

Identification/Réidentification de personnes sur des vidéos grâce à des algorithmes de classification incrémentaux.

Yanyun LU, Anthony FLEURY, Jacques BOONAERT, Stéphane LECOEUICHE

Unité de Recherche en Informatique et Automatique, École des Mines de Douai
941 Rue Charles Bourseul, CS 10838, 59508 France

yanyun.lu@mines-douai.fr, anthony.fleury@mines-douai.fr
jacques.boonaert@mines-douai.fr, stephane.lecoeuiche@mines-douai.fr

Résumé – La reconnaissance de personnes dans le cadre de la vidéo-surveillance est un thème très important de nos jours. Cette communication s'intéresse à la reconnaissance automatique, avec des données qui ne sont pas complètes lors de la phase initiale d'apprentissage, et donc qui nécessite des algorithmes de classification en ligne. Sur une base de 20 personnes présentées avec des orientations différentes et en marchant, nous donnons les résultats de la classification de ces personnes avec tout d'abord un algorithme de SVM incrémental, qui a une connaissance de toutes les classes mais avec un très faible nombre d'éléments pour chaque classe à l'apprentissage puis s'adapte, et dans un second temps avec un algorithme dénommé SAKM, qui va permettre l'apprentissage de nouveautés et donc la découverte de nouvelles personnes arrivant. Nous obtenons des résultats à plus de 95% de bonne classification dans les deux cas, le premier la classification des 20 personnes avec une très faible base d'apprentissage (une première partie de séquence) et dans le second cas avec l'apprentissage uniquement sur trois classes et la découverte de sept classes au total, dont 4 nouvelles.

Abstract – People identification in video-surveillance systems is a problem of interest. This paper deals with automatic recognition of person with incomplete data during training phase requiring on-line training algorithms. In a database including 20 persons walking with different orientation comparing to the camera, we obtained results, first, on classification using Incremental SVM, using at the begining small amount of data belonging to the whole set of classes but adapting the models as the data comes (training set contains a very few number of instances). In a second experiment, we used SAKM algorithm, which can deal with novelty detection. Both experiments gave very good results (over 95% of correct classification) in the first case with the 20 persons and in the second case by training on few images of 3 persons and testing on 7 persons.

1 Introduction

La reconnaissance de personnes est une thématique primordiale dans les systèmes de vidéo-surveillance, de plus en plus utilisés dans les transports et les lieux publics, pour laquelle de nombreuses techniques ont été développées. Ces techniques permettent de détecter une personne et de l'isoler dans l'image, mais aussi de l'identifier [1]. De nombreuses recherches ont émergé ces dernières années au sein de cette thématique. Ces recherches sont, pour un bon nombre, basées sur des caractéristiques biométriques (marche, visage, iris, etc.) [2], mais certaines se basent maintenant sur des caractéristiques d'apparence [3], spécifiquement pour des applications liées à la détection des piétons et plus généralement tous les systèmes d'analyse multi-caméra [4]. L'apparence se base sur les vêtements visibles de la personne qui sont facilement détectables et obtenus par élimination du fond. Pendant une période de temps relativement courte, nous pouvons espérer que l'apparence de la personne soit constante (pas de changement de vêtements). Cependant, les conditions d'illumination et l'orientation de la personne par rapport à la caméra peuvent, elles, changer.

L'apprentissage de toutes les orientations avec toutes les illuminations ne pouvant être réalisé, nous avons précédemment

utilisé des techniques d'apprentissage incrémentales basé sur les SVM pour réaliser cette identification de personne [5]. Cela permet d'apprendre sur une courte séquence et de reconnaître la personne avec des orientations ou des illuminations différentes dans d'autres séquences. Cependant, un autre problème émerge alors, il s'agit de pouvoir, dans des endroits par exemple publics, identifier une personne qui est encore inconnue du système (l'apprentissage de nouveautés). Avec les techniques basées sur les SVM incrémentaux, cela est rendu très compliqué car il faudrait trouver une condition sur les données qui soit assez spécifique pour permettre de détecter la différence entre une nouvelle classe et une évolution d'une classe connue et ainsi adapter l'ensemble du SVM avec une nouvelle classe à partir de données inconnues. Pour surmonter cet obstacle, nous avons choisi d'appliquer l'algorithme SAKM (Self-Adaptive Kernel Machine) [9] sur les données images et de comparer avec les résultats des exécutions précédentes utilisant les SVM incrémentaux.

La section 2 s'attardera sur la sélection des attributs que nous présenterons au classifieur. La section 3 présentera les deux algorithmes de classification qui sont comparés, avant que la section 4 ne présente les résultats de l'exécution de ces différents algorithmes. Ces résultats seront discutés dans la section 5 qui

introduira également les limitations et futurs travaux.

2 Extraction et sélection d'attributs

La base de données CASIA est une base de données vidéo contenant 20 personnes différentes, dans laquelle les personnes sont filmées en marchant selon différentes orientations. Chaque personne marche avec 6 orientations différentes par rapport à la caméra vidéo. Ces personnes n'ont pas changé de vêtements entre les différents essais. Des exemples d'images obtenues sont montrés sur la Fig. 1. Il est à noter que bien que cette base de données soit à l'origine destinée à l'analyse de la marche, elle peut très bien être utilisée dans ce cadre d'identification de personnes. Le fait qu'elle offre différentes illuminations et orientations de la personne sans changement de vêtement nous permet d'évaluer la pertinence et l'efficacité de nos choix algorithmiques.

La base est fournie avec un fond déjà soustrait. En plus des images initiales sont aussi fournies les images représentant les silhouettes des personnes (sous forme de masques). À partir de ces silhouettes, nous avons segmenté le corps en trois différentes parties : la tête, le buste puis les jambes. Afin d'être insensible aux conditions d'illumination, une normalisation a été effectuée sur les couleurs.

Pour la couleur, nous calculons 54 attributs à partir des valeurs moyennes, des écarts types, des histogrammes, etc. sur les trois couleurs de base et les différentes parties du corps prises indépendamment. À ceci s'ajoutent 13 attributs évaluant la texture présente dans les parties de l'image, évaluation qui se fait en niveau de gris et suivant les attributs définis par Haralick [6, 5]. Avec tout ceci, un total de 93 attributs (auxquels nous ajoutons la classe) est alors calculé pour caractériser une personne dans chacune des images.

Afin de sélectionner les informations discriminantes (en enlevant des attributs trop corrélés, trop peu corrélés à la classe ou des attributs ne variant pas assez) dans l'ensemble d'apprentissage, nous avons utilisé trois méthodes de réduction de la dimension, nous permettant d'obtenir trois ensembles d'attributs de taille différente. La première est l'analyse en composantes principales (PCA). Dans cette analyse, nous avons retenu un ensemble de 26 attributs (avec comme critère de réglage la conservation d'au moins 95% de la variance totale). La seconde est une méthode de sélection d'attributs basée sur les corrélations (CFS). Cette méthode se base sur le calcul d'une heuristique [10] :

$$M_s = \frac{k \cdot \bar{r}_{cf}}{\sqrt{k + k \cdot (k - 1) \cdot \bar{r}_{ff}}}$$

considérant la corrélation classe-attribut (r_{cf}) ainsi que les corrélations attribut-attribut (r_{ff}) et le nombre d'attributs (k). La finalité était de maximiser cette expression. Afin de rechercher le sous-ensemble optimal (sans parcourir l'ensemble des possibilités qui est bien trop important) d'attributs dans les 93 initiaux, une recherche basée sur du Greedy Hill Climbing bidirectionnel [7] est appliquée avec un critère d'arrêt définissant le

nombre de pas sans amélioration possible. L'exécution de cet algorithme nous donne un ensemble de 40 attributs au total.

Enfin, la dernière méthode (Wrapper) se base sur le même algorithme de recherche mais, en lieu et place du calcul d'une heuristique, les performances d'un classifieur donné (ici SVM classique) sont évaluées pour chaque sous-ensemble testé. Le critère est donc de trouver le sous-ensemble pour lequel la classification avec l'algorithme considéré donne les résultats optimaux en termes de bonne classification. Pour cette dernière méthode, 16 attributs sont retenus.

La PCA effectue une projection de l'espace d'attributs de départ dans un nouvel espace dont la base est une combinaison linéaire des précédents vecteurs. L'interprétabilité des résultats n'est donc pas vraiment possible, surtout en dimension 26. Pour ce qui est des deux autres méthodes, les attributs sélectionnés sont des attributs de l'ensemble d'origine. Dans ces deux ensembles, la plupart des attributs sont bien sûr basés sur la couleur. Il est à noter également qu'une grande partie des attributs donnés par la troisième méthode se retrouvent également dans l'ensemble de la seconde méthode (11 attributs sur les 16)

3 Apprentissage par SVM Incrémentaux et SAKM

Le cadre applicatif de ce travail est la reconnaissance de personnes par méthodes de classification. Notre but est d'effectuer l'apprentissage du classifieur sur un faible ensemble de données et d'adapter ce classifieur par la suite pour qu'il puisse reconnaître tout de même la personne même orienté de manière inconnue ou apprendre une nouvelle classe d'individu. Deux algorithmes sont implémentés à cet effet. Le premier, basé sur les SVM incrémentaux, connaît l'ensemble des classes au début et apprend au fur et à mesure alors que l'algorithme SAKM adapte l'espace de décision et le nombre de clusters au fil de l'eau. Ces deux algorithmes sont présentés par la suite.

3.1 Apprentissage par SVM Incrémental

La base de cet algorithme est d'entraîner un SVM avec une portion réduite de l'ensemble de données, de conserver uniquement les vecteurs supports et les vecteurs erreurs à chaque étapes du processus d'apprentissage, qui créera un ensemble d'apprentissage pour l'étape suivante.

La classification par SVM est définie par :

$$x_i \in C_k; k = \arg \max_{j=1, \dots, K} f_j(x_i)$$

où les fonctions de décision sont $f_i(x) = w_i^T \Phi(x) + b_i$ (avec w_i et b_i définissant les coefficients de la séparatrice et Φ est la fonction noyau). Afin d'obtenir la meilleure marge entre les classes, la méthode des multiplicateurs de Lagrange est appliquée et les fonctions de décisions sont réécrites avec les coefficients de Lagrange α_i . Ensuite, en se basant sur les conditions dites



FIGURE 1 – Une personne effectuant les 6 actions de marche différentes dans la base CASIA.

KKT en un point x_m , l'ensemble des données T est divisé en trois catégories en fonction des valeurs de g_m :

$$g_m = \begin{cases} > 0; & \text{if } \alpha_m = 0; & \text{Ensemble des vecteurs de données } DV \\ = 0; & \text{if } 0 < \alpha_m < C; & \text{Ensemble des vecteurs supports } SV \\ < 0; & \text{if } \alpha_m = C; & \text{Ensemble des vecteurs erreurs } EV \end{cases}$$

où $C \geq 0$ permet de paramétrer le nombre d'erreurs admissibles.

L'apprentissage incrémental par SVM repose sur le fait de préserver les conditions KKT sur tous les vecteurs de la base d'entraînement lorsqu'un nouveau vecteur est ajouté de manière adiabatique. La procédure incrémentale est succinctement donnée par l'algorithme 1.

Algorithm 1: Classification en ligne par SVM incrémental [8]

```

1 Nouvelle donnée  $x_c$ ; Initialise  $\alpha_c = 0$ ;
2 if  $g_c > 0$  then termine;
3 else if  $g_c \leq 0$  then
4   Applique le plus grand incrément  $\alpha_c$  afin que l'une des
   conditions suivantes se produise;
5   (1) $g_c = 0$  : Ajoute  $x_c$  aux vecteurs supports  $SV$ , met à
   jour la fonction de décision et termine;
6   (2) $\alpha_c = C$  : Ajoute  $x_c$  aux vecteurs erreurs  $EV$  et
   termine;
7   (3) $\alpha_c$  proche de 0, les vecteurs supports correspondant
   bougent dans la marge et la fonction de décision est
   mise à jour;
8   Puis répéter tant que nécessaire.
9 end

```

Une description plus précise et complète de l'ensemble de cet algorithme est donnée par [8].

3.2 Self-Adaptive Kernel Machine (SAKM)

SAKM [9] est un algorithme de clustering en ligne permettant d'agir sur des données non-stationnaires dans un contexte multiclasse. L'idée principale est de calculer le degré de proximité d'un nouvel échantillon avec les plus proches vecteurs supports de chacun des clusters dans l'espace de données, ceci avec une métrique induite par le noyau. Une nouvelle fonction de similarité est introduite en considérant $\mu\phi_{t,m}$. Dans l'espace de Hilbert défini par un noyau Gaussien, la distance d'un nouvel échantillon X_t par rapport à chaque cluster C_m^t au temps t s'exprime par :

$$\begin{aligned} \mu\phi_{t,m} &= \frac{\delta}{\sqrt{2}} \|\phi(X_t) - \phi(SV_{win,m})\|_{\Gamma} \\ &= \delta \times \sqrt{1 - \kappa(X_t, SV_{win,m})} \\ &= \delta \times \sqrt{1 - \exp(-\lambda \|X_t - SV_{win,m}\|)} \end{aligned}$$

où $SV_{win,m}$ est le vecteur support gagnant du cluster C_m^t et δ est une fonction signe égal à 1 si l'évaluation de la fonction définissant la marge en X_t est négative.

Lorsqu'une nouvelle donnée X_t est présentée, la procédure à suivre (création, adaptation, fusion) est sélectionnée, comme montré dans l'algorithme 2, suivant le critère :

$$\Omega^{win} = \{C_m^t \in \Omega^t / \mu\phi(X_t, C_m^t) \leq \varepsilon_{th}\}$$

où ε_{th} est un paramètre d'acceptabilité d'un cluster.

Algorithm 2: Classification en ligne avec SAKM [9]

```

1 Récupération des données en ligne  $X$ ; Configuration des
  paramètres et seuils;
2 Initialisation des fonctions de décision et des paramètres
  des noyaux avec les données d'apprentissage;
3 while Nouvelles données disponibles do
4   Évaluation de la fonction de similarité basée sur les
   noyaux :  $\mu\phi_{t,m}$ ; Critère de similarité des clusters :
    $\Omega^{win}$ ;
5   if  $\text{card}(\Omega^{win} = 0)$  then Procédure de création;
6   else if  $\text{card}(\Omega^{win} = 1)$  then Procédure d'adaptation;
7   else  $\text{card}(\Omega^{win} \geq 2)$  Procédure de fusion;
8   if Nombre de données dans les clusters inférieure au
   seuil then Procédure d'élimination
9 end

```

4 Résultats

Pour tester la méthode basée sur les SVM incrémentaux, 10% des images de la base de données sont réservés à l'apprentissage, et le reste est utilisé pour tester en mettant à jour la base de connaissance. En effet, à chaque test d'un élément, ce même élément est ensuite inséré comme élément d'apprentissage avec comme classe celle qui a été décidée. Les résultats sont montrés dans le tableau 1. Les résultats sont très bons et très proches des 99.5% donnés par les SVM avec un apprentissage plus classique (60% de la base de départ) et sans incrémentation. Cependant, cette méthode, aussi bons soient ses résultats, ne permet pas de découvrir de nouvelles classes.

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20	Global
Wholeset	92.16	99.44	100	95.77	99.10	99.37	100	100	100	100	99.78	99.31	98.98	97.34	100	96.87	98.23	91.89	97.42	99.22	98.21
PCA	91.07	99.72	98.74	95.37	99	92.15	93.65	99.76	100	100	99.23	96.11	94	94.42	91.69	97.74	98.11	85.70	85.77	100	95.6
CFS	92.59	99.44	100	97.99	99.10	99.79	100	100	100	100	99.78	98.28	99.39	97.72	100	98.62	98.23	92	96.56	100	98.46
Wrapper	91.83	99.44	99.58	91.15	99.10	92.25	99.08	99.88	100	100	96.92	95.65	86.98	93.03	94.16	95.11	97.99	83.88	93.37	99.67	95.39

TABLE 1 – Résultats de classification avec les SVM incrémentaux

Le second tableau présente les résultats avec l’algorithme SAKM. Pour ces résultats, trois classes sont au départ considérées comme connues. Pour ces trois classes, 10% des données sont prises (en début de la première séquence) pour former une base d’apprentissage et apprendre trois clusters. Une fois cet apprentissage réalisé, le reste des éléments est utilisé pour l’apprentissage en ligne. Afin de tester la possibilité de découverte de nouveautés, nous avons ainsi ajouté à cet ensemble de test quatre autres classes qui n’ont jamais été apprises. Le but étant que l’algorithme puisse détecter ces nouveautés afin de créer automatiquement une nouvelle classe. La dernière colonne du tableau nous indique le nombre total de cluster trouvé dans les données. Une étape de fusion est ensuite nécessaire pour rassembler les clusters appartenant à une même personne. À noter aussi que pour cette expérimentation nous avons choisi aléatoirement 7 individus dans le set de données. Ainsi, P1 ne correspond pas forcément à C1 du premier tableau. Les hyperparamètres des deux classifieurs ont été optimisés.

	P1	P2	P3	P4	P5	P6	P7	Nb Cluster
Wholeset	100	100	97.48	97.53	95.34	90	100	23
PCA	100	100	95.27	98.02	88.34	84.33	100	16
CFS	100	100	99.26	100	96.64	90.83	100	16
Wrapper	100	100	97.69	100	95.96	89.17	100	9

TABLE 2 – Résultats de classification avec SAKM et la découverte de nouveautés (3 classes apprises et 7 testées).

Les deux résultats sont très bons et se rapprochent de ceux de la classification non incrémentale bien qu’un nombre plus important de données d’apprentissage ait été utilisé. Nous avons ensuite conduit une seconde expérimentation avec l’algorithme non-incrémental dans laquelle les données présentées pour l’apprentissage sont limitées à celles présentées initialement dans l’algorithme incrémental. Nous observons une diminution significative du pourcentage de bonne classification sous les 85%.

5 Discussion et Conclusion

Les deux techniques d’apprentissage en ligne présentées nous montrent qu’en identification de personnes, nous arrivons à atteindre des résultats proches des algorithmes de classification habituels avec un apprentissage initial plus léger. La seconde méthode nous permet même de ne pas avoir la connaissance originelle du nombre de personnes et des personnes à classer.

Ces résultats sont cependant à approfondir pour plusieurs raisons. La première est que, pour les algorithmes en ligne, le résultat final dépend logiquement de l’ordre de présentation des images. Il faut ainsi déterminer l’impact sur les résultats d’un

ordre d’arrivée des images en adéquation avec un système de vidéo-surveillance multi-caméra dans un endroit public donné. De plus, le réglage des algorithmes est relativement sensible et la généralisation à un système de vidéo-surveillance en conditions réelles n’est pas triviale, notamment il sera nécessaire de faire un meilleur usage des procédures d’oubli.

Références

- [1] D. Truong Cong, L. Khoudour, C. Achard, C. Meurie, O. Lezoray, People re-identification by spectral classification of silhouettes, *Signal Processing* 90 (8) (2010) 2362–2374.
- [2] X. Zhou, B. Bhanu, Feature fusion of side face and gait for video-based human identification, *Pattern Recognition* 41 (3) (2008) 778 – 795.
- [3] D. Makrisa, N. Doulamisc, S. Middleton, Vision-Based Production of Personalized Video, *Signal Processing : Image Computation* 24 (5) (2009) 158–176.
- [4] D. Truong Cong, L. Khoudour, C. Achard, L. Douadi, People Detection and Re-Identification in Complex Environments, *IEICE Transactions on Information and Systems* 93 (7) (2010) 1761–1772.
- [5] Y. Lu, K. Boukharouba, J. Boonaert, A. Fleury, S. Lecœuche, Application of an Incremental SVM algorithm for On-line human recognition from video surveillance using texture and color features, *Neurocomputing* (2013) In Press. DOI :10.1016/j.neucom.2012.08.071.
- [6] R. Haralick, K. Shanmugam, I. Dinstein, Textural features for image classification, *Systems, Man and Cybernetics, IEEE Transactions on* 3 (6) (1973) 610–621.
- [7] J. Pearl, *Heuristics : Intelligent Search Strategies for Computer Problem Solving*, Addison-Wesley, 1984.
- [8] K. Boukharouba, L. Bako, S. Lecœuche, Incremental and Decremental Multi-category Classification by Support Vector Machines, in : 2009 International Conference on Machine Learning and Applications, IEEE, 2009, pp. 294–300.
- [9] H.A. Boubacar, S. Lecœuche, S. Maouche, SAKM : Self-adaptive kernel machine A kernel-based algorithm for online clustering, *Neural Networks* 21 (9) (2008) 1287–1301.
- [10] M. Hall, Correlation-based feature selection for machine learning, Ph.D. thesis, University of Waikato, New-Zealand (1999).