Système de classification à deux niveaux de décision combinant approche par modélisation et machines à vecteurs de support

Two-stage Classification System Combining Model-based Approach and Support Vector Machines

Jonathan Milgram, Robert Sabourin et Mohamed Cheriet

Laboratoire d'Imagerie, de Vision et d'Intelligence Artificielle, École de Techonologie Supérieure, Montréal, Canada,
milgram@livia.etsmtl.ca

Manuscrit reçu le 19 mai 2005

Résumé et mots clés

Il est possible de distinguer deux types de données pouvant causer des problèmes à un classifieur : les données ambiguës et les données aberrantes. À ces deux types d'erreurs peuvent être associés deux types de rejet : le rejet d'ambiguïté et le rejet d'ignorance. Or, si les approches de classification agissant par séparation sont mieux adaptées au premier type de rejet, elles s'avèrent peu efficaces pour traiter les données aberrantes. Par contre, les approches qui agissent par modélisation sont par nature mieux adaptées à ce second type de rejet, mais ne s'avèrent que peu discriminantes. Ainsi, nous proposons de combiner les deux types d'approche au sein d'un système de classification à deux niveaux de décision. Au premier niveau, une approche par modélisation sera utilisée pour rejeter les données aberrantes et pré-estimer les probabilités *a posteriori*. En cas de conflit entre plusieurs classes, des machines à vecteurs de support (SVM) appropriées seront utilisées pour ré-estimer plus précisément les probabilités des classes en conflit et permettre de rejeter efficacement les données ambiguës. En outre, cette combinaison présente l'avantage de réduire la complexité de calcul associée à la prise de décision des SVM. Ainsi, les résultats obtenus sur un problème classique de reconnaissance d'images de chiffres manuscrits isolés ont montré qu'il est possible de maintenir les performances associées aux SVM, tout en réduisant la complexité d'un facteur 8.7 et en permettant de filtrer efficacement les données aberrantes.

Combinaison de classifieurs, machine à vecteurs de support, approches par modélisation, estimation de probabilités *a posteriori*, rejet d'ambiguïté, rejet d'ignorance, coût de classification.

Abstract and key words

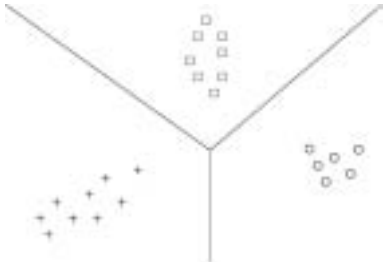
The motivation of this work is based on two key observations. First, the classification algorithms can be separated into two main categories : discriminative and model-based approaches. Second, two types of patterns can generate problems : ambiguous patterns and outliers. While, the first approach tries to minimize the first type of error, but cannot deal effectively with outliers, the model-based approaches make the outlier detection possible, but are not sufficiently discriminant. Thus, we propose to combine these two different approaches in a two-stage classification system embedded in a probabilistic framework. In the first stage we pre-estimate the posterior probabilities with a model-based approach and we re-estimate only the highest probabilities with appropriate Support Vector Machine

(SVM) in the second stage. Another advantage of this combination is to reduce the principal burden of SVM : the processing time necessary to make a decision. Finally, the first experiments on the benchmark database MNIST have shown that our dynamic classification process allows to maintain the accuracy of SVMs, while decreasing complexity by a factor 8.7 and making the outlier rejection available.

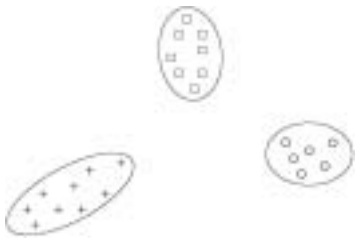
Classifier Combination, Support Vector Machine, Model-Based Approach, Posterior probability Estimation, Outlier Detection, Errror-Reject Tradeoff, Classifying Cost.

1. Introduction

Lors de la conception d'un système de reconnaissance de formes, l'objectif principal est de minimiser les erreurs de classification. Cependant, un autre critère important est la capacité à estimer une mesure de confiance dans la décision prise par le système. En effet, une telle mesure est essentielle pour permettre de ne pas prendre de décision lorsque le résultat de la classification est incertain. Ainsi, il est important de différencier deux types de rejet, correspondant à deux catégories de données délicates. Le rejet d'ambiguïté consiste, comme son nom l'indique, à filtrer les exemples ambigus. Alors que le second type de rejet concerne les données aberrantes qui ne correspondent à aucune des classes du problème. On parle alors de rejet d'ignorance, de rejet de distance ou encore de détection d'« *outliers* ». Or, parmi l'ensemble des techniques de classification, il est possible de distinguer deux catégories d'approches, celles agissant par séparation et celles agissant par modélisation. L'objectif du premier type d'approche (Figure 1-a) est d'optimiser des frontières de décision de manière à séparer au mieux les classes, alors que le second type (Figure 1-b) cherche à déterminer un modèle le plus fidèle possible de chacune des classes. La décision est alors prise dans le premier cas en se basant sur la position de l'exemple par rapport aux frontières et dans le second cas en utilisant une mesure de similarité pour comparer la donnée à classifier à chacun des modèles.



(a) par séparation



(b) par modélisation

Figure 1. Deux types d'approche de classification.

Ainsi, comme il est montré expérimentalement dans [LIU 02], de part leur nature discriminante, les approches par séparation sont plus performantes pour traiter les données ambiguës, mais peu aptes à gérer les « *outliers* ». Par contre, les approches par modélisation s'avèrent plus efficace pour détecter ces données aberrantes, mais sont généralement peu discriminantes. A partir de ces constatations, les auteurs proposent deux options : soit fusionner les deux approches de manière interne au sein d'un système hybride, soit les combiner de manière externe. Ainsi, dans un article plus récent [LIU 03], les mêmes auteurs présentent un système hybride qui utilise un apprentissage discriminant pour améliorer les performances de leur approche par modélisation. Mais, bien que très satisfaisants, les taux de reconnaissance obtenus restent inférieurs à ceux rendus possible par l'utilisation de machines à vecteurs de support (SVM).

Par conséquent, nous proposons de combiner approche par modélisation et SVM au sein d'un système de classification à deux niveaux de décision. L'idée consiste alors à utiliser dans un premier niveau de décision une approche par modélisation pour rejeter les *outliers*, classer les données ne présentant aucune ambiguïté et isoler les éventuelles classes en conflit. Alors, en cas de conflit, le second niveau de décision utilisera les SVM appropriés pour permettre une meilleure classification. De plus, cette combinaison présente l'avantage de réduire le principal fardeau des SVM : la complexité de calcul nécessaire à la prise de décision.

Ainsi, bien qu'un certain nombre d'idées similaires aient été introduites dans des articles récents [BEL 03][PRE 03] [VUU 03], notre système reste différent et original. En effet,

une première combinaison entre approche par modélisation et approche par séparation a été proposée dans [PRE 03], mais les auteurs n'utilisent alors que quelques perceptrons multi-couches (MLP) pour améliorer les performances de leur approche par modélisation et ne s'intéressent ni à la notion d'« *outliers* », ni au coût de classification. Par contre, le problème de la complexité liée aux SVM est traité dans [BEL 03], mais le système n'utilise pas d'approche par modélisation. En effet, le premier niveau de décision utilise un MLP pour sélectionner automatiquement ce qui lui semble être le « bon » SVM. Ainsi, l'utilisation de deux approches par séparation ne permet pas de rejeter efficacement les « *outliers* ». Pour résoudre ce problème, les auteurs proposent d'utiliser un autre SVM. Mais, ceci nécessite alors de disposer d'une base conséquente de données aberrantes et risque de s'avérer très coûteux en terme de complexité. Par ailleurs, si les combinaisons proposées dans [BEL 03] et [PRE 03] supposent qu'un conflit ne peut engager que deux classes, plusieurs méthodes de détection de conflits ne se limitant pas à deux classes sont proposées dans [VUU 03]. Mais les auteurs utilisent alors comme premier niveau de décision un ensemble de classifieurs qui s'avère particulièrement lourd. Il est alors possible de se demander s'il ne serait pas préférable d'utiliser directement l'ensemble des SVM.

Pour notre part, nous avons choisi, contrairement aux travaux cités précédemment, de nous placer dans un contexte probabiliste. En effet, il est généralement essentiel de disposer d'une mesure de confiance dans la décision lorque le classifieur ne contribue qu'en partie à la décision finale ou lorsqu'il est préférable ne pas prendre de décision en cas d'incertitude. Ainsi, une approche par modélisation sera utilisée au premier niveau de notre système pour pré-estimer les probabilités *a posteriori*, alors que le second niveau utilisera en cas de conflit les SVM appropriés pour ré-estimer uniquement les probabilités les plus fortes.

Enfin, de manière à comparer la qualité des probabilités estimées par les différentes approches, nous utiliserons la règle de Chow pour évaluer le compromis erreur-rejet. En effet, comme il est montré dans [FUM 00], cette règle de rejet est optimale uniquement si les lois de probabilités sont connues, ce qui n'est pas le cas pour la majorité des applications réelles. Nous pouvons ainsi considérer qu'entre deux classifieurs, celui qui permettra d'obtenir le meilleur compromis erreur-rejet sera celui qui estime le mieux les probabilités *a posteriori*.

2. Approche par modélisation

Ce type d'approche est basé sur le développement d'un modèle pour chacune des classes et l'utilisation d'une mesure de similarité entre l'exemple à classer et les différents modèles.

2.1. Caractérisation du problème de reconnaissance de formes

Bien que peu discriminante, ce type d'approche peut servir de premier niveau de décision et permettre de caractériser le problème de classification. Trois cas de figure sont alors envisageables :

- Une seule mesure de similarité est significative. La décision peut donc être prise directement.
- Plusieurs mesures de similarité sont significatives. Il s'agit alors d'une donnée ambiguë et il est donc préférable d'utiliser une approche discriminante pour prendre la décision.
- Aucune des mesures de similarité n'est significative. Il s'agit alors d'une donnée aberrante qu'il est préférable de rejeter.

Un exemple simple est présenté Figure 2. Les mesures de similarité correspondant aux modèles des deux classes sont représentées par des lignes de niveaux en (a) et (b), alors que la combinaison de ces deux mesures montre en (c) comment il est possible de détecter les « *outliers* » en utilisant le minimum des deux mesures et en (d) comment isoler les données ambiguës en utilisant le maximum des deux mesures.

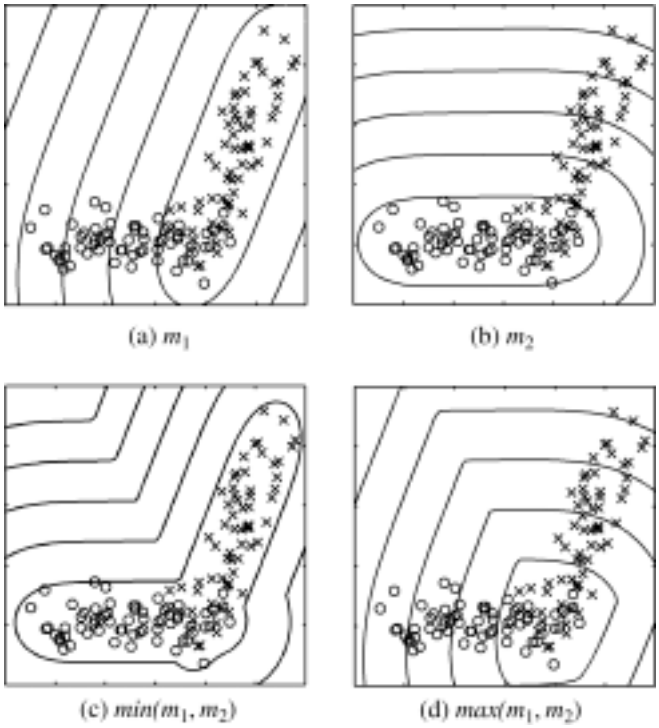


Figure 2. Utilisation d'une approche par modélisation pour caractériser le problème de classification.

D'autre part, la modularité de ce type d'approche présente l'avantage de permettre de traiter efficacement des problèmes où le nombre de classes est très grand. En effet, comme il est montré dans [OH 02], l'utilisation d'une approche globale tel qu'un réseau MLP s'avère inefficace lorsque le nombre de classes augmente comme dans le cas des 352 caractères coréens utilisés dans les adresses postales.

2.2. Modélisation à l'aide de sous-espaces vectoriels

Afin de modéliser chacune des classes, nous avons choisi une méthode simple basée sur l'hypothèse que les données des différentes classes suivent approximativement des distributions gaussiennes. Chaque classe ω_j est alors modélisée par le sous-espace vectoriel défini par la matrice Ψ_j qui contient les k premiers vecteurs propres ϕ_j^i extraits de la matrice de covariance et par la moyenne μ_j des données de la classe. Notons par ailleurs que les ϕ_j^i sont des vecteurs colonnes alors que nos exemples ainsi que les moyennes μ_j sont représentés par des vecteurs lignes.

Le principal avantage d'une telle méthode réside dans sa capacité à interpoler les données de manière à obtenir des modèles très compacts et donc extrêmement légers en termes de complexité de calcul. La mesure de similarité utilisée (ou plutôt de dissimilarité) est alors la distance de projection sur le sous-espace :

$$d_j(x) = ||x - f_j(x)|| \tag{1}$$

Ainsi, pour tout point x de l'espace de représentation, le degré d'appartenance à une classe ω_j peut être évalué en calculant la distance d_j entre le point x considéré et sa projection sur le sous-espace correspondant :

$$f_j(x) = (x - \mu_j)\Psi_j\Psi_j^T + \mu_j \tag{2}$$

Par ailleurs, afin de réduire la complexité de calcul, il est possible de calculer directement la distance de projection d_j sans avoir à calculer $f_j(x)$:

$$d_j(x) = ||x - \mu_j||^2 - \sum_{i=1}^{k} \left\{(x - \mu_j)\phi_j^i\right\}^2 \tag{3}$$

Un exemple simple en deux dimensions est présenté Figure 3. Les données de chaque classe sont alors modélisées par leur axe principal $(k = 1)$ et le point x est projeté en $f_1(x)$ et $f_2(x)$.

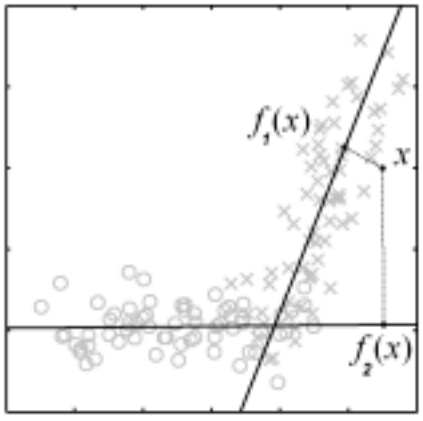


Figure 3. Exemple de modélisation d'un problème en deux dimensions. Le sous-espace associé à chacune des classes correspond alors à l'axe principal des données de la classe (k = 1).

En outre, cette approche ne nécessite que l'optimisation du paramètre k correspondant à la dimension des sous-espaces vectoriels. Toutefois, comme nous le verrons expérimentalement, il est important de ne pas négliger ce paramètre qui joue un rôle important dans la qualité de la modélisation. En effet, si k est trop petit, la perte d'information est importante et la modélisation peu précise. Si l'on considère le cas extrême où $k = 0$, une classe ω_j n'est alors modélisée que par le prototype μ_j correspondant à la moyenne des données de la classe. Par contre, si k est trop grand, les modèles engendrés ne sont plus discriminants. Si l'on considère l'autre cas extrême où $k = d$, d étant le nombre de caractéristiques, une classe ω_j n'est alors plus modélisée par un sous-espace vectoriel mais par un espace de même dimension que l'espace de représentation. Alors, quel que soit le point x, la distance de projection est nulle.

2.3. Pré-estimation des probabilités *a posteriori*

Si l'exemple traité ne semble pas aberrant, il est alors possible d'estimer à ce premier niveau du système les probabilités *a posteriori* d'appartenance aux différentes classes. Ainsi, si l'on suppose que les distributions des distances de projection correspondant aux exemples ambigus suivent un comportement exponentiel, il est alors possible d'utiliser la fonction « *softmax* » (voir la section 6.6.2 de [DUD 01]) pour estimer les probabilités *a posteriori* à partir des distances de projection :

$$\widehat{P}_f(\omega_j|x) = \frac{\exp(-\alpha d_j(x))}{\sum_{j'=1}^{c} \exp(-\alpha d_{j'}(x))} \tag{4}$$

où c représente le nombre de classes du problème.

Le paramètre α sera alors optimisé de manière à minimiser la fonction d'entropie croisée :

$$-\sum_{k=1}^{n}\sum_{j=1}^{c} t_k^j \log(\widehat{P}_f(\omega_j|x_k)) \tag{5}$$

où n représente le nombre de données d'apprentissage et t_k^j la probabilité cible.

3. Combinaison avec une approche discriminante

Par la suite, en cas d'ambiguïté, des experts discriminants appropriés seront utilisés au second niveau du système de manière à ré-estimer les probabilités *a posteriori* les plus fortes.

3.1. Détection de conflits

La première étape consiste donc à détecter les données ambiguës. Considérant qu'un conflit peut parfois impliquer plus que deux classes, il nous semble préférable de faire varier dynamiquement le nombre de classes en conflit en fonction des probabilités estimées au premier niveau du système. Pour ce faire, nous déterminons une liste de p classes $\{\omega_{\ell(1)},\ldots,\omega_{\ell(p)}\}$ pour lesquelles les probabilités *a posteriori* estimées au premier niveau sont supérieures à un seuil ε. Alors, $\ell(j)$ correspond à l'indice de la $j^{\text{ème}}$ classe vérifiant :

$$\hat{P}_f(\omega_{\ell(j)} \mid x) > \varepsilon \tag{6}$$

Ainsi, si p est supérieur à 1, il s'agit d'une situation de conflit. Alors des experts discriminants seront utilisés pour ré-estimer les probabilités *a posteriori* des p classes.
Finalement, ce paramètre contrôle la tolérance du premier niveau de classification et permet par conséquent d'effectuer un compromis entre le coût de calcul et les performances en classification. En effet, plus le seuil ε est petit, plus le nombre p aura tendance à devenir grand. Le second niveau du système risque alors d'être utilisé abusivement au détriment de la complexité. Par contre, si ε est trop grand, le second niveau sera alors trop peu utilisé pour permettre de bonnes performances en classification.

3.2. Utilisation de machines à vecteurs de support

Récemment, par le biais d'une étude comparative des principales techniques utilisées en reconnaissance de chiffres manuscrits, il a été montré dans [LIU 03] que les machines à vecteurs de support permettent une meilleure généralisation que les classifieurs neuronaux classiques tel que les MLP ou les réseaux RBF (Radial Basis Function). Par ailleurs, grâce à l'augmentation de la puissance de calcul et aux développements de nouveaux algorithmes d'apprentissage, il est aujourd'hui possible d'entraîner des SVM à résoudre des problèmes réels de grandes dimensions. Il semble donc intéressant d'utiliser des machines à vecteurs de support au second niveau de notre système de classification.
Ainsi, étant donné un exemple de test x et un ensemble de données d'apprentissage étiquetées $\{(x_k,y_k) : k = 1,\ldots,n\}$, où $x_k \in R^d$ et $y_k \in \{1,-1\}$, la sortie du SVM correspondant est

$$f(x) = \sum_{k=1}^{n} y_k \alpha_k K(x_k,x) + \beta, \tag{7}$$

où les données d'apprentissage, dont les multiplicateurs de Lagrange α_k sont différents de 0, sont nommées vecteurs de support. Malheureusement, cette sortie n'est pas nécessairement calibrée. Mais une solution est proposée dans [PLA 99] pour estimer facilement des probabilités *a posteriori* à partir des sorties d'un SVM. En effet, puisque les distributions des sorties correspondant aux exemples situés entre les marges semblent suivre un comportement exponentiel, l'auteur suggère d'utiliser une fonction sigmoïde (eq. 8) pour estimer les probabilités.

$$\widehat{P}(y = 1 \mid x) = \frac{1}{1 + \exp(af(x) + b)} \tag{8}$$

Les paramètres a et b sont alors obtenus par minimisation de la fonction d'entropie croisée :

$$-\sum_{k=1}^{n} \Big(t_k \log\left(\widehat{P}(y_k = 1 \mid x_k)\right) + (1 - t_k) \log\left(1 - \widehat{P}(y_k = 1 \mid x_k)\right)\Big), \tag{9}$$

où $t_k = \dfrac{y_k + 1}{2}$ représente la probabilité cible.
Ainsi, de manière à résoudre ce problème, l'auteur utilise un algorithme d'optimisation de type Levenberg-Marquardt. Mais il a été récemment montré dans [LIN 03] que l'utilisation de cet algorithme pose quelques problèmes de stabilité numérique. Les auteurs ont donc proposé un autre algorithme de minimisation plus fiable. C'est donc ce second algorithme qui sera utilisé pour ajuster les sigmoïdes additionnelles qui permettront d'estimer les probabilités *a posteriori* à partir des sorties des SVM.
D'autre part, un SVM est un classifieur binaire, il est donc nécessaire de combiner plusieurs SVM pour résoudre un problème multi-classe. La stratégie la plus classique est le « un contre tous » qui consiste à construire un SVM par classe. Chaque classifieur est alors entraîné à distinguer les exemples de sa classe de ceux de toutes les autres classes. Une autre stratégie classique est le « un contre un » qui consiste à construire un SVM par paire de classes. Pour un problème à c classes, cette stratégie revient donc à entraîner $c(c-1)/2$ classifieurs binaires, mais comme il est montré dans [CHA 01], étant donné que chaque sous-problème est beaucoup moins complexe à résoudre, il est plus rapide d'entraîner l'ensemble des SVM de la stratégie « un contre un » que les c SVM de l'approche « un contre tous ». Par ailleurs, dans notre cas, il semble préférable d'utiliser la seconde stratégie qui est plus modulaire et qui permettra de se focaliser uniquement sur les p classes en conflit. En effet, l'utilisation de la première stratégie conduirait à calculer des distances aux vecteurs de support de classes improbables, ce qui aurait pour effet d'augmenter inutilement le coût de classification.
Enfin, de manière à combiner les probabilités *a posteriori* de chaque SVM, il est possible d'appliquer le modèle de ressemblance proposé dans [HAM 03]. Alors, si l'on considère que toutes les probabilités *a priori* sont identiques, les probabilités *a posteriori* peuvent être estimées par :

$$\widehat{P}(\omega_j \mid x) = \frac{\prod_{j' \neq j} \widehat{P}(\omega_j \mid x \in \omega_{j,j'})}{\sum_{j''=1}^{c} \prod_{j' \neq j} \widehat{P}(\omega_{j''} \mid x \in \omega_{j'',j'})}, \quad (10)$$

où $\omega_{j,j'}$ représente l'union des classes ω_j et $\omega_{j'}$ et c le nombre de classes du problème.

3.3. Ré-estimation des probabilités *a posteriori*

Étant donné que notre système n'utilise que $p(p-1)/2$ SVM pour ré-estimer uniquement les probabilités *a posteriori* les plus significatives, les probabilités finales peuvent être estimées par différentes approches et ne sont donc pas nécessairement homogènes. Cependant, lorsque p est supérieur à un, le premier niveau n'estime que les probabilités les plus faibles qui peuvent être considérées comme étant négligeables. Les p probabilités significatives sont alors obtenues par :

$$\widehat{P}_s(\omega_{\ell(j)} \mid x) = \frac{\prod_{j''=1, j'' \neq j}^{p} \widehat{P}_s\left(\omega_{\ell(j)} \mid x \in \omega_{\ell(j),\ell(j'')}\right)}{\sum_{j'=1}^{p} \prod_{j''=1, j'' \neq j'}^{p} \widehat{P}_s\left(\omega_{\ell(j')} \mid x \in \omega_{\ell(j'),\ell(j'')}\right)} \times \left(1 - \sum_{j'=p+1}^{c} \widehat{P}_f\left(\omega_{\ell(j')} \mid x\right)\right), \quad (11)$$

où le premier terme correspond aux probabilités estimées au second niveau, alors que les probabilités du second terme proviennent de l'estimation effectuée au premier niveau. L'objectif de ce second terme est donc de maintenir la somme des probabilités égale à un.

Une vue d'ensemble du système proposé est présentée Figure 4. Notons que le rejet d'ignorance pourra être réalisé au premier niveau en exploitant les distances de projection ; alors que les SVM du second niveau seront utilisés de manière dynamique pour mieux estimer les probabilités *a posteriori* qui pourront être exploitées pour réaliser le rejet d'ambiguïté.

4. Résultats expérimentaux

De manière à tester l'approche proposée, nous avons choisi de nous intéresser à un problème de reconnaissance de formes classique : la reconnaissance d'images de chiffres manuscrits isolés. Ainsi, l'ensemble des expériences a été réalisé sur la base de données MNIST (Modified NIST)[1]. Il s'agit d'une base publique couramment utilisée et dont les résultats pour de nombreux classifieurs sont disponibles. Dans une étude comparative récente [LIU 03], l'utilisation conjointe de SVM et d'une procédure d'extraction de caractéristiques a permis d'obtenir le

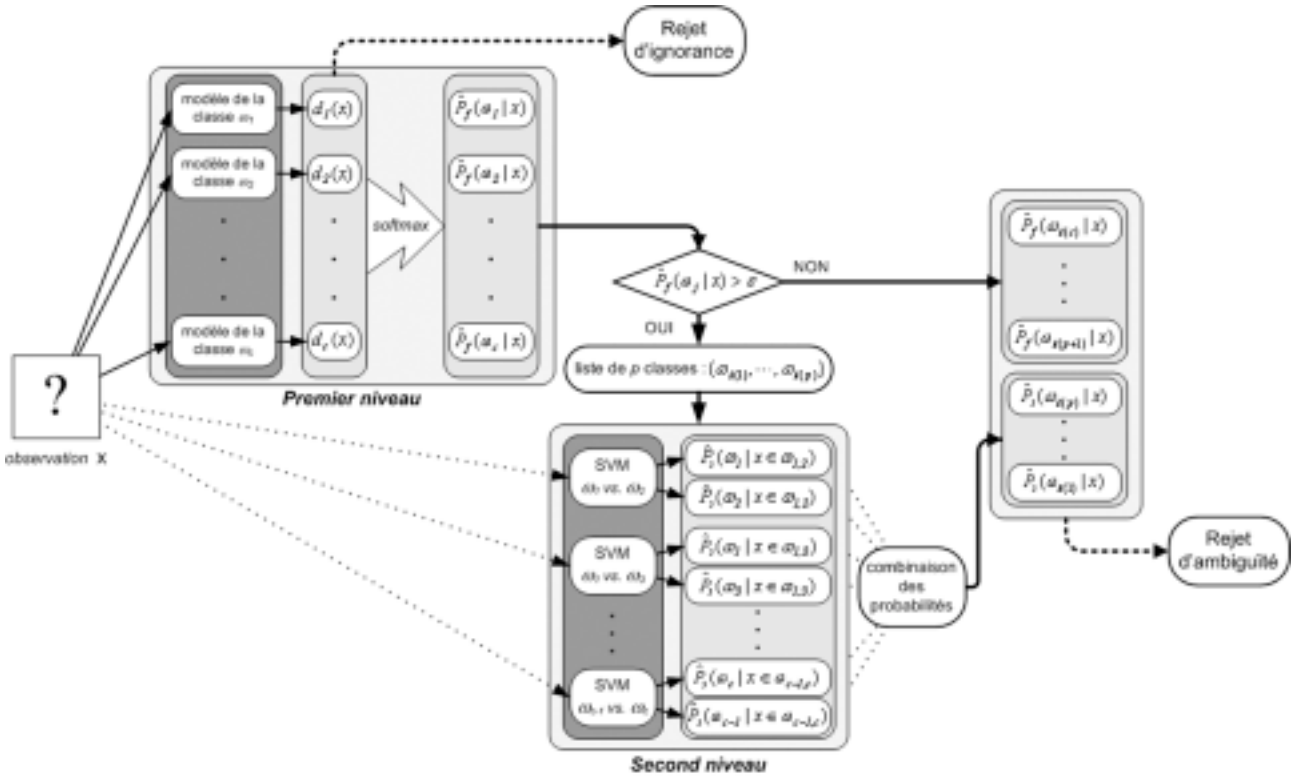


Figure 4. Vue d'ensemble de notre système de classification à deux niveaux de décision.

[1] disponible à l'adresse suivante http://yann.lecun.com/exdb/mnist/

meilleur résultat reporté à ce jour. Un bref résumé des résultats obtenus lors de cette étude est présenté ci-dessous.

Tableau 1. Taux d'erreur obtenus sur la base de test MNIST reportés dans [LIU 03].

	RBF	MLP	SVM
sans extraction de caractéristiques	2.53 %	1.91 %	1.41 %
avec extraction de caractéristiques	0.69 %	0.60 %	0.42 %

Ainsi, bien que l'extraction de caractéristiques discriminantes permette d'obtenir de meilleurs résultats, nous avons choisi d'utiliser la base originale [LEC 98] constituée d'images normalisées en dimension (20×20) puis centrées dans une rétine 28×28 en faisant coïncider le centre de gravité du caractère avec le centre géométrique de la rétine. Les 50 000 premiers exemples de la base d'apprentissage seront utilisés pour l'entraînement des classifieurs et les 10 000 suivants forment une base de validation, qui servira à l'optimisation des hyper-paramètres des différentes approches. En effet, il est préférable d'utiliser une base de données, distinctes des données utilisées lors de l'entraînement des classifieurs, de manière à éviter le phénomène de surapprentissage (voir la section 9.6 de [DUD 01]). Par ailleurs, la base de test qui est composée de 10 000 exemples sera exclusivement réservée à l'évaluation des résultats finaux de manière à comparer rigoureusement la capacité à généraliser des différentes approches.

4.1. Approche par modélisation

Dans un premier temps, il est nécessaire de fixer la dimension k des sous-espaces vectoriels. La base de validation a alors été utilisée pour estimer l'effet de k sur les performances en classification. Ainsi, comme il est possible de constater Figure 5, ce paramètre est très influent. Par conséquent, sa valeur a été fixée à $k = 25$ et le taux d'erreur sur la base de test est alors de 4.09 %. Bien que ce résultat soit peu satisfaisant, il reste comparable au 3.34 % obtenu avec la règle du plus proche voisin.

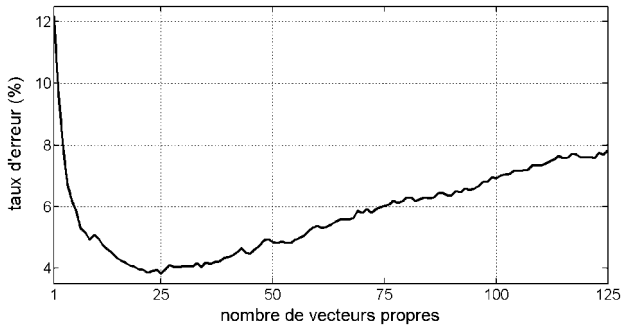


Figure 5. Effet de la dimension des sous-espaces sur les performances en classification (base de validation).

Ensuite, le paramètre α de la fonction « *softmax* » (eq. 4) a été fixé empiriquement de manière à minimiser l'entropie croisée (eq. 5). Pour ce faire, nous avons discrétisé la valeur de α entre 1 et 10 par pas de 0.1. Le meilleur résultat a alors été obtenu avec $\alpha = 5.6$. La Figure 6 illustre l'effet de la fonction « *softmax* » ainsi optimisé qui permet d'améliorer significativement le compromis erreur-rejet de l'approche par modélisation. En effet, nous pouvons par exemple constater que si le rejet est effectué en exploitant directement les distances de projection, il est nécessaire de rejeter un peu plus de 30 % des données de la base de validation pour obtenir 1 % d'erreur parmi les exemples restants ; alors qu'en exploitant les probabilités estimées à l'aide de la fonction « *softmax* », il est nécessaire de n'en rejeter qu'environ 8 % pour atteindre le même taux d'erreur.

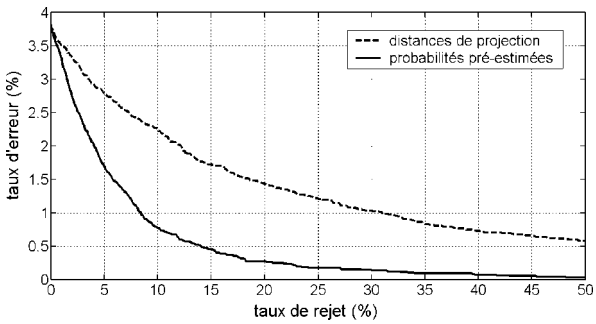


Figure 6. Capacité au rejet d'ambiguïté de l'approche par modélisation (base de validation).

Finalement, bien que peu discriminante, notre approche par modélisation permet tout de même de classer correctement la moitié des exemples avec un très haut niveau de confiance. D'autre part, cette approche permet la caractérisation du problème de classification. En effet, les trois cas de figure considérés dans la section 2.1 peuvent être observés dans une application réelle telle que la reconnaissance de chiffres manuscrits :

- Une seule distance de projection est petite. L'exemple peut alors être considéré comme **non-ambigu** et les probabilités *a posteriori* peuvent être directement estimées par l'approche par modélisation (voir Figure 7).
- Plusieurs distances de projection sont petites. L'exemple peut être considéré comme étant **ambigu** et il semble donc préférable de ré-estimer les probabilités *a posteriori* avec une approche discriminante (voir Figure 8).
- Toutes les distances de projection sont grandes. L'exemple peut être considéré comme étant **aberrant** et peut être rejeté (voir Figure 9).

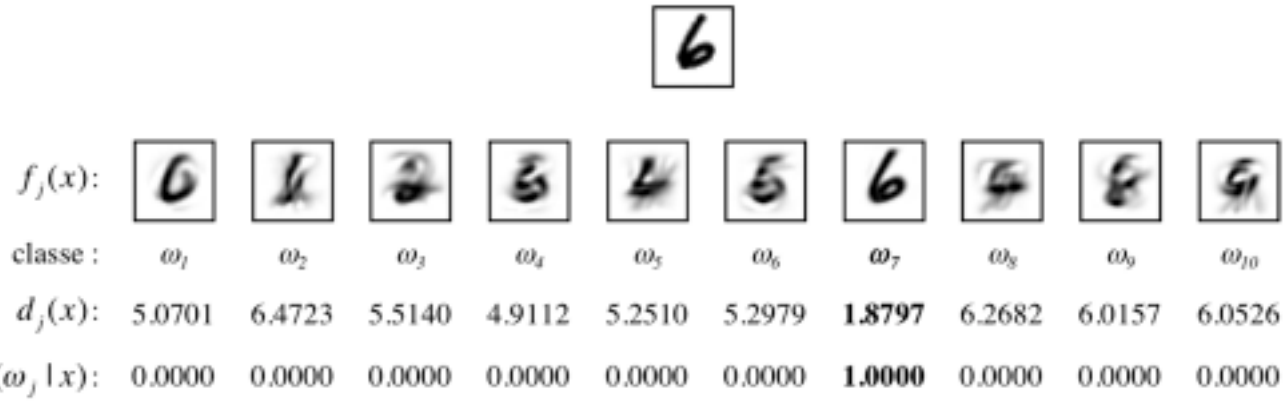


Figure 7. Exemple de donnée non-ambiguë (8400ème exemple de la base de test).

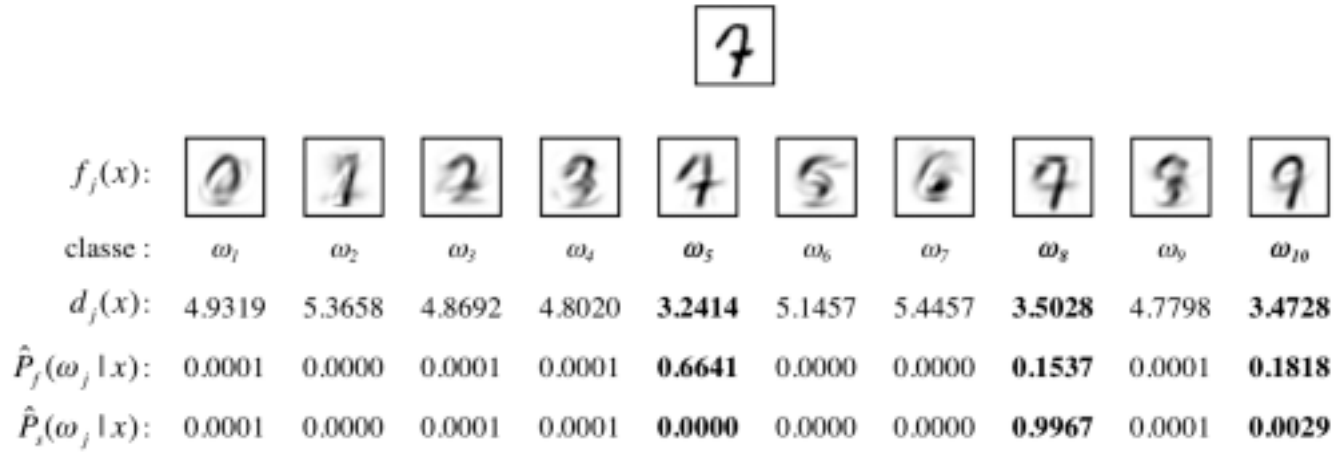


Figure 8. Exemple de donnée ambiguë (5907ème exemple de la base de test).

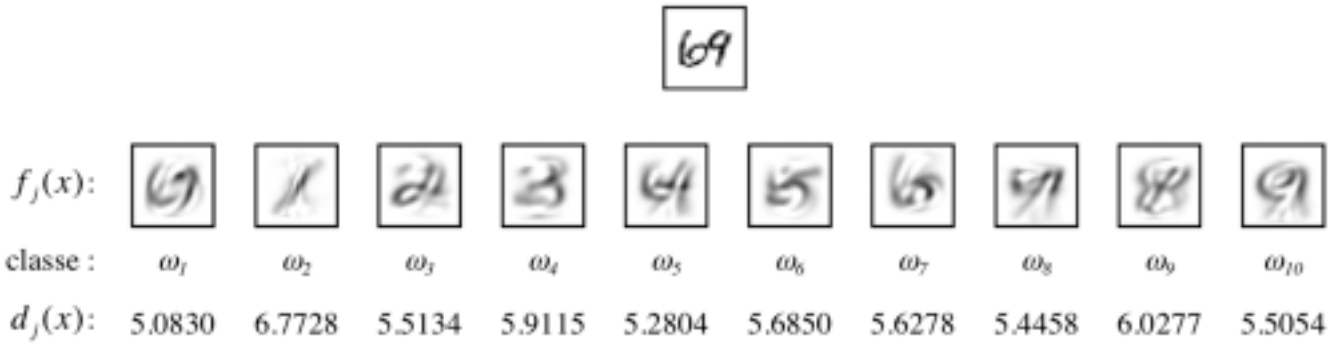


Figure 9. Exemple de donnée aberrante (généré à partir du 12ème et du 13ème exemple de la base de test).

4.2. Machine à vecteur de support

L'apprentissage et le test de tous les SVM ont été réalisés à l'aide du logiciel LIBSVM[2] dont les algorithmes sont décrits dans [CHA 01]. Nous avons choisi d'utiliser le C-SVM avec un noyau gaussien $K(x_k,x) = \exp(-\gamma||x_k - x||^2)$. Le paramètre de pénalité C et le paramètre de noyau γ ont été déterminés empiriquement en cherchant à minimiser le taux d'erreur sur la base de validation. L'ensemble de SVM construit avec les valeurs retenues ($C = 10$ et $\gamma = 0.0185$) permet finalement d'obtenir un taux d'erreur de 1.48 % sur la base de test. Notons que ce résultat est comparable à celui reporté dans [LIU 03] lorsque aucune caractéristique discriminante n'est extraite.

[2] disponible à l'adresse suivante http://www.csie.ntu.edu.tw/~cjlin/libsvm/

Finalement, les probabilités estimées par les SVM sont bien meilleures que celles estimées par l'approche par modélisation (voir Figure 10).

D'autre part, nous adopterons le nombre de vecteurs de support comme mesure du coût de classification. En effet, le calcul de la distance à l'ensemble de ces exemples d'apprentissage est nécessaire à la prise de décision et représente donc le principal effort de calcul durant la phase de test. Ainsi, notons que cet ensemble de 45 SVM utilise 11 118 vecteurs de support.

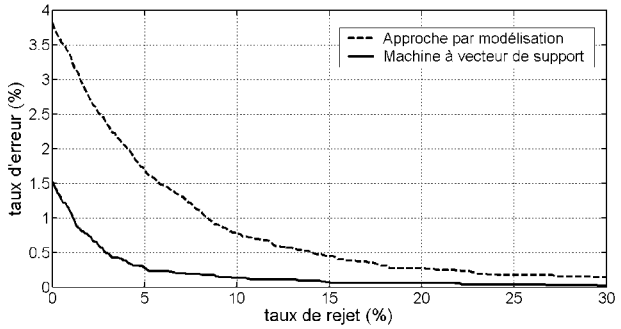


Figure 10. Capacité au rejet d'ambiguité de l'approche par séparation (base de validation).

4.3. Système de classification à deux niveaux

Notons tout d'abord que le label d'un exemple n'est pas nécessairement présent parmi les deux premières solutions proposées par l'approche par modélisation (voir Tableau 2). Ceci justifie donc le choix d'un nombre dynamique de classes en conflit.

Tableau 2. Distribution de la position du label délivré par l'approche par modélisation (base de validation).

Position du label	1	2	3	> 3
% de la base	96.18	2.50	0.76	0.56

D'autre part, en fonction des contraintes fixées par l'application visée, il est parfois nécessaire de faire un compromis entre fiabilité et complexité. Le seuil ε de l'équation (6) permet le contrôle de ce type de compromis. Ainsi, la base de validation peut une nouvelle fois être utilisée pour fixer ce paramètre (voir Figure 11).

Nous pouvons donc constater que l'utilisation d'une valeur de seuil égale à 10^{-3} permet d'obtenir exactement le même taux d'erreur (1.53%) qu'avec l'ensemble des SVM. De plus, l'utilisation d'un seuil plus petit ($\varepsilon = 10^{-4}$) ne permet d'obtenir un compromis erreur-rejet que très légèrement meilleur (voir Figure 12), au prix d'une complexité multipliée par deux.

Ainsi, le seuil de tolérance ε a été fixé à 10^{-3} ; ce qui semble être un bon compromis entre fiabilité et complexité. Par ailleurs, le nombre p de SVM utilisé étant dynamique, il est intéressant d'observer la distribution de p (voir Figure 13). Nous pouvons notamment remarquer qu'avec un seuil de 10^{-3}, pour la moitié des exemples le second niveau ne sera pas utilisé ; ce qui confirme l'observation relative à la Figure 6. De plus, il est possible de constater qu'il est parfois nécessaire d'utiliser bien plus d'un SVM pour résoudre les conflits. Ceci prouve donc que notre approche par modélisation n'est pas assez précise.

Finalement, le système à deux niveaux utilise seulement une moyenne de 1 120.1 vecteurs de support et permet d'obtenir un taux d'erreur sur la base de test de 1.50 % ; ce qui est comparable au résultat obtenu avec l'ensemble des SVM (1.48 %).

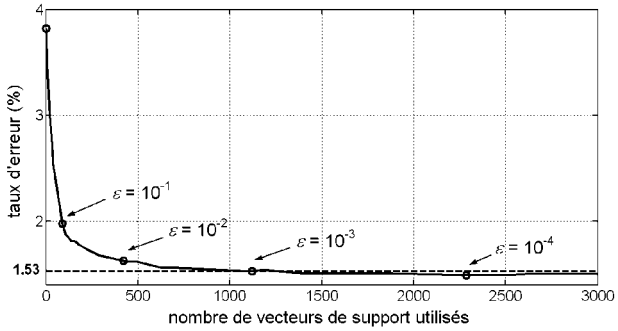


Figure 11. Compromis entre fiabilité et complexité (base de validation).

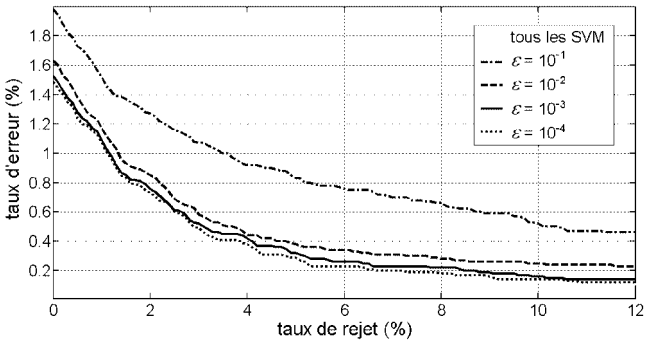


Figure 12. Capacité au rejet d'ambiguité du système à deux niveaux (base de validation).

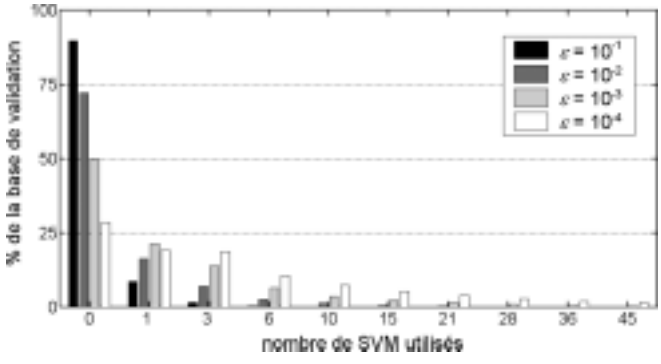


Figure 13. Distribution du nombre de SVM utilisés pour classer les données de la base de validation.

4.4. Rejet d'ignorance

Afin d'évaluer la capacité au rejet d'ignorance de l'approche par modélisation, une base d'« *outliers* » a été générée artificiellement. Les exemples de la base de test ont ainsi été assemblés deux à deux de manière consécutive pour obtenir 10 000 images de chiffres collés, dont les 100 premiers exemples sont présentés ci-dessous.

Plusieurs approches ont alors pu être comparées en observant le compromis entre le taux d'« *outliers* » parmi les exemples acceptés et le taux de « chiffres » parmi les exemples rejetés. La Figure 15 présente ainsi les résultats obtenus en utilisant les 10 000 exemples de test et les 10 000 « *outliers* » correspondants. Dans le cas de notre approche par modélisation, un exemple est rejeté si la plus petite des distances de projection aux différents sous-espaces est supérieure au seuil de rejet. Dans le cas du plus proche voisin, c'est alors la distance au plus proche des 50 000 exemples de la base d'apprentissage qui est utilisée. Concernant les SVM, deux stratégies ont été testées. La première consiste à rejeter les exemples dont la plus forte probabilité *a posteriori* est inférieure au seuil de rejet. La seconde utilise les distances aux 11 118 vecteurs de supports de la même manière que pour le plus proche voisin.

Ainsi, il est possible de constater que notre approche par modélisation s'avère légèrement plus performante que la règle du plus proche voisin utilisant l'ensemble des exemples d'apprentissage et significativement plus performante que celle n'utilisant que les vecteurs de support. Enfin, les résultats obtenus à partir de l'estimation des probabilités *a posteriori* confirment que les approches agissant par séparation sont moins adaptées au rejet d'ignorance que les approches agissant par modélisation.

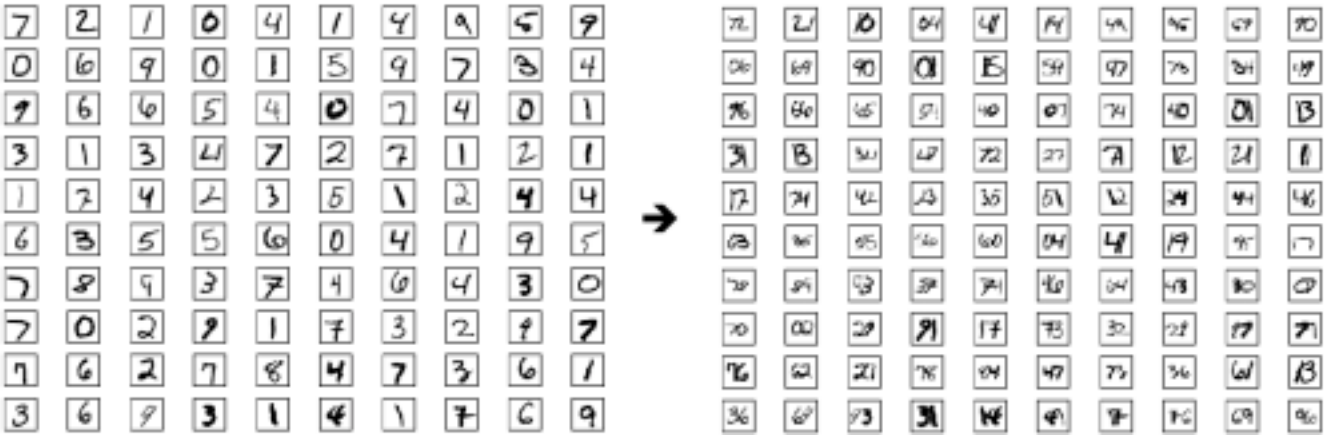


Figure 14. Création d'une base d'« outliers » à partir des données de la base de test.

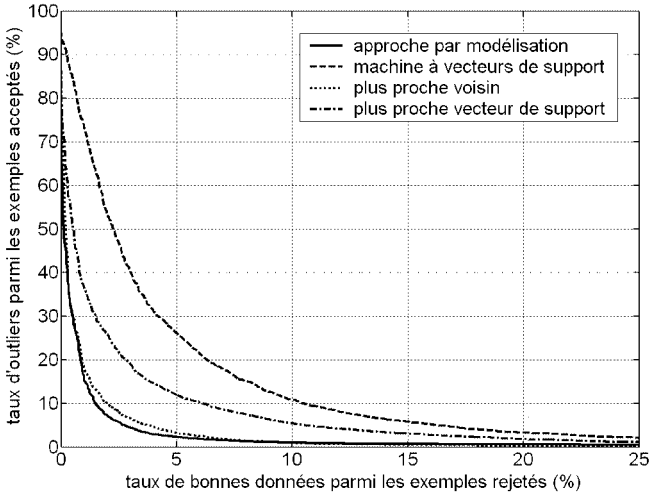


Figure 15. Évaluation de la capacité au rejet d'ignorance.

5. Conclusions et perspectives

Nous avons donc présenté dans cet article une nouvelle architecture qui présente plusieurs propriétés intéressantes pour la reconnaissance de caractères. Le système proposé combine les avantages des approches par modélisation, tels que la modularité et la possibilité de rejeter les données aberrantes, avec l'important pouvoir discriminant des SVM. Un autre avantage de cette combinaison à deux niveaux est la réduction du coût de classification lié aux SVM. En effet, le temps de traitement est un facteur très important dans la majorité des applications réelles. Or, les résultats obtenus sur la base de donnée MNIST montrent que l'utilisation d'un premier niveau permet de réduire d'un facteur 8.7 le coût de classification, tout en conservant quasiment la même fiabilité qu'avec l'ensemble des SVM (voir

Tableau 3). En effet, si nous exprimons la complexité de calcul en nombre d'opérations (FLOPs), le calcul de la distance à un vecteur de support requiert 2 355 FLOPs, alors que le calcul de la distance de projection sur un sous-espace requiert 40 755 FLOPs. Ainsi, le coût de calcul nécessaire pour classer un exemple est d'approximativement 26.2 MFLOPs avec l'ensemble des 45 SVM, seulement 0.4 MFLOPs avec l'approche par modélisation et de 3.0 MFLOPs en moyenne avec la procédure dynamique utilisant les deux niveaux de classification.

Tableau 3. Capacité au rejet d'ambiguïté des trois approches (base de test).

	taux d'erreur (%)	0.5	0.4	0.3	0.2	0.1
taux de rejet (%)	approche par modélisation	12.68	13.74	16.97	20.01	28.59
	système à deux niveaux	3.31	3.99	4.94	6.57	9.85
	machine à vecteurs de support	3.29	4.00	5.13	6.34	9.55

Bien entendu plusieurs points de notre approche pourraient être améliorés pour une meilleure intégration dans un système réel. Concernant l'approche par modélisation utilisée, une amélioration possible serait de décomposer chaque classe en sous-classes et de modéliser chaque sous-classe par un sous-espace différent. D'autre part, de manière à améliorer significativement la capacité de généralisation du système, il serait possible d'utiliser des connaissances *a priori* du problème. Deux solutions sont alors envisageables. La première consiste à extraire des caractéristiques discriminantes telles que celles utilisées dans [LIU 03] qui ont permis de réduire le taux d'erreur sans rejet à seulement 0.4%. La seconde possibilité consiste à générer des exemples artificiels comme cela est réalisé dans [PRE 03].

Références

[BEL 03] A. BELLILI, M. GILLOUX, P. GALLINARI, « An MLP-SVM combination architecture for offline handwritten digit recognition », *International Journal on Document Analysis and Recognition*, Vol. 5, #4, 2003, p. 244-252.

[CHA 01] C.-C. CHANG, C.-J. LIN., « LIBSVM : a library for support vector machines », Technical rapport, Department of Computer Science and Information Engineering, National Taiwan University, 2001. Software available at : http ://www.csie.ntu.edu.tw/~cjlin/libsvm/.

[DUD 01] R.O. DUDA, P.E. HART, D.G. STORK, « Pattern Classification », Wiley-Interscience, Second Edition, 2001.

[FUM 00] G. FUMERA, F. ROLI, G. GIACINTO, « Reject option with multiple thresholds », *Pattern Recognition*, Vol. 33, #12, 2000, p. 2099-2101.

[HAM 03] T. HAMAMURA, H. MIZUTANI, B. IRIE, « A multiclass classification method based on multiple pairwise classifiers », *International Conference on Document Analysis and Recognition*, 2003, p. 809-813.

[LEC 98] Y. LECUN, L. BOTTOU, Y. BENGIO, P. HAFFNER, « Gradient-based learning applied to document recognition », *Proceedings of IEEE*, Vol. 86, #11, 1998, p. 2278-2324.

[LIN 03] H.-T. LIN, C.-J. LIN, R.C. WENG, « A note on Platt's probabilistic outputs for support vector machines », Technical report, Department of Computer Science and Information Engineering, National Taiwan University, 2003.

[LIU 02] C.-L. LIU, H. SAKO, H. FUJISAWA, « Performance evaluation of pattern classifiers for handwritten character recognition », *International Journal on Document Analysis and Recognition*, Vol. 4, #3, 2002, p. 191-204.

[LIU 03] C.-L. LIU, K. NAKASHIMA, H. SAKO, H. FUJISAWA, « Handwritten digit recognition : benchmarking of state-of-the-art techniques », *Pattern Recognition*, Vol. 36, #10, 2003, p. 2271-2285.

[OH 02] I.-S. OH, C.-Y. SUEN, « A class-modular feedforward neural network for handwriting recognition », *Pattern Recognition*, Vol. 35, #1, 2002, p. 229-244.

[PLA 99] J.C. PLATT, « Probabilities for SV Machines », *Advances in Large Margin Classifiers*, MIT Press, 1999, p. 61-74.

[PRE 03] L. PREVOST, C. MICHEL-SENDIS, A. MOISES, L. OUDOT, M. MILGRAM, « Combining model-based and discriminative classifiers : application to handwritten character recognition », *International Conference on Document Analysis and Recognition*, 2003, p. 31-35.

[VUU 03] L. VUURPIJL, L. SCHOMAKER, M. VAN ERP, « Architectures for detecting and solving conflicts : two-stage classification and support vector classifiers », *International Journal on Document Analysis and Recognition*, Vol. 5, #4, 2003, p. 213-223.

Jonathan **Milgram**

Jonathan Milgram est né à Paris en 1975. Il est titulaire d'une maîtrise EEA (Electronique, Electrotechnique et Automatique) et d'un DEA de robotique de l'Université Pierre et Marie Curie (Paris 6). Il est actuellement étudiant en doctorat à l'École de Technologie Supérieure (ETS) de Montréal (Canada). Ses travaux de recherche au sein du LIVIA (Laboratoire d'Imagerie, de Vision et d'Intelligence Artificielle) portent sur l'utilisation de machines à vecteurs de support pour la reconnaissance de formes. Par ailleurs, M. Milgram a obtenu durant trois années consécutives une bourse d'excellence de l'ETS et s'est vu décerner le prix du meilleur papier « jeune chercheur » lors de CIFED'04 (Conférence Internationale Francophone sur l'Ecrit et le Document).

Robert **Sabourin**

En 1977, le professeur R. Sabourin rejoint le département de physique de l'université de Montréal, où il est responsable de la conception, l'expérimentation et le développement d'instrumentation scientifique destiné à l'observatoire astronomique du mont Mégantic. En 1983, il rejoint l'école de technologie supérieure (Université du Québec, Montréal) et participe à la création du département de génie de la production automatisée, où il est encore aujourd'hui professeur titulaire. Il y enseigne la reconnaissance de formes, les algorithmes évolutionnistes, les réseaux de neurones et les systèmes flous. En 1992, il rejoint également le département d'informatique de la Pontifícia Universidade Católica do Paraná (Curitiba, Brésil), où il a participé à la création d'un programme de Master en 1995 et de Doctorat en 1998. Depuis 1996, il est membre sénior du Centre for Pattern Recognition and Machine Intelligence (CENPARMI, Université Concordia).

Le professeur Sabourin est auteur (ou co-auteur) de plus de 150 publications scientifiques. Il a été co-président de CIFED'98 (Conférence Internationale Francophone sur l'Écrit et le Document, Québec, Canada) et co-président du comité de programme du IWFHR'04 (International Workshop on Frontiers in Handwriting recognition, Tokyo, Japan). Il a été nommé comme co-président du prochain ICDAR'07 (International conference on Document Analysis and Recognition) qui aura lieu à Curitiba, Brésil.

Ses intérêts de recherche sont la reconnaissance de l'écriture et la vérification de signature pour les applications bancaires et postales.

Mohamed **Cheriet**

Mohamed Cheriet a reçu son degré d'ingénieur d'état en informatique de l'Université USTHB d'Alger en 1984, et a reçu les degrés de DEA et de Doctorat également en informatique de l'Université de Paris 6 (Pierre et marie Curie) en 1985 et 1988 respectivement. De 1988 à 1990, il a travaillé en tant qu'associé de recherche au laboratoire de LAFORIA/CNRS à l'École des Ponts et Chaussées de Paris. Il a ensuite joint CENPARMI (Centre pour la reconnaissance des formes et de machines intelligentes) à l'Université de Concordia à Montréal, où il a travaillé en tant que stagiaire post-doctoral pendant deux années. Il a été nommé professeur adjoint en 1992, professeur agrégé en 1996 et professeur titulaire en 1998, au département de génie de la productique automatisée de l'École de Technologie Supérieure (ETS) à Montréal. Il est aussi le directeur du laboratoire LIVIA (laboratoire d'imagerie, de vision, et d'intelligence artificielle) à l'ETS depuis 2000. La recherche de Dr. Cheriet se concentre sur les modèles mathématiques de traitement d'image (EDPs, théorie variationelle, filtres multi-echelles à support compact), la reconnaissance de caractères, l'analyse et la reconnaissance de documents manuscrits, et la perception. Dr. Cheriet est un membre sénior de IEEE et un membre actif de CENPARMI. Il a publié plus de 100 articles dans des actes de conférences internationales et dans des journaux de renom. Il a servi comme président et coprésident de plusieurs conférences internationales : Vision Interface'98&2000, IWFHR'2002&2008, et comme coprésident des arrangements locaux d'ICDAR'95, IJCNN'2005. Dr. Cheriet est Editeur Associé du journal IJPRAI «The International Journal of Pattern Recognition and Artificial Intelligence».