Détection de piétons par stéréovision et noyaux de graphes

Frédéric SUARD, Alain RAKOTOMAMONJY, Abdelaziz BENSRHAIR

Laboratoire PSI (Perception, Systèmes d'Information), CNRS FRE 2645
INSA de Rouen, FRANCE
Tél : 02 32 95 98 84 Fax : 02 32 95 97 08
frederic.suard@insa-rouen.fr

Résumé – Cet article présente une méthode concernant la reconnaissance de piétons à l'aide de graphes et de méthodes à noyaux. La détection du piéton est limitée à cause de la grande variabilité de la forme du piéton : taille, posture. Pour surmonter ce problème, nous avons choisi de le représenter à l'aide d'un graphe. Le but de la méthode est d'extraire le graphe de chaque objet (piétons ou non-piétons) présent dans une base d'images et de calculer un noyau à partir de ces graphes afin d'effectuer un apprentissage supervisé basé sur les SVMs (Séparateurs à Vaste Marge). L'application sur une base d'images réelles nous permet de démontrer l'efficacité de cette méthode, au niveau des invariances en échelle, avec un bon taux de reconnaissance.

Abstract – This article presents a novel method concerning pedestrian detection, thanks to graph kernels. Nowadays, the pedestrian detection is a hard task, due to the variability of its shape : size and posture. To address this problem, we choose to transform a pedestrian into a graph. The aim of this method consists of extracting a graph from each object (pedestrian or non-pedestrian), contained in a database. We compute the kernel with the inner product between graphs in order to apply a supervised classifier, here the SVMs (Support Vector Machine). We applied this method on a real images database in order to test its efficiency, particularly for scale invariance, and we obtained a good classification rate.

1 INTRODUCTION

Depuis quelques années le problème de la détection de piétons a inspiré de nombreux travaux dans le domaine du traitement d'images et de la reconnaissance de formes. Malgré diverses approches pour résoudre ce problème, aucune n'a réellement apporté de solutions définitives, car, d'après Shashua [9], la détection est confrontée à la très grande variabilité dans la forme du piéton.

Aujourd'hui, la tendance serait d'associer des méthodes de vision artificielle et des méthodes d'apprentissage telles que les réseaux de neurones [12], l'Adaboost [11], les moindres carrés régularisés [9] et les Séparateurs à Vaste Marge(SVM) [6].

La méthode présentée suit cette association : stéréovision et méthode à noyaux ([3]). Dans un premier temps, nous utilisons un système de stéréovision afin d'extraire les images des objets appartenant à la scène filmée. Il est ainsi possible de traiter le problème de l'occlusion à l'aide de la 3D. Nous représentons ensuite chaque objet à l'aide d'un graphe, apportant ainsi une réponse au problème de variabilité et d'invariance en échelle. La classification est basée sur les SVMs, dont le noyau est un noyau de graphes, basé sur le produit scalaire de graphes, obtenu en parcourant aléatoirement chaque graphe.

Dans cet article, nous reviendrons sur la construction des graphes à partir d'une paire d'images stéréo, puis nous décrirons rapidement le classifieur utilisé. Enfin, nous présenterons quelques résultats de cette méthode appliquée à des images réelles.

2 DESCRIPTION DE LA METHODE

Nous allons maintenant décrire rapidement la méthodologie que nous avons suivie. Dans un premier temps, il faut extraire les objets de la scène à partir des images, puis appliquer le processus de reconnaissance de formes.

2.1 Extraction des objets

L'idée originale de la méthode est de décrire un piéton à l'aide d'un graphe. En effet, un piéton est composé de plusieurs parties : une tête, un ou deux bras, un buste, une ou deux jambes. Le graphe correspond donc grossièrement au squelette, avec les arcs désignant les membres et les noeuds les articulations entre eux. La transformation d'un piéton en graphe rend ainsi la détection moins sensible aux différentes postures du piéton. Dans notre cas, un piéton de profil et un piéton de face auront quasiment le même squelette, ce qui nécessite par la suite un apprentissage moins volumineux mais produisant des résultats similaires par rapport à d'autres méthodes qui sont plus sensibles à la forme du piéton observé. Les graphes possèdent quelques propriétés que nous nous proposons d'exploiter. Un graphe permet en effet de décrire la forme générale de l'objet en conservant ses caractéristiques géométriques. Un graphe est plus facilement manipulable qu'une image et prend moins de place en mémoire, ce qui est intéressant dans le cas de grandes bases d'apprentissage. Il est également possible, au travers des étiquettes du graphe, de choisir les meilleures informations à retenir pour caractériser l'objet. Enfin, la comparaison de deux graphes est assez rapide pour un faible nombre de nœuds.

Avant de transformer le piéton en graphe, il convient donc d'extraire l'image du piéton de l'image globale. Pour cela, une segmentation est effectuée selon deux critères : une segmentation région et une segmentation 3D. Cette double segmentation est effectuée dans le but d'améliorer la qualité des images des objets extraits. La figure 1 montre un exemple de piéton po-

sant des problèmes pour la reconnaissance. Si le piéton est mal segmenté, le graphe devient alors de mauvaise qualité et la classification devient difficile.

(a) (b) (c)

FIG. 1: *Exemple de piéton mal segmenté.*

Une des deux images stéréo est tout d'abord segmentée par régions selon une méthode simple : nous regroupons les pixels voisins dans une même région dont le niveau de gris est proche. Cependant, cette segmentation ne permet pas de récupérer l'intégralité des objets. Dans le but d'étudier également le problème d'occlusion, nous utilisons la deuxième image stéréo afin de calculer les disparités de chaque région obtenues dans la première image. Ce calcul permet ainsi de regrouper les régions voisines dont la disparité est proche, c'est à dire qui appartiennent au même plan de l'image, et de différencier les régions voisines n'appartenant pas au même plan. La disparité de chaque région est calculée selon un algorithme de mise en correspondance classique : on recherche dans l'autre image la région correspondant le mieux à la région considérée [7]. Cette mise en correspondance est efficace car elle compare des régions et non des points, mais elle se révèle au final beaucoup plus coûteuse en temps de calcul. Lorsque la disparité de toutes les régions a été calculée, nous procédons à une fusion de régions. Les régions voisines dont la disparité est proche sont ainsi regroupées afin de reconstituer l'intégralité des objets. La figure 2 montre un exemple de traitement à partir d'un paire d'images stéréoscopiques (a) et (b), le résultat de la segmentation (c) région sur l'image droite, et l'image obtenue après le calcul de disparité (d). Les régions les plus claires sont les plus proches de la caméra.

Chaque région dont la disparité est non nulle est ensuite extraite et transformée en graphe.

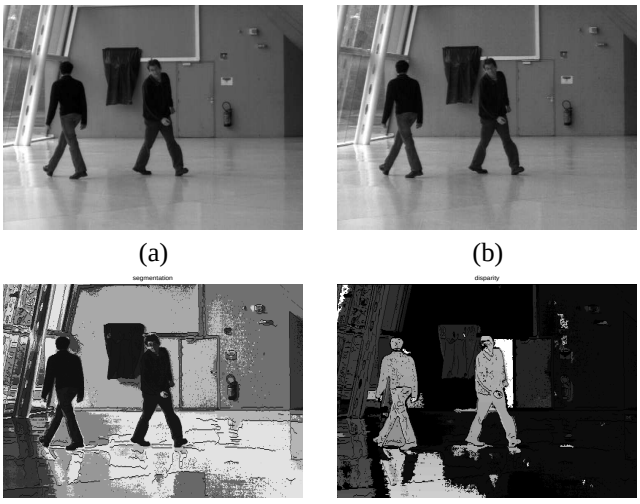
Il faut tout d'abord binariser l'image du piéton avant d'extraire le squelette de cette forme. Le squelette possède les propriétés suivantes

- son épaisseur est de 1 pixel sur tout le squelette, sauf aux intersections,
- la géométrie et la topologie de l'objet sont conservées,
- le squelette est situé au milieu de l'objet.

Il est obtenu de la façon suivante ([8]): Le squelette d'un ensemble X selon une famille d'éléments structurant δ_n est le lieu géométrique des centre de tous les éléments maximaux.

Le squelette contient donc l'information de la forme générale de l'objet, exploitable pour la reconnaissance. Il faut ensuite transformer le squelette en graphe. Un graphe $G = (N, A)$ est constitué d'un ensemble de nœuds N, reliés entre eux par un ensemble d'arcs A. Un nœud est défini de la façon suivante : c'est un pixel du squelette placé à l'extrémité d'un arc. C'est donc soit l'extrémité d'une branche du squelette, soit un pixel placé à une intersection de plusieurs branches.

Lorsque tous les nœuds ont été trouvés, il faut ensuite les relier entre eux afin de créer les arcs. Un arc est présent entre 2 nœuds si il existe un chemin de pixels appartenant au squelettre entre les 2 nœuds, sans que ce chemin ne contienne d'autres nœuds. La figure 4 illustre la transformation d'une image en graphe.



(a) (b) (c) (d)

FIG. 2: *Image originale gauche (a) et droite (b). Segmentation région de l'image droite (c). Disparité (d) obtenue à partir des images (a) et (b).*

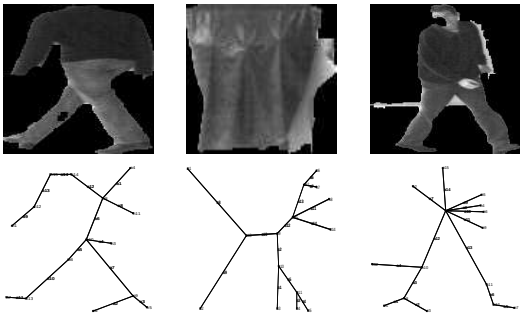
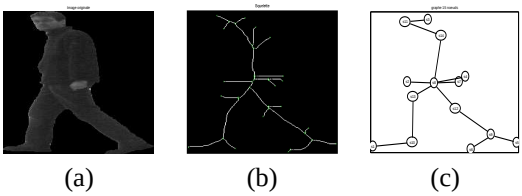


FIG. 3: *Exemples d'images d'objets extraits de la scène de la figure 2 et leur graphe correspondant.*



(a) (b) (c)

FIG. 4: *Image originale (a), son squelette (b) et le graphe résultat (c)*

L'étape suivante consiste à étiqueter le graphe, c'est à dire à ajouter de l'information relative à l'objet au niveau des arcs et des nœuds. Pour les arcs, nous avons choisi les caractéristiques suivantes :

- la longueur de l'arc,
- l'orientation de l'arc par rapport à l'orientation de l'axe principal de l'ellipse contenant l'intégralité de l'objet,
- l'aire du voisinage de l'arc, c'est à dire l'ensemble des pixels décrits par la branche du squelette représentée par

l'arc.

Pour les nœuds, nous avons retenu quelques caractéristiques :

- les coordonnées du nœud dans l'image,
- la taille du plus grand élément structurant contenu dans l'image binaire à l'emplacement du nœud.

2.2 Reconnaissance de formes

La reconnaissance de formes est basée sur une méthode d'apprentissage supervisée. Dans notre cas, les SVMs basés sur un noyau de graphes. Les SVMs étant des classifieurs binaires [2, 1, 10], ils correspondent parfaitement au problème posé en indiquant l'appartenance à la classe 'piéton' ou non. La fonction de décision est calculée selon un ensemble d'apprentissage contenant des graphes de piétons et des graphes de non-piétons.

Le noyau de graphe est défini comme étant le produit scalaire entre graphes : $K(G_i, G_j) = \langle G_i, G_j \rangle$. Le produit scalaire de deux graphes est obtenu en comparant les chemins présents dans chaque graphe en tenant compte des étiquettes d'arcs et de nœuds présents sur le chemin. Pour calculer des produits scalaires entre ces représentations graphiques, nous nous sommes basés sur l'article de H. Kashima [4].

Les graphes sont représentés par plusieurs matrices, chacune définissant les étiquettes des nœuds et des arcs. Un graphe comporte autant de matrices que de catégorie d'étiquettes. Le calcul du noyau aboutit à la résolution d'un système linéaire de taille $(|G^1||G^2|)^2$, avec $|G^i|$ le nombre de nœuds du graphe i. Cette complexité étant assez important, elle nous a forcé à réduire au maximum le nombre de nœuds dans chacun des graphes. Etant donné qu'un graphe est censé représenter les articulations d'un piéton, d'après David Moore[5], 15 nœuds sont suffisants pour représenter un piéton. Nous avons donc développé une fonction permettant de réduire le nombre de nœuds d'un graphe en conservant les informations contenues dans les nœuds et les arcs afin de travailler avec des graphes comportant au maximum 15 nœuds.

3 RESULTATS

Nous allons maintenant détailler les résultats obtenus. Nous avons réalisé plusieurs acquisitions vidéos stéréoscopiques, en intérieur, avec un éclairage constant, et une dizaine de piétons différents. Chaque image acquise comporte au maximum 4 piétons.

Nous avons ainsi obtenu environ 300 images de piétons et 500 de non-piétons (mur, poteau, escalier, etc.)

Chaque image a été transformée en graphe, et nous avons sélectionné plusieurs type d'étiquettes différentes :

- les coordonnés du nœud,
- la longueur des arcs,
- les paramètres des droites support des arcs.

Voici le mode opératoire pour tester la méthode. Nous prenons successivement un certain nombre de données d'apprentissage : 1, 5, 10, 25, 50 et 100, piétons et non-piétons tirés aléatoirement dans la base d'exemple. Pour chaque apprentissage, nous testons plusieurs paramètres pour le noyau et le classifieur SVM. Chaque procédé est itéré 10 fois afin d'obtenir des résultats robustes vis-à-vis du tirage aléatoire.

Lors du calcul du noyau, nous avons le produit $K_n(N_1, N_2)$, résultat de la comparaison des nœuds selon un noyau gaussien de largeur de bande σ_i :

$K_n(N_1, N_2) = e^{-\Sigma_{i=1}^{m} \frac{N_1^i - N_2^i}{2\sigma_i^2}}$, avec N_j^i l'étiquette j du nœud i, m le nombre de caractéristiques de chaque nœud. Nous testons donc plusieurs valeurs pour σ_i afin de retenir le meilleur paramètre.

Au niveau du classifieur, un paramètre C est utilisé pour pondérer l'importance des graphes mal classés. Nous testons également différentes valeurs pour ce paramètre de 1 à 1000.

Le résultat est affiché sur la figure 5. Comme nous pouvons le voir, lorsque C est trop faible, les graphes mal classés n'ont pas beaucoup de poids, et la frontière de décision est mal choisie d'où un taux de classification peu élevé. A partir d'une certaine valeur de C, le taux de classification se stabilise, l'augmentation de C ne modifie plus le taux de classification.

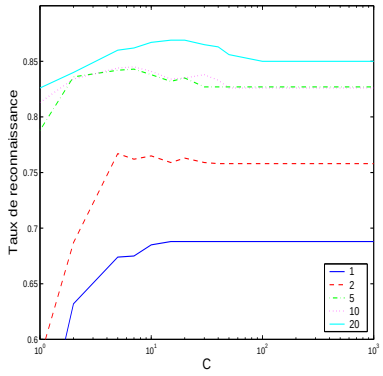
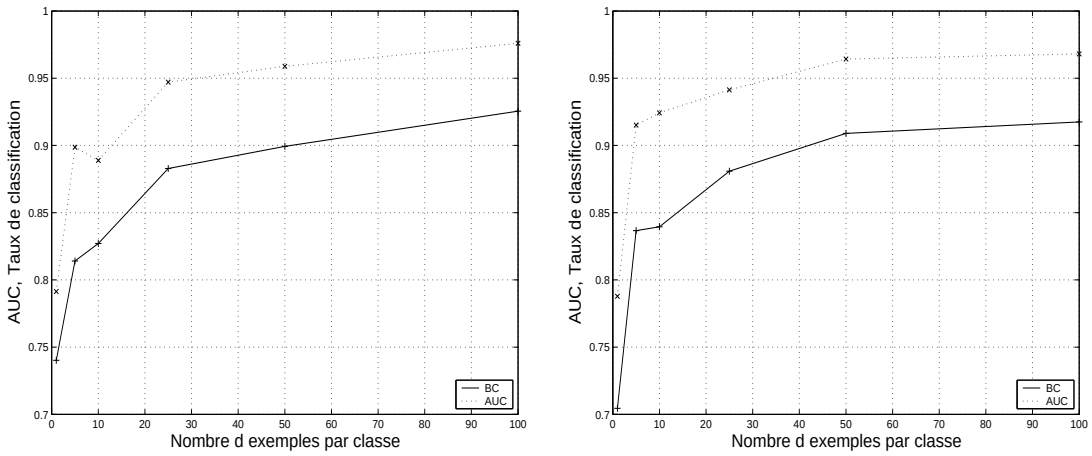


FIG. 5: *Taux de classification en fonction du paramètre C obtenus avec 3 étiquettes (a) : coordonnées(x,y) du nœud, longueur de l'arc. Chaque courbe est le résultat obtenu pour un nombre de graphes en apprentissage différent : 1,2,5,10,20.*

Nous avons également souhaité connaître l'influence des étiquettes sur les résultats. Dans un premier temps, le test est effectué sans prendre en compte les coordonnées de la droite pour les étiquettes d'arcs. Puis le test est refait en tenant compte de toutes les étiquettes citées ci-dessus. La figure 6 montre les résultats obtenus pour 3 étiquettes et 5 étiquettes, en fonction du nombre de graphes par classe utilisées lors de l'apprentissage. Nous affichons également la courbe AUC : aire sous la courbe ROC. La courbe ROC permet de connaître le lien entre le taux de vrais positifs et le taux de faux positifs. Plus l'aire est importante, plus le rapport est correct, c'est à dire que nous obtenons très peu de faux positifs pour un certain nombre de vrais positifs. Les résultats sont légèrement meilleurs en rajoutant des étiquettes, mais le gain n'est pas significatif. Les étiquettes ajoutées sont donc redondantes et ne permettent pas de renforcer les étiquettes initiales, nous allons donc continuer à tester d'autres étiquettes. Sur la figure 6, nous constatons ainsi que la courbe (a) dont les graphes possèdent 3 types d'étiquettes donne des résultats légèrement inférieurs à ceux de la courbe (b), dont les graphes possèdent 5 types d'étiquettes. Nous constatons également qu'avec peu de graphes utilisés lors de l'apprentissage, le taux de classification est satisfaisant. Nous en déduisons donc que quelques graphes de piétons permettent de caractériser correctement un ensemble important de piétons.



(a) (b)

FIG. 6: *Taux de classification (BC) et Aire Sous la Courbe ROC (AUC) en fonction du nombre de graphes par classe pour l'apprentissage obtenus avec 3 étiquettes (a) : coordonnées(x,y) du nœud, longueur de l'arc. Taux et AUC pour 5 étiquettes (b) : coordonnées(x,y) du nœud, longueur de l'arc et paramètres de la droite support de l'arc.*

4 CONCLUSION

Nous avons présenté une nouvelle méthode de détection de piétons basée sur la stéréovision et les noyaux de graphes (Random Walk Kernel). A partir d'une base d'images stéréoscopiques, nous avons extrait plusieurs images de piétons et non-piétons afin de constituer un ensemble d'apprentissage et de test. Le noyau utilisé pour le classifieur SVM est un noyau de graphes, calculé à partir du produit scalaire entre les graphes obtenus à partir des formes des piétons/non-piétons présents dans la base. Les résultats préliminaires sont très encourageant vis-à-vis de la viabilité de la méthode, notamment pour le taux de reconnaissance, élevé pour un faible ensemble d'apprentissage. La méthode apporte également une réponse au problème de variabilité du piéton. Nous prévoyons d'étendre l'expérience à d'autres séquences vidéos présentant des situations et des environnements variés. Si la méthode donne toujours de bons résultats, nous envisagerons alors une application mutli-classes et pourrons ainsi envisager la détection d'autres obstacles : vélos, poussettes, animaux, etc. Nous prévoyons également d'utiliser les graphes dans le cadre du suivi temporel de piétons. Les graphes étant a priori pratiques à utiliser pour cette application.

Remerciements Ce travail est financé en partie par le Programme IST de la Communauté Européenne, avec le réseau d'excellence PASCAL, IST-2002-506778. Cette publication reflète uniquement le point de vue des auteurs.

Références

[1] O. Chapelle, P. Haffner, and V. Vapnik. SVMs for histogram-based image classification. *IEEE Transaction on Neural Networks*, 10(5):1055–1064, 1999.

[2] N. Cristianini and J. Shawe-Taylor. *Introduction to Support Vector Machines*. Cambridge Univeristy Press, 2000.

[3] T. Gartner. Survey of kernels for structured data. In *SIGKDD Explorations*, 2003.

[4] H. Kashima, K. Tsuda, and A. Inokuchi. Marginalized kernels between labeled graphs. In *Proceedings of the Twentieh International Conference on Machine Learning*, 2003.

[5] David Moore. *A real-world system for human motion detection and tracking*. PhD thesis, California Institute of Technology, 2003.

[6] C. Papageorgiou and T. Poggio. Trainable pedestrian detection. In *Proceedings of the 1999 International Conference on Image Processing*, pages 35–39, 1999.

[7] C. Tomasi S. Birchfield. Dethp discontinuities by pixel-to-pixel stereo. In *IEEE Int. Conf on Computer Vision*, 1998.

[8] J. Serra. Morphologie mathématique. *Traité d'Informatique Géologique*, 6(6):194–238, 1972.

[9] A. Shashua, Y. Gdalyahu, and G. Hayon. Pedestrian detection for driving assistance systems: Single-frame classification and system level performance. In *Proceedings of IEEE Intelligent Vehicles Symposium*, 2004.

[10] V. Vapnik. *Statistical Learning Theory*. Wiley, 1998.

[11] P. Viola, M. Jones, and D. Snow. Pedetrian using patterns of motions and appearance. In *IEEE Int. Conf on Computer Vision*, pages 734–741, 2003.

[12] L. Zhao and C. Thorpe. Stereo and neural network based pedestrian detection. *IEEE Trans. on Intelligent Vehicles Transportation systems*, 1(3):148–154, 2000.