

Système de reconnaissance multi-classes du type de véhicules

Multiclass Vehicle Type Recognition System

P.A. Negri¹, X. Clady¹, M. Milgram¹, R. Poulencard²

¹ Institut Systèmes Intelligents et Robotique
Université Pierre et Marie Curie-Paris 6, CNRS FRE 2507

² LPREditor, Montpellier, France

ISIR, BC252, 3 rue Galilée, 94200 Ivry-sur-Seine
pablo.negri@lisif.jussieu.fr

Résumé

Cette communication présente un système de reconnaissance du type (constructeur, modèle) de véhicules par vision. A partir d'une vue de face avant d'un véhicule, nous construisons une représentation de celle-ci basée sur les points de contour orientés. La classification est réalisée essentiellement en se fondant sur des algorithmes de votes. Une construction en cascade permet de réduire la confusion entre les types de véhicules proches. Plusieurs résultats effectués sur des bases d'images prises en conditions réelles sont présentés et analysés.

Mots Clef

Reconnaissance des formes, classification multi-classes, méthode de votes, type de véhicules, vidéo-surveillance.

Abstract

This article deals with a system for multiclass vehicle type identification (make and model). From a vehicle greyscale frontal image, an oriented points set is computed in order to match it with vehicle type models. The decision process combines four classifiers : three voting algorithms and a distance error. A dynamic cascade reduces confusion between classes. The method is robust to partial occlusions. This method has been tested on a realistic data set.

Keywords

Pattern Recognition, multiclass classification, voting algorithm, vehicles type, video remote sensing.

1 Introduction

Les systèmes de vision pour la reconnaissance de véhicules s'insèrent dans diverses applications dites Systèmes de Transports Intelligents. Trois principales doivent être distinguées. En premier lieu, des caméras embarquées sont dédiées à la détection des véhicules dits obstacles [11] se présentant au devant du véhicule équipé. En second lieu,

la télésurveillance des autoroutes et autres voies de circulation permet la mesure des flots de véhicules [3]. Enfin il existe des systèmes ayant pour rôle d'identifier les véhicules se présentant aux entrées ou sorties de zones télésurveillées.

Ce sont à ces derniers que s'intéresse l'étude décrite dans cette communication. Elle est vouée à la reconnaissance du type (constructeur et modèle) des véhicules à partir d'une image fournie par une caméra. Cette dernière utilise généralement la lecture automatique de la plaque d'immatriculation. Cet identifiant est en théorie suffisant pour identifier le véhicule. Cependant, en pratique, les systèmes de vision s'y conformant uniquement, peuvent renvoyer des informations erronées, soit du fait d'une mauvaise qualité d'image soit d'une plaque illisible ou fausse. Nous pensons qu'adjoindre à ce type de système, un système de reconnaissance du type de véhicule ne peut qu'accroître l'efficacité et les capacités d'un tel système. En effet, l'information du type de véhicule peut confirmer ou infirmer celle fournie par la reconnaissance de la plaque (cf. figure 1). Les cas suspects ou indécis seraient alors signalés à une personne compétente (gardien de parking, police,...).

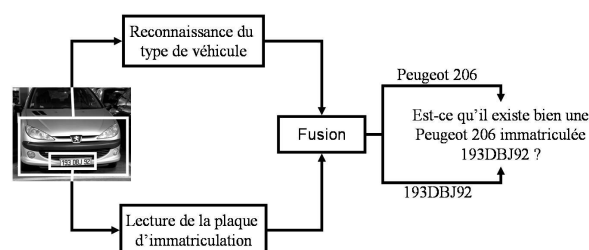


FIG. 1 – Système d'identification de véhicule fusionnant la reconnaissance de son type et la lecture de sa plaque d'immatriculation.

Dans une situation réelle d'application, les images aux en-

trées de parking sont essentiellement cadrées de manière à rendre bien visibles les plaques d'immatriculation (cf. fig. 2). Aussi pour reconnaître le type des véhicules, nous allons utiliser leur image des calandres avant (que nous appellerons prototype). Les calandres de différentes voitures ont des caractéristiques très variées d'un modèle à l'autre. Cependant, chaque modèle de véhicule a un gabarit constant, défini par le constructeur. Ceci rend les contours des calandres relativement constants d'une image d'un véhicule à celle d'un autre du même type. Afin de mieux discriminer ceux-ci, nous ajoutons à l'information du point de contour l'orientation du gradient en ce point.

A notre connaissance, seuls Petrovic et Cootes [10] ont abordé le problème de la reconnaissance automatique du type de véhicules par vision dans une étude comparative sur la représentation d'objets à structure rigide. Ils ont appliqués différentes méthodes d'extraction de *features* sur une base d'images de véhicules en stationnement. Une région d'intérêt dans l'image du véhicule, cadrée sur la calandre avant de celui-ci, est obtenue en référence aux coins de la plaque minéralogique. Ils utilisent deux types de distances : la distance euclidienne et un opérateur multiplicatif (défini par $d = 1 - f_1 f_2$, où f_i représente le vecteur d'apparence). Les meilleurs résultats sont obtenus pour les représentations à base de gradients.

Les résultats de [10] nous ont confortés dans le choix d'une représentation à base de contours orientés. Nous écartons cependant leur approche via apparence du fait qu'ils n'ont pas tenus compte de la présence potentielle d'occlusion (due à une barrière). Nous avons donc sélectionné et développé une approche robuste à des informations manquantes ou erronées. En traitement d'images et vision par ordinateur, une catégorie de méthodes a démontré leur efficacité vis-à-vis d'une telle difficulté : les méthodes à base de votes. La méthode que nous allons décrire par la suite est donc essentiellement une méthode de vote utilisant pour votants les pixels de contour orientés.

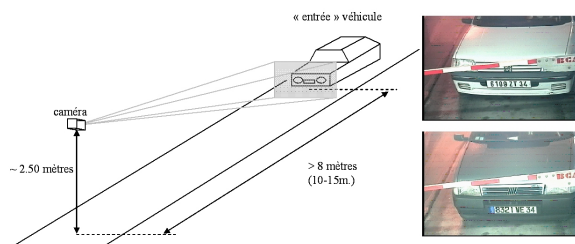


FIG. 2 – Exemple de système de contrôle d'accès.

Parmi les méthodes utilisant des représentations à contours orientés pour la reconnaissance d'objets, il faut distinguer celles qui les utilisent pour caractériser la forme globale de l'objet [9] ou un ensemble de formes simples [7] appartenant à l'objet, de celles qui les utilisent pour caractériser localement des points ou des zones d'intérêts sur l'objet [6, 2]. Certaines méthodes se placent à la confluence des 2 catégories en caractérisant chaque point selon la forme

globale de l'objet [1].

Nous situons nos travaux dans la première catégorie : c'est l'ensemble des votants (i.e. des points de contours) qui, d'une certaine façon, caractérise la forme d'objet. La caractérisation locale de chaque pixel de contours orientés est réduite au minimum. Ceci est motivé par le fait que dans les méthodes de votes, c'est le nombre qui fait la force : si la caractérisation des points d'intérêts (i.e. des votants) est trop sélective, le nombre de votants diminue. Par contre, nous allons proposer de pondérer chaque point en fonction de son potentiel discriminatoire. La section qui suit, expose le processus de création des modèles à base de points de contour orientés, et le calcul de cette pondération.

Dans la suite de cet article, nous allons décrire la construction de notre modèle de représentation de chaque type de véhicules à reconnaître, ainsi que notre fonction de classification. Cette fonction combine une mesure de distance et des scores des méthodes de votes. Par ailleurs, nous proposons une construction en cascade afin de réduire la confusion entre les modèles. Plusieurs résultats, démontrant l'efficacité de notre méthode seront exposés. Enfin, la conclusion de cet article sera l'occasion de proposer quelques perspectives pour des travaux futurs.

2 Création du modèle

Dans cette section, nous décrivons comment nous construisons une représentation unique pour chaque type de véhicules. Nous appelons ici cette représentation, modèle. Nous baptisons Base de Connaissance, la liste des classes que le système est capable de reconnaître. Notre Base de Connaissance est composée de K classes de véhicules ; nous notons cet ensemble de classes $\Omega(K)$.

2.1 Obtention du prototype



FIG. 3 – (a) image originale, (b) prototype I .

A partir de l'image originale d'une voiture, nous obtenons une imagette de taille 252x600 où la plaque minéralogique est placée dans une position connue (cf. figure 3) via une transformation affine. Nous considérons que le plan de la calandre et celui défini par les 4 coins de la plaque sont confondus. Les coins de la plaque sont obtenus via une méthode développée par LPREditor (pour plus d'informations, voir <http://www.lpreditor.com>).

2.2 Pixels de contours orientés

Pour le calcul des contours orientés, nous utilisons des filtres de Sobel de taille 3x3 sur l'image prototype I .

Une matrice \mathbf{E}_I représentant les contours du prototype I est déterminée après une opération de seuillage sur les valeurs du module du gradient. La valeur du seuil est obtenue en éliminant 85 % des pixels (les valeurs les plus faibles du module).

Dans la suite, nous nommerons $\mathbf{p} = [x, y]$ les pixels de contour appartenant à \mathbf{E}_I , où (x, y) est la position du point. Les orientations de ces pixels sont quantifiées pour qu'ils prennent des valeurs entières entre 0 et $N-1$ (ici N est égal à 4). Pour gérer les cas des voitures du même type mais de différentes couleurs (qui changent la polarité du contour), nous avons utilisé le module π au lieu du module 2π [2]. $\phi(\mathbf{p})$ désignera l'orientation quantifiée du gradient de I au point \mathbf{p} .

2.3 Un modèle unique

Pour chaque type de véhicule, nous allons dans cette section déterminer un modèle unique et représentatif de sa classe à partir de toutes les imagettes de véhicules de ce type. En effet, une classe k est représentée dans la Base d'Apprentissage par n prototypes. Cette quantité varie d'une classe à l'autre (voir le tableau 1).

Soient n prototypes pour la classe $k \in \mathbf{KnB}$
 pour $i=1, \dots, n$
 obtenir \mathbf{E}_i
 pour $j=1, \dots, n \wedge j \neq i$
 $A_{i,j}(600, 252, N) = 0$
 détermine \mathbf{E}_j
 $\forall \mathbf{p}_i \in \mathbf{E}_i$, trouver $\mathbf{p}_j \in \mathbf{E}_j / \phi(\mathbf{p}_i) = \phi(\mathbf{p}_j) \wedge \min_{\forall j} |\mathbf{p}_i \mathbf{p}_j| < t$

$\mathbf{m}(x_m, y_m) = \frac{\overline{\mathbf{p}_i \mathbf{p}_j}}{2}$
 ajout d'une gaussienne centrée en \mathbf{m} dans $A_{i,j}^{\phi(\mathbf{p}_i)}$

$A^k = \sum_{i,j} A_{i,j}$
 $\mathbf{M}^k = f_{max}(A^k)$

FIG. 4 – Pseudocode pour la création du modèle.

Le pseudocode contenu dans la figure 4 décrit la procédure employée. Pour chaque classe, nous prenons un couple (i, j) parmi les n prototypes de la classe k . Les matrices de contours orientés $(\mathbf{E}_i, \mathbf{E}_j)$ sont alors évaluées. Nous définissons un accumulateur $A_{i,j}$ pour chaque couple. Ensuite, en prenant un point \mathbf{p}_i de \mathbf{E}_i , on cherche dans \mathbf{E}_j le plus proche \mathbf{p}_j avec la même orientation. En respectant l'orientation du gradient, chaque accumulateur $A_{i,j}$, limité au plan d'orientation $\phi(\mathbf{p}_i)$ (noté $A_{i,j}^{\phi(\mathbf{p}_i)}$ dans le pseudo-code) est incrémenté via un noyau gaussien, centré au point milieu \mathbf{m} du segment $\overline{\mathbf{p}_i \mathbf{p}_j}$. Itérativement, ce procédé est répété pour tous les points \mathbf{p}_i de \mathbf{E}_i , puis pour tous les couples possibles de l'ensemble de n exemples de la classe k . Enfin, une matrice d'accumulation de la totalité des votes de la classe k est obtenue en faisant l'addition de tous les $A_{i,j}$:

$$A^k = \sum_{i,j \wedge i \neq j} A_{i,j}$$

Cet accumulateur revient à une estimation de la densité des

points via des fenêtres de Parzen[5]. L'étape de vote pour le point milieu permet d'avoir une meilleure estimation des densités malgré un nombre faible d'échantillons (souvent nous disposons de moins de 10 images par modèle) tout en conservant des covariances faibles au niveau du noyau (ce qui permet de conserver une certaine précision sur la forme).

Il distingue les points de contour les plus redondants dans les n prototypes, donc les plus représentatifs de la classe k , selon la quantité de votes. Pour obtenir un modèle unique (composé de points de contour orientés) de la classe k , on cherche itérativement le point $\mathbf{a}_m = [x, y, o]$ le plus voté de A^k , en faisant attention que les points soient distants d'au moins 5 pixels afin que leur distribution soit relativement homogène. Nous conservons la position et l'orientation du maximum \mathbf{a}_m dans une matrice \mathbf{M}^k (voir figure 5). La méthode est réitérée jusqu'à obtention du nombre de points voulu (ici 3500 points).

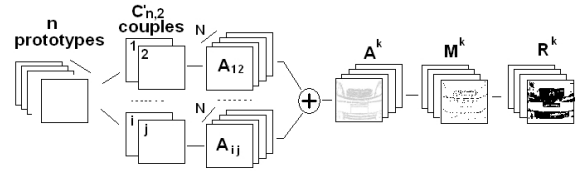


FIG. 5 – Création du modèle pour la classe k .

Quand il existe un seul exemple dans la base d'apprentissage, il n'est plus possible de créer l'accumulateur de votes A^k . Il faut donc utiliser une autre source d'information qui nous estime un poids pour chaque pixel de contour. Dans ce cas, l'accumulateur A^k est remplacé par le module du gradient dans le procédé de génération du modèle.

2.4 Matrices de pondération

Notre algorithme de classification est essentiellement basé sur plusieurs calculs de votes. Un de ceux-ci repose sur le principe suivant : chaque point de contour de l'exemple de test \mathbf{t} vote pour un modèle s'il tombe dans le voisinage d'un point de ce modèle. Les autres sont construits selon ce modèle. Ils requièrent donc la construction de cartes de voisinage.

Pour estimer ces voisinages, la distance de Chamfer est appliquée à \mathbf{M}^k pour déterminer les cartes des distances D_i^k par rapport à ses éléments (où i représente l'orientation et k la classe de voiture). La figure 6 montre les quatre R_i^k matrices binaires obtenues après le seuillage de la matrice D_i^k en gardant les distances plus petites que r ($r = 5$). Les R_i^k sont regroupées par concaténation dans une seule matrice R^k , de dimension $600 \times 252 \times N$.

Afin d'optimiser la méthode de votes, nous allons calculer un poids w pour chaque point des régions définies dans R^k : les points les plus discriminants du modèle de \mathbf{p}_m par rapport aux autres modèles, vont avoir un vote de poids plus important. Ces poids sont regroupés dans une matrice W_+^k . Les pixels du modèle k qui sont rarement présents

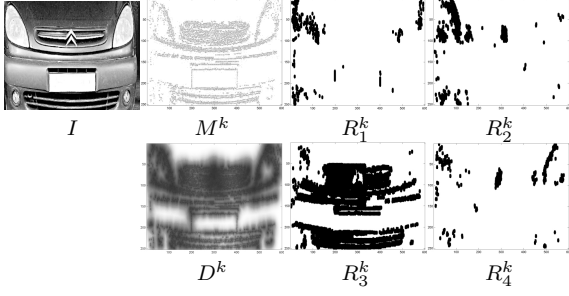


FIG. 6 – M^k est le modèle de points de contour orientés avec les N orientations superposées. D^k est la carte de distance de Chamfer de M^k . Les R_i^k sont les voisinages après application d'un seuil r pour chaque orientation du gradient i .

dans les autres classes obtiennent des poids plus importants. Des poids faibles sont donnés aux points présents dans la plupart des autres classes. Ceci peut s'exprimer via l'équation suivante :

$$W_+^k = \frac{1}{K-1} \sum_{i, i \neq k} (R^k - R^i \cap R^k)$$

De la même façon que nous récompensons les modèles proches de \mathbf{t} par le vote des points qui tombent dans les voisinages, nous pouvons aussi punir ceux qui ne le sont pas (via un vote des points qui ne tombent pas dans les voisinages). En considérant alors des poids négatifs, nous pouvons construire une matrice de pondération W_-^k :

$$W_-^k = -\frac{1}{K-1} \sum_{i, i \neq k} (R^i - R^i \cap R^k)$$

3 Classification

La classification consiste à associer un exemple \mathbf{t} dit de test à une classe k du dictionnaire des classes. Ceci sera réalisé via une fonction de discrimination. Elle se présente comme la combinaison de scores provenant de deux types de classifieurs. Le premier type consiste en trois processus de vote : les votes positifs de \mathbf{t} vers k , les votes négatifs de \mathbf{t} vers k et les votes de chacun des k vers le \mathbf{t} . Le deuxième classifieur évalue, pour chaque k , l'erreur en distance de mise en correspondance entre les points de contours orientés des modèles M^k et les points de \mathbf{t} .

3.1 Les classifieurs

Nous appliquons à \mathbf{t} les mêmes opérations que lors de la création des modèles. Nous obtenons donc une matrice des points de contour orientés \mathbf{E}_t . Parmi ces points, nous en sélectionnons T (ici $T = 2000$) qui sont a priori les plus discriminants pour chaque classe. Ceci est réalisé en triant les points de \mathbf{E}_t selon la matrice de pondération W_+^i avec $i = 1, \dots, K$. Nous obtenons alors pour l'exemple \mathbf{t} , une matrice \mathbf{P}_t de taille $600 \times 252 \times N$ et construite telle que \mathbf{P}_t a la valeur 1 dans la position des points sélectionnés et 0 ailleurs.

Votes positifs Le score dit de votes positifs est incrémenté si un point de \mathbf{P}_t tombe dans le voisinage d'un point de M^k . Le voisinage d'un point d'un modèle M^k est défini comme un cercle de rayon r ; nous rappelons que ces voisinages sont en fait définis par le seuillage de la carte de distance D^k . Ceci peut être réalisé simplement en faisant le produit terme à terme entre \mathbf{P}_t et W_+^k :

$$v_+^k = \sum_x \sum_y \sum_o P_t \bullet W_+^k$$

Votes négatifs Les votes négatifs prennent en compte les points du \mathbf{P}_t qui ne sont pas tombés dans le voisinage de M^k . Nous punissons la classe k en accumulant ces points (pondérés par la matrice W_-^k) :

$$v_-^k = \sum_x \sum_y \sum_o P_t \bullet W_-^k$$

Votes vers le test Ensuite, nous allons calculer les votes des modèles vers l'exemple de test. Sommairement, la méthode est la même que celle détaillée dans la section précédente. Nous construisons la matrice des Distances de Chamfer pour \mathbf{E}_t . Après un seuillage des points qui se trouvent à une distance inférieure à r , nous obtenons la matrice des régions R^t de l'exemple de test. Par la suite, nous prenons les premiers T points de la liste M^k et construisons la matrice P^k . Chaque point de la matrice P^k est pondérée avec W_+^k :

$$v_+^t = \sum_x \sum_y \sum_o R^t \bullet P^k \bullet W_+^k$$

Erreur de distance Le dernier score est issu de l'erreur en distance de mise en correspondance entre les points de \mathbf{P}_t avec les points plus proches de M^k pour la classe k . Elle est connue comme la distance de Hausdorff modifiée [4]. Celle-ci s'exprime de la manière suivante :

$$H(\mathbf{P}_t, M^k) = \max(h(\mathbf{P}_t, M^k), h(M^k, \mathbf{P}_t))$$

avec :

$$h(\mathbf{P}_t, M^k) = \text{moyenne}_{a \in \mathbf{P}_t} (\min_{b \in M^k} \|a - b\|)$$

Afin de réaliser la fusion avec les autres scores, nous y appliquons une fonction décroissante pour attribuer un score maximum dans le cas d'une erreur minimale.

Fonction de discrimination Les quatre scores $\{v_+^k, v_-^k, v_+^t, d^k\}$ sont combinés dans la fonction discriminante $g_k(t)$ en une mesure de concordance entre l'exemple de test \mathbf{t} et la classe k .

Avant la fusion, une pseudo-distance de Mahalanobis est utilisée pour normaliser les valeurs des scores : par exemple, pour le vote positif, $v_{+k}^{mh} = (v_+^k - \mu_+)/\sigma_+$, où (μ_+, σ_+) représentent respectivement la moyenne et l'écart-type des votes sur l'ensemble des modèles. La fonction de discrimination est définie alors par :

$$g_k(t) = \alpha_1 v_{+k}^{mh} + \alpha_2 v_{-k}^{mh} + \alpha_3 v_{+t}^{mh} + \alpha_4 d_k^{mh} \quad (1)$$

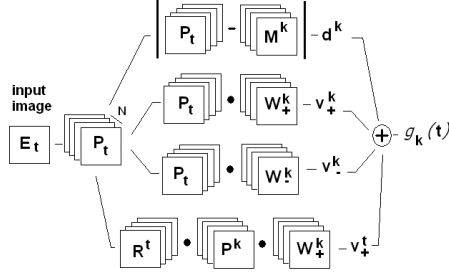


FIG. 7 – Obtention de la fonction de discrimination

Les coefficients α_i peuvent donner des poids différents aux classificateurs. Dans notre système, nous fixons tous les α_i à 0.25. La classe de l'exemple de test t est la classe qui obtient le plus grand score pour la fonction de discrimination :

$$t \in c/c = \text{ArgMax}_{k=1..K} g_k(t)$$

Analogie En fait, la méthode décrite ici via un algorithme de votes peut trouver une justification en utilisant la théorie des probabilités et les modèles non paramétriques basés sur les fenêtres de Parzen, où la fonction noyau est une fonction uniforme sur le voisinage de chaque point et nulle ailleurs. Avec ce formalisme, R_i^k est proportionnel à une fonction de densité, $P(\mathbf{p}|k)$, des points \mathbf{p} d'orientation i pour chaque type k . La matrice de pondération positive introduit alors une notion de fonction de densité, $P(\mathbf{p}/\mathbf{p} \notin \cup_{l \neq k} l|k)$, des points de k n'appartenant pas aux autres types, $\{l\}_{l \neq k}$. Celle négative, la fonction de densité complémentaire $P(\mathbf{p}/\mathbf{p} \notin \cup_{l \neq k} l|k)$. Enfin, l'erreur en distance rejoint ce formalisme en considérant alors un noyau de forme triangulaire. Les scores sont alors équivalents à des probabilités $P(k)$ obtenus via la règle de Bayes.

3.2 La cascade de classificateurs

Afin de réduire la confusion entre les modèles, nous introduisons dans ce papier, une construction en cascade à plusieurs étages. La discrimination entre les modèles est principalement introduite par la position géométrique des points de contours choisis et par la pondération de ceux-ci. Cette sélection et cette pondération sont établies via les matrices W_*^i avec $*$ = $\{+, -\}$ et i le numéro du modèle considéré (cf. section 2.4). Pour chaque modèle, ces matrices sont calculées vis-à-vis des autres modèles considérés : moins on considère de modèles, plus la pondération sera discriminante vis-à-vis de ceux-ci. Aussi dans notre construction en cascade, nous allons réduire le nombre de modèles : à chaque étage, nous classifions notre image de test à un sous-ensemble $\Omega(K_{x+1})$ de $\Omega(K_x)$ où K_x est le nombre de classes ($K_{x+1} < K_x$ et $K_1 = K$) présentées à l'étage x . Comme le calcul des matrices de pondérations consiste en un simple calcul matriciel, il est envisageable de le réaliser *on-line* et donc de réaliser le choix des sous-ensembles de manière dynamique. La figure 8 illustre ce processus.

Ainsi $\Omega(K_x)$ sera composé des modèles k dont les scores

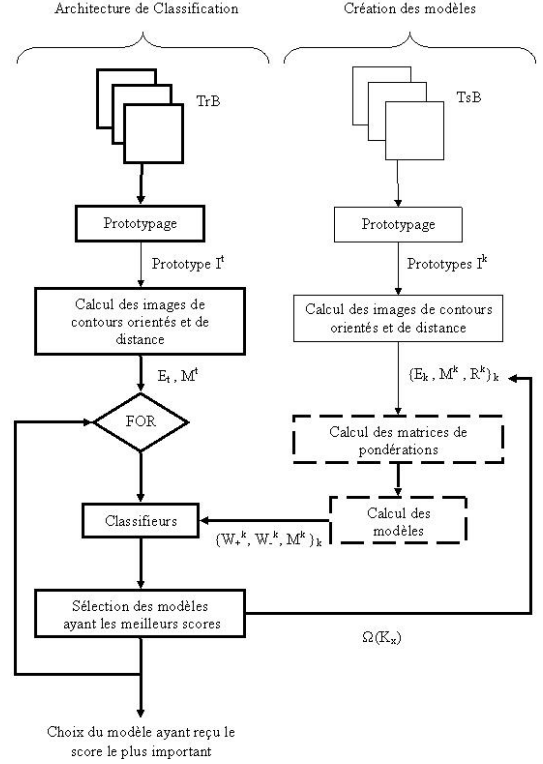


FIG. 8 – Synopsis de la cascade des classificateurs

$g_k^{x-1}(t)$ obtenus via l'étage précédent, $x - 1$, les placent à un rang inférieur à K_x . Le rang 1 correspondant au score maximum, le rang 2 au second meilleur score, ... Cette stratégie est confortée par l'observation des résultats que nous avons obtenu précédemment pour le classifieur simple (à un seul étage). En effet, nous avons constaté que le score pour le type du véhicule testé le place très fréquemment dans les rangs les plus bas. Cette observation est confirmée par la courbe *Cumulative Match Characteristic* (CMC¹) de la figure 13.

Lors du fonctionnement du système, un exemple test ne sera soumis à l'étage supérieur que si son score est relativement faible ; un seuil sur le score sera alors fixé à chaque étage. Bien entendu, un autre critère d'arrêt sera le nombre maximum d'étages.

4 Résultats et Discussion

Dans cette section, nous allons commenter plusieurs résultats obtenus à l'aide du système décrit précédemment. Ces résultats ont été essentiellement réalisés sur 2 Bases de Connaissances différentes : une composée de 20 modèles

¹La courbe CMC est utilisée pour mesurer la performance d'un système d'identification (on utilise aussi le terme de reconnaissance 1 : n). Cette courbe donne le taux de reconnaissance (ici en micro-précision) en fonction du rang. On dit alors que le système reconnaît au rang 1 lorsqu'il choisit le modèle au meilleur score comme résultat de la reconnaissance. On dit qu'un système reconnaît au rang 2, lorsqu'il choisit le type du véhicule de l'image de test appartenant aux 2 modèles ayant obtenu les 2 meilleurs scores, etc.

(cf. tab. 1) et une autre de 50 modèles. La première servira essentiellement à illustrer le fonctionnement du classifieur sans l'utilisation de cascade (aussi appelé classifieur à 1 étage). La seconde nous permettra d'analyser plus finement l'apport de la cascade. Enfin, suite à cette analyse, nous discuterons des perspectives possibles afin d'améliorer les performances du système.

4.1 Bases d'Images

Nous disposons de deux bases d'images en niveaux de gris des voitures vues de face : une Base d'Apprentissage (TrB) et une Base de Test (TsB). Afin de tester notre approche dans des conditions proches de celles industrielles, nous avons réalisé deux protocoles différents pour la collecte des échantillons qui vont constituer nos bases. Ainsi les images de la TrB ne sont pas prises dans les mêmes conditions que celles de la TsB. En effet, dans l'optique d'une application commerciale, il n'est pas envisageable que la TrB soit réalisée sur le site d'implantation du système ; aussi l'apprentissage doit être réalisé sur une TrB indépendante du site, et donc de la TsB. La Base d'Apprentissage est composée des images prises dans des parkings avec des appareils numériques. La résolution de ces images varie entre 1280x960 pixels et 1296x976 pixels. Nous avons choisi de se servir de ces images pour construire les modèles des types de véhicules, du fait de leur résolution de qualité supérieure. La Base de Test est composée d'images prises avec un caméscope numérique. La résolution des images de cette base va de 640x480 pixels à 720x576 pixels. Les images sont prises sous différents points de vue et sous différents éclairages. La première ligne de la figure 9 présente des exemples de la Base d'Apprentissage et la deuxième ligne les exemples de la Base de Test de la même classe.



FIG. 9 – Exemples de la Base d'Apprentissage et de la Base de Test.

4.2 Résultats sans la cascade

Pour la Base de Connaissance de 20 modèles, nous avons réuni 173 images d'apprentissage et 480 de test, dont la répartition en 20 classes de type de véhicules est donnée dans le tableau 1.

Notre algorithme identifie correctement 93.75 % des 480 exemples, ce qui correspond au taux de reconnaissance en micro-précision [12] ; le taux de reconnaissance en macro-précision, c'est-à-dire la moyenne des taux de reconnaissance individuels (pour chaque classe), est de 94.07%.

Afin de comparer notre système avec l'existant, nous avons reproduit sur notre base, quelques unes des méthodes décrites dans [10] parmi celles obtenant les meilleurs scores.

Type de véhicule	Citroen Berlingo A	Citroen C3	Citroen Picasso B	Citroen Saxo B	Ford Focus A	Peugeot 206 B	Peugeot 307	Peugeot 405	Peugeot 406 B	Renault 19 B	Renault Clio A	Renault Clio C	Renault Clio D	Renault Laguna B	Renault Scenic B	Renault Scenic C	Renault Twingo A	Renault Twingo B	VW Golf C	VW Golf D	Total
TrB	1	1	4	9	5	10	12	3	6	4	7	15	32	5	5	5	19	21	4	5	173
TsB	11	20	19	21	13	21	28	17	23	20	21	37	62	22	20	14	33	33	22	23	480

TAB. 1 – Répartition de la Base de Connaissance avec 20 modèles.

Les résultats obtenus sur notre base (cf. tab. 2) situent les performances de notre méthode au dessus de celles présentées dans [10]. Remarquons que Petrovic et al. obtenaient pourtant des taux de reconnaissance plus élevés sur une base composée de plus de types de véhicules ; nous expliquons cette différence par le fait que, d'après leur papier, les images composant leur base de données ont été prises dans des conditions plus favorables (bien en face et proche du véhicule, à la même hauteur que la plaque d'immatriculation, avec le plan image parallèle au véhicule,...) que les nôtres (prises dans des conditions plus réalistes vis-à-vis des applications visées : de biais, avec une caméra éloignée, etc - cf. fig. 9).

représentation	Résultats de [10] sur notre base	Résultats de [10] sur leur propre base
<i>Sobel Edge responses</i>	58.2%	75.0%
<i>Direct normed grads</i>	76.5%	84.5%
<i>Square mapped grads</i>	52.6%	97.7%

TAB. 2 – Taux de reconnaissance en micro-précision de quelques unes des méthodes décrites dans [10] obtenus sur notre base à 20 modèles et donnés dans [10] sur leur propre base à 77 types de véhicules.

Dans un autre test, nous avons simulé la présence de la barrière en quatre positions différentes. La figure 10-a montre les différentes positions de celles-ci. Les résultats obtenus pour chaque position de la barrière sont exposés dans la figure 10-b. Le taux de reconnaissance plus faible pour la 4^{ème} position peut s'expliquer par le fait que la barrière cache les points proches à la plaque minéralogique : ce sont des points importants pour la classification et sont aussi les plus stables à la transformation affine.

Position	Taux de reconnaissance
1	91.25 %
2	91.88 %
3	90.92 %
4	88.33 %

FIG. 10 – (a) les 4 positions de la barrière virtuelle, (b) Taux de reconnaissance.

Nous invitons le lecteur intéressé à consulter le papier [8] qui présente d'autres résultats ainsi qu'une caractérisation du classifieur à un seul étage en fonction des paramètres du système (quantité de points, quantités d'orientation, rayon de voisinage,...). Cette caractérisation montre notamment que la méthode est relativement insensible à ces paramètres sur des plages assez importantes.

4.3 Résultats avec la cascade

Dans les prochains paragraphes, nous allons analyser les résultats spécifiquement obtenus avec la cascade. Cette cascade a été testée sur une Base de Connaissances avec 50 modèles. Cette base est composée de 288 images pour la Base d'Apprentissage et de 830 pour la Base de Test. La cascade est composée de 4 étages au maximum ; à chaque étage, nous divisons le nombre de modèles à considérer par deux (ainsi $K_x \in \{50, 25, 13, 7\}$).

étages	1	2	3	4
taux en micro-préc.	80.2	83.6	83.1	83.9
taux en macro-préc.	69.4	72.8	72.1	73.1

TAB. 3 – Evolution des taux de reconnaissance en fonction du nombre d'étages dans la cascade.

Le tableau 3 montre les résultats obtenus à chaque étage (afin d'avoir une bonne représentation, nous soumettons toutes les images tests à tous les étages, sans le critère d'arrêt sur le score). Nous constatons que globalement nous gagnons plus de 3% sur les taux de reconnaissance en utilisant la cascade. Ce gain peut paraître faible. Cependant en analysant plus finement les résultats, nous pouvons dire que la cascade remplit sa fonction : celle de réduire la confusion entre les types de véhicules proches.

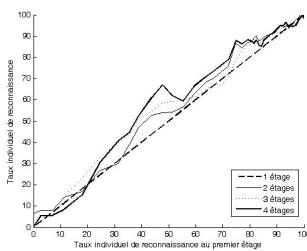


FIG. 11 – Evolution des taux de reconnaissance individuels selon les étages

En effet, en observant l'évolution des taux de reconnaissance individuels selon les modèles (représentée dans la figure 11 en fonction du taux obtenu au premier étage), nous pouvons remarquer que l'apport de la cascade est plus important pour les modèles n'obtenant que des taux de reconnaissance situés entre 25% et 80% sans la cascade, atteignant même un gain supérieur à plus de 20% au bout de 4 étages pour certains modèles. C'est en effet à leur niveau qu'il y a le plus de confusion avec d'autres modèles.

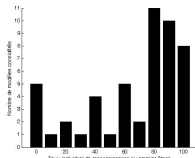


FIG. 12 – Répartition du nombre de modèles considérés en fonction des taux de reconnaissance individuels obtenus au premier étage

Mais comme il ne s'agit pas de modèles majoritairement présents dans notre Base de Connaissances (cf. figure 12), leur influence est moindre sur les taux globaux.

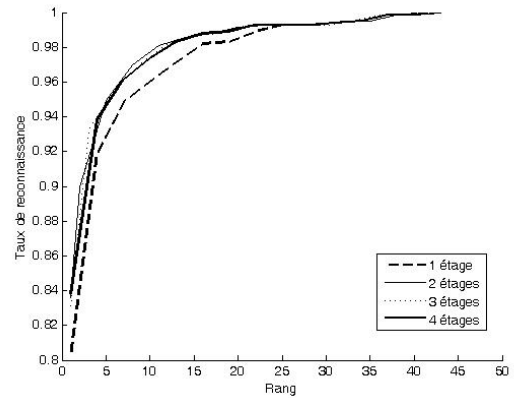


FIG. 13 – Courbe *Cumulative Match Characteristic*

Par ailleurs, nous constatons sur la figure 13 que la cascade améliore nettement la CMC. Ce qui est particulièrement intéressant car en effet, l'objectif final est de fusionner l'information obtenue sur le type avec la lecture de la plaque d'immatriculation. Aussi le système devra peut-être validé plus d'un modèle. Nous constatons aussi que les courbes à 2, 3 et 4 étages sont quasiment confondues, ce qui laisse à penser que seul un système réduit à étages serait vraiment utile. Cependant, ici nous divisons par deux le nombre de modèles considérés à chaque étage : avec une portion de classes retenues plus importante, plus d'étages pourraient être nécessaires.

4.4 Discussion et perspectives

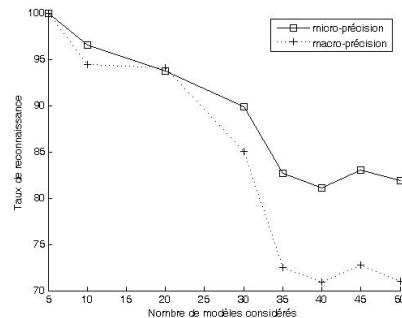


FIG. 14 – Taux de reconnaissance en fonction de la taille de la Base de Connaissances, et en ne considérant qu'un seul étage

Nous avons vu que le système de reconnaissance obtient des résultats intéressants. Cependant, nous constatons aussi une forte baisse de ces performances lors que le nombre de modèles augmente dans notre Base de Connaissances, comme l'indique aussi la figure 14. En fait, cela est essentiellement dû à l'introduction de types de véhicules repré-

sentés par un faible nombre d'images (< 5) dans la base d'apprentissage. Dans la Base de Connaissances à 20 modèles, ils représentent moins de 20% des types. Dans la base à 50 modèles, ils représentent 46% des types. Ceci influence fortement les taux de reconnaissance (cf. figure 15). En effet, le choix des points de contours caractéristiques est issu d'un processus d'accumulation entre les images d'apprentissage (cf. sections 2.2 et 2.3). Pour que celui-ci filtre efficacement les points de contours issus du bruit ou de reflets sur la carrosserie (cf. figure 16), il faut un nombre minimum d'images pour chaque type dans la TrB.

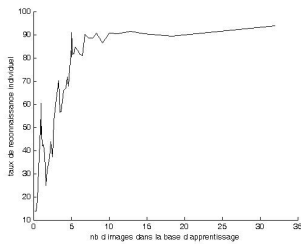


FIG. 15 – Evolution moyenne des taux de reconnaissance individuels selon le nombre d'images par modèle dans la TrB

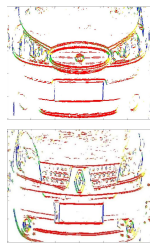
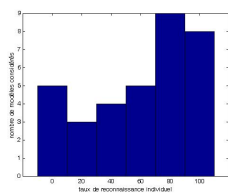
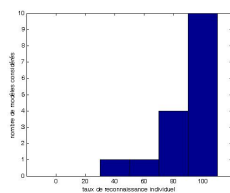


FIG. 16 – Exemples d'extraction de contours orientés

Ceci est corroboré par un autre test où nous avons entraîné le système avec un seul exemple par classe dans la TrB limitée à 20 modèles. Le taux de reconnaissance alors obtenu est 85.6%. Les figures 15 et 17-b indiquent qu'il faut que ce nombre soit supérieur à 5 images pour que les taux individuels soient intéressants. Remarquons cependant que même avec peu d'images dans la TrB, certains types de véhicules obtiennent des taux de reconnaissance importants (cf. figure 17-a). Aussi nous pensons que un processus intra-image² de filtrage en plus de celui inter-images proposé, serait suffisant pour améliorer nettement nos résultats. Ceci sera l'objet de nos prochains travaux.



(a) avec moins de 5 images



(b) avec plus de 5 images

FIG. 17 – Répartitions du nombre de modèles considérés (représentés dans la Base de Connaissances par moins de 5 images ou par plus de 5 images) en fonction des taux de reconnaissance individuels (obtenus au premier étage)

²Par exemple, avec une étude locale au voisinage de chaque point de contours orientés (mesure d'entropie,...).

5 Conclusion

Cette article a présenté un système de votes pour une application de reconnaissance multi-classes du type de véhicule basé sur une représentation en points de contours orientés. Une fonction discriminante combine les scores obtenus de trois types de classifieurs basés sur les votes et une erreur en distance. Une mise en cascade de ces classifieurs nous permet de réduire la confusion entre les modèles. Nous avons testé cette méthode sur plusieurs bases d'images prises en conditions réelles (composées de 5 à 50 classes de type de véhicules). Les taux de reconnaissance sont similaires (voire supérieurs) à ceux obtenus par d'autres méthodes [10]. Nous avons montré aussi que le système est robuste vis-à-vis des occlusions partielles de l'image. Nos travaux futurs s'attacheront à améliorer la construction des modèles, en filtrant davantage les points de contours orientés, de sorte à éliminer ceux introduits par le bruit ou les reflets sur la carrosserie du véhicule. L'amélioration du processus de cascade est aussi à envisager : certains choix ont été réalisés de manière arbitraire (nombre d'étages, portion de classes retenues, nombre de points de contours gardé constant d'étage en étage). Une étude plus poussée de ces paramètres serait intéressante et permettrait peut-être d'en proposer une automatisation. Enfin nous pourrions nous intéresser au mécanisme de fusion des informations issues de notre système de reconnaissance et de la lecture de la plaque d'immatriculation, tel que l'illustre la figure 1.

Références

- [1] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4) :509–522, 2002.
- [2] T. Cootes and C. Taylor. On representing edge structure for model matching. In *CVPR*, volume 1, pages 1114–1119, Hawaii, USA, December 2001.
- [3] J. Douret and R. Benosman. A multi-cameras 3d volumetric method for outdoor scenes : a road traffic monitoring application. In *ICPR04*, pages III : 334–337, 2004.
- [4] MP. Dubuisson and A. Jain. A modified hausdorff distance for object matching. In *International Conference on Pattern Recognition*, volume A, pages 566–569, 1994.
- [5] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification (2nd ed.)*. John Wiley and Sons, 2001.
- [6] D. Hond and L. Spacek. Distinctive descriptions for face processing. In *BMVC*, University of Essex, UK, 1997.
- [7] F. Jurie and C. Schmid. Scale-invariant shape features for recognition of object categories. In *Computer Vision and Pattern Recognition*, Washington, DC, June-July 2004.
- [8] P.A. Negri, X. Clady, M. Milgram, and R. Poulencard. Reconnaissance par vision du type d'un véhicule automobile. In *Colloque MajecStic*, 2006.
- [9] C. F. Olson and D. P. Huttenlocher. Automatic target recognition by matching oriented edge pixels. *IEEE Transactions on Image Processing*, 6(1) :103–113, 1997.
- [10] V.S. Petrovic and T.F. Cootes. Analysis of features for rigid structure vehicle type recognition. In *BMVC*, volume 2, pages 587–596, 2004.
- [11] Z. Sun, G. Bebis, and R. Miller. On-road vehicle detection : A review. *PAMI*, 28(5) :694–711, May 2006.
- [12] C. J. Van Rijsbergen. *Information Retrieval*. Butterworths, London, Second Edition, 1979.