Cette première étape élimine une signature trop dissemblable des signatures de référence. Elle se base pour cela sur une comparaison des signatures représentées par des caractéristiques globales et stables [WIROTIUS 05b] : la longueur totale (C^1) et la durée totale (C^2) de la signature. Une signature test est acceptée – et l'étape *Fine* sera utilisée – si la condition suivante est vérifiée pour les deux caractéristiques C^j $(j = 1,2)$:

$$\delta \times min_i(C_i^j) \leqslant C^j \leqslant \lambda \times max_i(C_i^j) \qquad (1)$$

avec C^j la valeur de la caractéristique pour la signature test, C_i^j la valeur de cette même caractéristique pour la signature de référence i $(i = 1,...,5)$, et des coefficients λ et δ fixés expérimentalement à 0,6 et 1,4 respectivement.[4]

2.3.2 Étape *Fine*

Cette seconde étape effectue une nouvelle comparaison entre la signature testée et les 5 signatures de référence, mais cette fois en utilisant une mesure de distance plus précise. Dans notre cas, cette distance est calculée en mettant en correspondance les points des deux signatures par une variante de l'algorithme *Dynamic Time Warping* (DTW) [JAIN 02, OHISHI 00] adaptée à notre problème [WIROTIUS 05a, WIROTIUS 05b]. Plusieurs métriques peuvent être associées au DTW :

- la distance spatiale ($distS$) – distance usuelle – qui correspond à la distance euclidienne entre deux points P et P' mis en correspondance et projetés dans le même repère ;
- la distance temporelle ($distT$) qui correspond à l'écart de temps entre les points P et P' et qui est calculée par :

$$distT(P,P') = \left|t(P') - t(P)\right|, \qquad (2)$$

où $t(P)$ est l'instant auquel P a été fait ($t(P1) = t(P'1)$) ;
- la distance curviligne ($distC$) qui correspond à la différence de distance parcourue entre le début des signatures et les points P_i et P'_i :

$$distC(P_i,P'_j) = \left|\frac{long(P_i)}{long(P_n)} - \frac{long(P'_j)}{long(P'_m)}\right|, \qquad (3)$$

avec $long(P_i) = \sum_{k=2}^{i} dist(P_k,P_{k-1})$, $dist(P_k,P_{k-1})$ la distance euclidienne entre deux points successifs de la signature, P_n et P'_m le dernier point de chacune des signatures considérées ($long(P_n)$ correspond donc à la longueur totale de la signature). Pour notre prototype d'authentification destiné à être embarqué, nous utilisons la distance spatiale ($distS$) seule qui donne le meilleur rapport entre performances et temps de calcul. Lorsque nous souhaitons avoir les meilleures performances, nous utilisons la combinaison ($distF$) des trois distances. Dans ce dernier cas, la combinaison s'effectue de la façon suivante :

$$distF = \alpha \times distS + \beta \times distT + \gamma \times distC, \qquad (4)$$

où α, β, γ sont des coefficients compris entre 0 et 1 et dont la somme vaut 1. Lors d'expérimentations précédentes sur une base privée [WIROTIUS 05b], les meilleures performances ont été obtenues avec les valeurs suivantes : $\alpha = 0,2$, $\beta = 0,6$, $\gamma = 0,2$ (*cf.* section 4.2).
Pour qu'une signature test soit acceptée et le signataire authentifié, il faut que la distance entre celle-ci et au moins une des signatures de référence soit inférieure au seuil μ. Ce seuil global – indépendant du signataire – est déterminé lors d'une phase préalable d'apprentissage du système. Pendant celle-ci, une base de validation est utilisée pour simuler le processus d'authentification et obtenir les performances du système : Taux de Faux Acceptés (TFA, ou *False Acceptance Rate* – FAR) et Taux de Faux Rejet (TFR ou *False Rejection Rate* – FRR, *i.e.* le taux de signatures authentiques rejetées). Différentes valeurs de seuils sont ensuite testées de façon quasi exhaustive (par incrément fixe) jusqu'à obtenir le Taux d'Erreurs Egales (TEE, ou *Equal Error Rate* – EER) sur cette base de validation (*cf.* figure 3).

4. Ces coefficients ont été déterminés lors d'une étude précédente à partir d'expérimentations sur une base de signatures privée [WIROTIUS 05b].

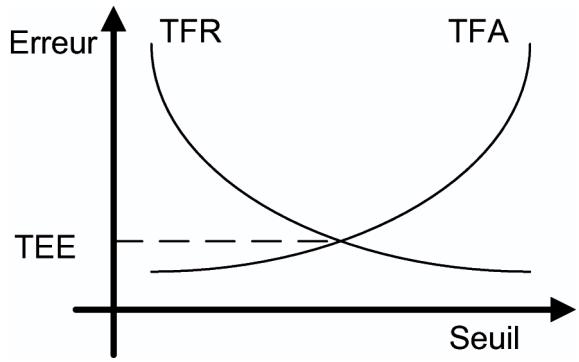

Figure 3. Définition du TEE.

3. Création automatique de classes de signatures

Un des inconvénients du système précédent réside essentiellement dans l'utilisation du seuil de décision global μ qui est unique quels que soient les utilisateurs du système. C'est pourquoi nous avons souhaité étudier l'impact, sur les performances du système, de la création de classes de signatures et de la spécialisation du système à ces classes. Nous avons alors deux sous-problèmes : trouver des classes de signatures ; adapter le système à ces classes. Ces deux points sont développés respectivement dans les sections 3.1 et 3.2. Enfin, la section 3.3 décrit le nouveau fonctionnement du système lors de l'authentification.

3.1 Création de classes de signatures

L'important pour la création de classes de signatures est avant tout d'arriver à déterminer une classification qui soit stable pour chacun des utilisateurs du système. Idéalement, on pourrait souhaiter qu'un signataire n'appartienne qu'à une seule classe, c'est-à-dire que toutes ses signatures (de référence et à authentifier) appartiennent à la même classe. Cette hypothèse peut sembler assez forte et limitante. Cependant, elle reste tout à fait raisonnable étant donné notre système d'authentification et le mécanisme d'enrôlement utilisé (*cf.* section 2.2). De plus, nos résultats expérimentaux (cf. section 4) mettent en évidence que la notion de classe de signatures reste difficile à modéliser et exploiter même avec cette simplification et les précautions prises lors de l'enrôlement.

L'espace de représentation utilisé pour l'extraction des classes doit permettre de se rapprocher du principe de stabilité tel qu'il est énoncé ci-dessus. Pour cela, nous avons choisi dans un premier temps de travailler sur des caractéristiques globales. Parmi celles-ci, la durée totale de la signature ainsi que sa longueur totale sont reconnues comme étant des caractéristiques stables pour un signataire (*cf.* étape *Coarse*, section 2.3.1). Cependant, pour aller plus loin, nous avons également travaillé sur un ensemble plus vaste de 12 caractéristiques, décrivant la forme et la dynamique des signatures. Ces caractéristiques sont :

- la longueur totale (longueur Totale) ;
- le rapport entre les déplacements vers la gauche et vers la droite (rapDepHor) ;
- le rapport entre les déplacements vers le haut et vers le bas (rapDepVer) ;
- le rapport entre les déplacements horizontaux et verticaux (rapDepXY) ;
- la distance entre le premier et le dernier point (distPremDer) ;
- le déplacement horizontal moyen (depHorMoy) ;
- le déplacement vertical moyen (depVerMoy) ;
- l'angle entre l'horizontale et la droite joignant le premier et le dernier point (angleSign) ;
- la durée totale (tpsTotal) ;
- l'accélération moyenne (accMoy) ;
- la vitesse moyenne verticale (vtsMoyVert) ;
- la vitesse moyenne horitzontale (vtsMoyHor).

Afin de déterminer les corrélations entre ces caractéristiques nous avons effectué une analyse en composantes principales (ACP). Le cercle des corrélations correspondant est donné dans la figure 4. Sur celle-ci on constate que la longueur totale et la durée totale restent assez fortement corrélés. On se doute alors que l'algorithme de classification ne pourra s'appuyer que sur un pouvoir de représentation de l'espace de description assez limité.

À partir de cet espace de représentation, et pour obtenir les classes de signatures, nous avons utilisé deux algorithmes classiques qui ont démontré leur utilité dans ce genre d'études (préclassification avant reconnaissance) : l'algorithme des *K-means* [MCQUEEN 67] et celui des *Fuzzy C-means* réputé moins sen-

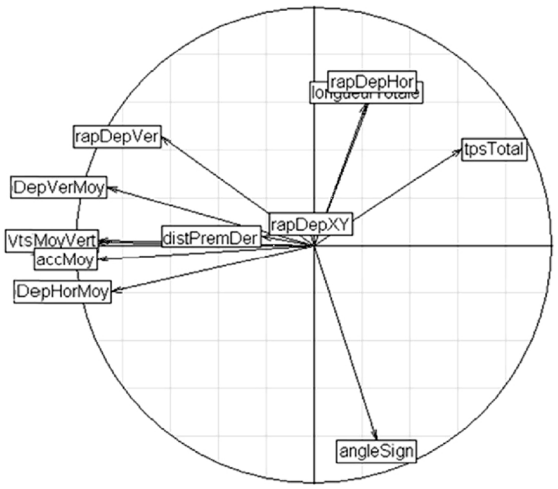

Figure 4. Corrélation entre les caractéristiques testées pour la création de classes de signatures.

sible aux variations dans les données et aux données abérrantes [BEZDEK 81]. Ces algorithmes sont employés sur l'ensemble des signatures de référence dont on dispose lors de la phase d'apprentissage pour une valeur de K (nombre de classes à trouver) choisie initialement.

3.2 Optimisation du système grâce aux classes de signatures

Une fois les classes de signatures obtenues à partir des signatures de référence de la base d'apprentissage, nous utilisons les signatures de la base de validation pour définir un seuil de décision μ_k pour chacune de ces classes k ($k = 1,...,K$). Le principe est le même que pour le système initial (*cf.* section 2.3.2) sauf que nous cherchons les seuils qui permettent d'obtenir le TEE sur chaque classe – et non le TEE global. Ainsi, chaque signature de la base de validation, est testée par rapport à chaque signataire (enrôlé) de la base d'apprentissage. Nous définissons la classe d'un signataire enrôlé comme étant celle dans laquelle se retrouve la majorité de ses signatures de référence (appartenance stricte).

Lorsqu'une signature test est donc comparée au profil d'un signataire (à ses 5 signatures de référence), le seuil μ_k correspondant à la classe de ce signataire est utilisé pour accepter ou refuser la signature test lors de l'étape *Fine*. Le TFA ou le TFR de la classe sont alors mis à jour en conséquence. Lorsque toute la base de validation a été utilisée, les seuils sont modifiés comme indiqué dans la section 2.3.2 puis une nouvelle phase de validation est effectuée jusqu'à obtenir le TEE sur chacune des classes.

3.3 Authentification à partir de classes de signatures

Une fois l'apprentissage des μ_k effectué, le prototype peut être utilisé dans le mode de fonctionnement « réel » : lors de la phase d'enrôlement, le profil du signataire est toujours constitué de ses 5 signatures de référence et le système utilisera comme seuil de décision μ_k pour les futures authentifications (ou attaques) celui correspondant à la classe du signataire qui s'est enrôlé.

4. Résultats expérimentaux

4.1 Protocole de test

Pour évaluer l'impact de la création de classes sur notre système d'authentification par signature manuscrite en ligne, nous avons utilisé la base SVC (*Signature Verification Competition*) [SVC 04, YEUNG 04]. Cette base est constituée de 40 signataires. Pour chaque signataire, 20 signatures sont authentiques et 20 autres sont des faux expérimentés, réalisés par des personnes ayant accès à une vidéo de la personne en train de signer. Dans notre travail d'expérimentation, nous ne nous intéressons pour l'instant qu'aux performances sur des faux aléatoires[5] et nous n'utilisons donc que 40×20 signatures. Nous sommes également conscients que cette base a une taille limitée. Cependant, il s'agit de l'une des rares bases disponibles et utilisées comme benchmark de référence. Il faut également noter que nous n'avons pas de connaissances *a priori* sur la représentativité de cette base quant aux différents styles de signatures et donc quant au nombre de classes. En effet, l'origine des contributeurs n'est pas connue. De plus, ceux-ci n'ont pas signé avec leur signature habituelle mais en ont inventé une pour l'occasion.[6]

Nous effectuons nos tests en *leave-one-out* (LOO). 39 signataires sont utilisés pour l'apprentissage et la validation : les cinq premières signatures de chacun des 39 signataires sont utilisées comme signatures de référence afin de simuler la phase d'enrôlement et effectuer la recherche de classes ; les 15×39 signatures restantes sont utilisées comme base de validation pour rechercher les seuils. Enfin, le dernier signataire est utilisé lors de la phase de test pour évaluer les performances du système en phase de généralisation : les 5 premières signatures du signataire simulent l'enrôlement de ce dernier ; les 15 suivantes servent pour évaluer le TFR ; enfin, les $15 * 39$ signatures de la base de validation[7] sont utilisées comme attaques aléatoires et permettent d'obtenir le TFA. Cette phase d'apprentissage/test est répétée de façon à ce que chaque signataire serve exactement une fois pour la phase de test ce qui permet d'obtenir les TFA et TFR moyens du système.

4.2 Résultats du système initial

Le tableau 1 donne les TFA et TFR moyens obtenus en LOO sur la base SVC avec notre système initial, c'est-à-dire sans classe de signatures. Les résultats sont présentés pour chacune des métriques associées au DTW, ainsi que pour la meilleure combinaison de celles-ci (paramètres obtenus sur une base privée, *i.e.* avec les valeurs présentées dans la section 2.3.2). Il s'agit donc de notre référence pour évaluer l'impact de l'utilisation de

5. Les faux expérimentés semblent moins intéresser les industriels qui estiment que, dans la réalité, il existe peu de chances que quelqu'un fasse un faux expérimenté en ligne (les faussaires auraient besoin d'une vidéo précise du signataire en train de signer, ce qui est difficilement réalisable en pratique).

6. Là encore, tout cela laisse penser que chaque signataire ne signe qu'avec un seul type de signature, ce qui corrobore les hypothèses faites dans la sections 2.2 et garantit la cohérence de notre approche pour ce cas d'étude.

7. Bien que ces signatures aient servi pour déterminer les seuils en phase de validation, elles n'ont pas servi par rapport au signataire testé en phase de généralisation. On peut donc les considérer comme des faux aléatoires quelconques et le biais éventuel reste ainsi très limité.

classes de signatures. Le TEE correspondant est obtenu en prenant la moyenne du TFA et du TFR.[8]

Tableau 1. Performances du système initial (en %).

	TFA	TFR	TEE
distS	1,88	2,00	1,94
distT	4,00	3,67	3,84
distC	12,90	11,50	12,20
distF	1,45	1,47	1,46

Comme annoncé dans la section 2.3.2, la métrique seule qui offre les meilleures performances est la distance spatiale. La combinaison des trois métriques permet quant à elle d'améliorer assez sensiblement le TEE, malgré les faibles performances obtenues avec les distances temporelle et curviligne.

Le positionnement de notre approche par rapport aux autres systèmes ne figure pas dans ce tableau puisque notre objectif est avant tout d'étudier l'apport de la création de classes de signatures sur notre système. De plus, une telle comparaison de différentes approches reste particulièrement délicate car beaucoup utilisent leur propre base de signatures et les protocoles expérimentaux diffèrent souvent de façon assez significative (nombre de signatures lors de l'enrôlement, pas de leave-one-out, etc.). Ainsi, même sur la base SVC, il est difficile de fournir des données comparatives. Pour information, la meilleure approche sur cette base avec les faux expérimentés (alors que nous travaillons sur des faux aléatoires) obtient un TEE de 2,8 %. Ensuite, les taux passent à 4,4 % puis 5 % pour les deuxièmes et troisièmes meilleures approches [KHOLMATOV 05].

Dans la suite des expérimentations, nous nous sommes limités à l'utilisation du système correspondant au prototype léger, c'est-à-dire utilisant la distance spatiale seule. Les résultats seront donc à comparer avec la première ligne du tableau 1.

4.2 Résultats avec la classification basée sur la durée totale et la longueur totale

Cette première série de résultats donne les performances obtenues selon le protocole défini plus haut en utilisant les caractéristiques de durée totale et de longueur totale pour déterminer des classes de signatures avec les *K-means*. Le tableau 2 montre les résultats pour différents nombres de classes (K) et avec la distance spatiale ($distS$). Les taux donnés représentent une moyenne sur 10 exécutions successives du protocole complet puisque l'algorithme de classification ne fournit pas nécessairement les mêmes résultats d'une éxécution sur l'autre. Nous fournissons alors l'écart type correspondant (EcA et EcR).

Tableau 2. Performances (en %) du système basé sur la distance spatiale et effectuant une classification des signatures avec les K-means *sur la durée totale et la longueur totale.*

K	TFA	EcA	TFR	EcR	TEE
2	1,47	0,02	2,20	0,6	1,84
3	1,48	0,04	2,33	0,00	1,92
4	1,58	0,03	2,17	0,00	1,87
5	1,61	0,09	2,17	0,00	1,89
6	1,56	0,02	2,22	0,18	1,89
7	1,64	0,12	2,06	0,07	1,85
8	1,65	0,11	2,89	0,77	2,27
9	1,8	0,06	3,06	0,6	2,43
10	1,78	0,17	3,22	0,53	2,5

Dans ce tableau, on constate que jusqu'à 7 classes environ, les résultats sont globalement meilleurs que ceux du système initial : jusqu'à 5,2 % d'amélioration du TEE. Au-delà, les performances se dégradent (y compris au-delà de 10 classes). On notera particulièrement que le TFA est plus faible, notamment avec un faible nombre de classes, puis qu'il ré-augmente à partir de K égal à 7. On peut donc en déduire que notre spécialisation des seuils est bien efficace (jusqu'à un certain point) puisque les attaques sont mieux décelées. Cependant, la contrepartie est que le TFR tend à augmenter avec K. Une explication est que pour certains signataires, toutes ses signatures ne correspondent pas en réalité à la même classe. Ces dernières sont donc plus souvent rejetées lors de l'authentification. Ce point semble confirmé en observant la répartition par classes des signatures de chaque signataire. Cet inconvénient devrait théoriquement pouvoir être limité en essayant d'obtenir une meilleure classification, c'est-à-dire en changeant d'algorithme ou bien en changeant d'espace de représentation pour la classification, ce qui fait l'objet des expérimentations suivantes. On notera également que l'écart-type est assez faible pour 3, 4, 5 et 7 classes alors qu'il est plutôt élevé dans les autres cas, ce qui signifie pour ces derniers que la classification n'est pas réellement stable d'une fois sur l'autre.

4.4 Résultats avec la classification et différentes caractéristiques

Le tableau 3 présente les résultats obtenus avec d'autres jeux de caractéristiques. Nous avons dans un premier temps sélectionné la longueur totale et l'accélération moyenne puisque celles-ci apparaissent comme peu corrélées (*cf.* figure 4). Pour les mêmes raisons, nous avons également sélectionné la durée totale et le rapport entre les déplacements haut et bas. Enfin, nous avons

8. Normalement, le TEE correspond au seuil de décision permettant d'avoir égalité entre le TFA et le TFR. Comme nous sommes dans la phase de test, le seuil est désormais fixé et le TEE exact, *i.e.* tel que défini précédemment, est donc impossible à obtenir sur les signatures de test. La moyenne du TFA et TFR représente donc une approximation de ce TEE.

évalué les résultats en utilisant les caractéristiques synthétiques correspondant aux 2 et 3 premiers axes principaux qui apportent respectivement 70 et 80 % de l'inertie.[9] Pour toutes ces expérimentations, les résultats sont toujours la moyenne et l'écart type sur 10 exécutions et en utilisant la distance spatiale. Nous ne reportons que les résultats pour la valeur de *K* qui donne les meilleures performances.

Tableau 3. Performances (en %) du systèmes basé sur la distance spatiale et effectuant une classification des signatures avec les K-means *sur différentes caractéristiques.*

Caractéristiques	K	TFA	EcA	TFR	EcR	TEE
longTotale, accMoy	2	1,42	0,04	3,04	0,17	2,23
tpsTotal, rapDepVer	2	1,84	0,07	1,87	0,21	1,86
ACP x 2	3	1,56	0,06	2,13	0,06	1,85
ACP x 3	2	1,85	0,00	2,00	0,00	1,93

La première constatation que nous pouvons faire est la même que précédemment : les meilleures performances sont obtenues pour un faible nombre de classes, ici 2 ou 3. Au-delà, le TEE est toujours moins bon que celui du système initial. Le second point, très important, est que l'utilisation de caractéristiques peu corrélées n'apporte pas nécessairement une amélioration et qu'elle s'accompagne généralement d'un EcR assez élevé. On peut en déduire que la classification produite n'est pas stable d'un coup sur l'autre. L'utilisation des axes de l'ACP permet d'améliorer nettement cette stabilité (EcA et EcR faibles), avec un TEE équivalent au meilleur trouvé jusqu'alors si l'on utilise 2 axes (avec 3 axes les performances se dégradent).
Dans tous les cas, les résultats sont toujours en-dessous de ce à quoi on pourrait s'attendre. Cela s'explique là encore par le fait que pour un certain nombre de signataires, des signatures ne sont pas dans la même classe que les signatures de référence, voire que toutes les signatures de référence ne sont pas elles-mêmes dans la même classe.

4.5 Résultats avec les *Fuzzy C-means*

Les résultats précédents tendent à montrer que la classification est en général extrêmement sensible (écart-type élévé) et pas toujours favorable à tous les signataires. Par conséquent, nos dernières expérimentations ont porté sur l'utilisation d'un algorithme de classification plus stable que les *K-means* : les *Fuzzy C-means*. Le tableau 4 montre les performances obtenues avec cet algorithme dans les mêmes conditions que précédemment. Les résultats sont fournis avec différents types de caractéristiques et à chaque fois avec la meilleure valeur de *K*.

9. Nous avons également tenté de sélectionner automatiquement des sous-espaces de représentation adapté en utilisant un algorithme de sélection de caractéristique classique (SFFS [PUDIL 94]) avec des critères non supervisés (mesures d'entropie ou d'ambiguïté [SEMANI 04]) mais les résultats n'ont rien donné de très intéressant.

Tableau 4. Performances (en M) du système basé sur la distance spatiale et effectuant une classification des signatures avec les Fuzzy C-means *sur différentes caractéristiques.*

Caractéristiques	K	TFA	EcA	TFR	EcR	TEE
longTotale, tpsTotal	3	1,75	0,00	2,00	0,00	1,87
ACP x 2	3	1,66	0,00	1,67	0,07	1,66
ACP x 3	2	1,79	0,00	2,00	0,00	1,89

Comme attendu, nous pouvons constater que la classification est très stable en général : EcA/EcR quasi nul (y compris pour d'autres valeurs de *K* faibles). De plus, le TEE est systématiquement meilleur par rapport aux tests équivalents avec les *K-means*. On arrive même à une classification particulièrement avantageuse avec 3 classes et les 2 axes de l'ACP. Le gain obtenu est alors de 14,4 % sur le TEE. Cependant, les résultats restent très sensibles au nombre de classes et se détériorent assez facilement.

5. Conclusions et perspectives

Dans cet article, nous avons étudié l'impact de la création de classes de signatures pour un système d'authentification en ligne. Ces classes sont élaborées automatiquement par les *K-means* ou les *Fuzzy C-means* sans utilisation de connaissances *a priori*, ce qui permet d'utiliser la méthode à partir de n'importe quelle base d'expérimentation. Une amélioration des performances peut ainsi être obtenue en spécialisant le seuil de décision pour chacune des classes. Quand un nouvel utilisateur s'enrôle, le seuil utilisé est celui de la classe la plus en adéquation avec son style de signature.
Les expérimentations, conduites sur la base SVC, ont permis de mettre en évidence l'importance du choix de l'espace de représentation, du nombre de classes ainsi que de l'algorithme de classification. Ainsi, on a pu constater que le choix de caractéristiques peu corrélées n'apportait pas toujours de bonnes performances et qu'elles s'accompagnaient souvent d'un écart-type important sur le TFR. L'utilisation des axes de l'ACP permet de réduire cette instabilité mais il ne garantit pas à lui seul de bonnes performances. Il en va de même de l'utilisation des *Fuzzy C-means*, bien qu'elles améliorent globalement la qualité de la classification et des résultats. Enfin, on constate qu'un trop grand nombre de classes dégrade rapidement le TEE. Ainsi, les meilleurs résultats – TEE de 1,66 %, *i.e.* un gain de 14,4 % par rapport au système initial – sont obtenus pour une unique combinaison de ces critères : utilisation des *Fuzzy C-means* avec 3 classes et les 2 axes issus de l'ACP. Les autres combinaisons améliorent très faiblement le TEE ou plus généralement le dégradent.
La difficulté à trouver la bonne combinaison vient du fait que la classification obtenue n'est pas pertinente pour tous les signa-

taires. Notamment, certains ont des signatures qui appartiennent à plusieurs classes. Cela ne signifie pas nécessairement qu'il n'existe pas de classes de signatures/signataires mais plutôt qu'il semble difficile d'obtenir une bonne classification pour tous les signataires, même si ceux-ci signent en utilisant un seul style de signatures. En partant de ce principe, il semblerait plus avantageux : soit de considérer qu'un même signataire puisse appartenir à différentes classes, soit de faire plusieurs classifications à partir de différentes caractéristiques et de choisir pour chaque signataire la classification la plus adaptée. Ces expérimentations seront conduites au cours de nos futurs travaux. Nous essaierons également d'utiliser plusieurs bases d'expérimentation (MCYT[10] ou SUSIG[11] par exemple) afin de pouvoir apprendre les classes de signatures sur des jeux de données bien différents de ceux utilisés pour les tests. Cela permettra de s'affranchir du protocole de validation croisée, tout en permettant d'étudier le pouvoir de généralisation des classes de signature.

Références

[BENSEFIA 05] A. BENSEFIA, T. PAQUET, L HEUTTE, « Identification et vérification du scripteur dans des documents manuscrits », *Traitement du Signal*, vol. 22, #3, 2005, p. 249-259.

[BEZDEK 81] J. C. BEZDEK, « Pattern recognition with fuzzy objective function algorithms », Plenum Press, 1981.

[CRETTEZ 95] J.-P. CRETTEZ, « A set of handwriting families: style recognition », *Proceedings of ICDAR'95*, vol. 1, 1995, p. 489-494.

[JAIN 02] A. JAIN, F. GRIESS, S. CONNELL, « On-line signature Verification », *Pattern Recognition*, vol. 35, #12, 2002, p. 2963-2972.

[KHOLMATOV 05] A. KHOLMATOV, B. YANIKOGLU, « Identity authentication using improved online signature verification method », *Pattern Recognition Letters*, vol. 26, 2005, p. 2400-2408.

[LECLERC 94] F. LECLERC, R. PLAMONDON, « Automatic Signature Verification: the State of The Art 1989-1993 », *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 8, #3, 1994, p. 643-660.

[LEJTMAN 01] D. Z. LEJTMAN, S. E. GEORGE, « On-line handwritten signature verification using wavelets and back-propagation neural networks », *Actes de ICDAR'01*, 2001, p. 992-996.

[MCQUEEN 67] J. B. MCQUEEN, « Some methods for classification and analysis of multivariate observations », *Actes du 5ème Symposium on Mathematical Statistics and Probability de Berkeley*, vol. 1, 1967, p. 281-296.

[NOSARY 99] A. NOSARY, L. HEUTTE, T. PAQUET, Y. LECOURTIER, « Defining writer's invariants to adapt the recognition task », *Actes de ICDAR'99*, vol. 1, 1999, p. 765-768.

[OHISHI 00] T. OHISHI, Y. KOMIYA, T. MATSUMOTO, « On-line signature verification using pen-position, pen-pressure and pen-inclination trajectories », *Actes de ICPR'00*, vol. 4, 2000, p. 547-550.

[PLAMONDON 89] R. PLAMONDON, G. LORETTE, « Automatic signature verification and writer identification - state of the art », *Pattern Recognition*, vol. 22, #2, 1989, p. 107-131.

[PUDIL 94] P. PUDIL, J. NOVOVICOVA, J. KITTLER, « Floating search methods in feature selection », *Pattern Recognition Letters*, vol.15, 1994, p. 1119-1125.

[RAGOT 08] N. RAGOT, J. FORTUNE, N. VINCENT, H. CARDOT, « Study of Temporal Variability in On-Line Signature Verification », *Proceedings of the Eleventh International Conference on Frontiers in Handwriting Recognition (ICFHR'08)*, 2008, p.556-561.

[SEDEYN 02] M.-J. SEDEYN, « Délits d'écrits : lettres anonymes, faux témoignages, chèques falsifiés... », *Éditions Alternatives*, 2002.

[SEMANI 04] D. SEMANI, C. FRÉLICOT, P. COURTELLEMONT, « Combinaison d'étiquettes floues/possibilistes pour la sélection de variables », *14ième Congrès Francophone AFRIF-AFIA de Reconnaissance des Formes et Intelligence Artificielle, RFIA'04*, vol. 2, 2004, pages 479-488.

[SEROPIAN 02] A. SEROPIAN, N. VINCENT, « Writers authentication and fractal compression », *Actes de IWFHR'02*, 2002, p. 434-439.

[SIDDIQI 07] I. SIDDIQI, N. VINCENT, « Writer Identification in Handwritten Documents », *Actes de ICDAR'07*, vol. 1, 2007, p. 108-112.

[SVC 04] *Signature Verification Competition*: \\http://www.cs.ust.hk/svc2004/download.html, 2004.

[WIROTIUS 05a] M. WIROTIUS, J.-Y. RAMEL, N. VINCENT, « Contribution of global temporal information for authentication by on-line handwritten signatures », *Actes de IGS'05*, 2005, p. 266-270.

[WIROTIUS 05b] M. WIROTIUS, « Authentification par signature manuscrite sur support nomade », Thèse de doctorat en Informatique, Université de Tours, 2005.

[YEUNG 04] D.-Y. YEUNG, H. CHANG, Y. XIONG, S. GEORGE, R. KASHI, T. MATSUMOTO, G. RIGOLL, « SVC2004: First International Verification Competition », *Actes de ICBA'04*, 2004, p. 16-22.

10. http://atvs.ii.uam.es/databases.jsp

11. http://fens.sabanciuniv.edu/biometrics/eng/SUSig.html

Nicolas **Ragot**

Nicolas Ragot a obtenu son doctorat d'informatique en 2003 à l'IRISA (Institut de Recherche en Informatique et Systèmes Aléatoires), université de Rennes 1. Depuis septembre 2005, il est maître de conférences au sein du Département Informatique de Polytech'Tours, université François Rabelais Tours. Ses travaux de recherches s'effectuent dans l'équipe RFAI (Reconnaissance des Formes et Analyse d'Images) du Laboratoire Informatique de l'université François Rabelais Tours (EA 2101). Ses recherches portent sur la reconnaissance de formes et plus spécifiquement sur l'architecture des systèmes de reconnaissance. D'un point de vue applicatif, ses travaux concernent principalement la biométrie et le traitement automatique des documents.

Julie **Fortune**

Julie Fortune a commencé son doctorat d'informatique au sein de l'équipe RFAI (Reconnaissance des Formes et Analyse d'Images) du Laboratoire Informatique de l'université François Rabelais Tours (LI EA 2101), et en collaboration avec la société ATOS Worldline. Ses travaux de recherche portaient sur l'authentification forte par signature manuscrite.

Paul **M'Bongo**

Paul M'Bongo est un ancien élève ingénieur en informatique de Polytech'Tours, université François Rabelais Tours. Lors de son projet de fin d'étude, il a travaillé au sein l'équipe RFAI (Reconnaissance des Formes et Analyse d'Images) du Laboratoire Informatique de l'université François Rabelais Tours (LI EA 2101) sur la classification de signatures manuscrites en ligne.

Nicole **Vincent**

Nicole Vincent, normalienne agrégée de mathématiques, est professeur depuis 1996 et travaille actuellement à l'université Paris Descartes, avec l'équipe Systèmes Intelligents de Perception (SIP) du laboratoire d'Informatique université Paris Descartes (LIPADE). Elle est investie dans le domaine de l'analyse du document, imprimé et manuscrit, en particulier pour l'authentification des scripteurs. Plus généralement ses travaux concernent la reconnaissance des formes, pour l'indexation d'images ou la détection d'objets dans des séquences vidéo.

Hubert **Cardot**

Hubert Cardot est professeur en informatique à l'université François Rabelais Tours (Polytech'Tours / Laboratoire Informatique : LI EA 2101). Il est directeur-adjoint de ce laboratoire et responsable de l'équipe Reconnaissance de Formes et Analyse d'Images (RFAI) regroupant une vingtaine de chercheurs. Son doctorat (1993) a porté sur l'authentification hors-ligne de signatures manuscrites. Il effectue sa recherche sur le thème de la reconnaissance des formes en particulier pour la classification, la prévision et l'extraction de caractéristiques.