Segmentation d'itinéraires en catégories de vitesse maximale autorisée par analyse d'images

Philippe FOUCHER, Emmanuel MOEBEL, Pierre CHARBONNIER

Cerema, DTer Est, Laboratoire Régional de Strasbourg, ERA 27
11, rue Jean Mentelin, BP9, 67035 Strasbourg, France
philippe.foucher@cerema.fr, emmanuel.moebel@inria.fr,
pierre.charbonnier@cerema.fr

Résumé – Dans cette contribution, nous proposons une méthode en deux étapes pour segmenter un itinéraire routier en tronçons homogènes de catégories de limites de vitesse. Dans une première étape, l'algorithme de classification *Random forest* permet d'identifier la catégorie de l'image. Les attributs d'entrée du classifieur sont des descripteurs bas-niveau (mCentrist) ou des caractéristiques haut-niveau issues d'une représentation simplifiée de l'image. Dans un second temps, nous proposons un lissage des résultats de classification d'une image en considérant les résultats des images voisines dans la séquence. Les performances des algorithmes sont analysées et comparées sur une séquence d'images.

Abstract – This contribution proposes a two-step algorithm to segment routes into speed limit categories. Firstly, the images are classified into a category by using the *Random forest* classifier. The input features of the classifier are either low-level (mCentrist) descriptors or high-level descriptors. In a second step, a sequential smoothing of the classification is applied. Finally, the performances of the algorithms are analyzed and compared on an image sequence.

1 Introduction

Le respect des limitations de vitesse par un usager passe par une bonne lisibilité de la section routière empruntée en termes de vitesse maximale autorisée (VMA). La mise en place des limites de vitesse doit ainsi non seulement répondre à des critères administratifs, techniques et de sécurité définis par le gestionnaire et/ou le législateur mais doit également être en cohérence avec la perception de l'infrastructure routière et son environnement. Localiser et évaluer d'éventuelles incohérences entre les deux (VMA administratives et VMA « lisibles » selon l'infrastructure) est un réel besoin pour le gestionnaire. L'utilisation de techniques d'analyse d'images nous semble pertinente pour la mise au point d'outils automatiques d'évaluation de la lisibilité.

Dans cet article, nous nous intéressons à la segmentation d'itinéraires en sections homogènes de vitesse maximale autorisée (VMA). Nous nous focalisons sur 4 catégories de VMA : 50 km/h (zones urbaines) ; 70 km/h (zones dangereuses) ; 90 km/h (routes bi-directionnelles) et 110 km/h (2×2 voies avec présence d'une terre-plein central). La première contribution de l'article est la mise au point d'un algorithme de catégorisation de scènes en deux étapes : La première phase consiste à classer chaque image, prise individuellement, dans l'une des quatre catégories sémantiques. La classification des images sans considérer le caractère temporel des prises de vues sur l'itinéraire n'est pas forcément pertinente. Les images étant acquises sous forme de séquences, avec une image tous les 5 mètres, nous proposons d'introduire un lissage des résultats de classification en considérant les catégories des images avoisinantes. La deuxième contribution de l'article consiste à effectuer une évaluation systématique des résultats sur une séquence d'images. Ces travaux reprennent en partie les résultats présentés dans [3]. L'article est organisé de la façon suivante : dans la section 2, un bref état de l'art des travaux en lien avec cet article est proposé. La méthode de segmentation d'itinéraires est présentée Sect. 3. Le protocole expérimental est ensuite décrit Sect. 4 et les résultats sont détaillés Sect. 5. Enfin, une conclusion et quelques perspectives sont proposées en Sect. 6.

2 État de l'art

La catégorisation d'une scène routière par des outils de reconnaissance de formes n'est apparue que récemment dans la littérature. Dans [7], les auteurs ont développé un algorithme de classification en quatre types d'environnements routiers : chemin, route urbaine, route bi-directionnelle et 2×2 voies. Notons que, pour effectuer cette classification, les auteurs s'appuient sur des descripteurs de couleur et de texture calculés sur trois régions d'intérêt de l'image. Plus récemment, une méthode de reconnaissance de scènes routières en huit catégories a été proposée dans [6]. L'objectif des ces travaux est d'enrichir l'information GPS fournie à l'usager par l'ajout de données sémantiques à l'endroit où se positionne le véhicule. A notre connaissance, il n'existe pas de travaux relatifs à la catégorisation de scènes en fonction des *VMA*. Signalons cependant que la problématique est explorée dans [4] où les auteurs proposent de

FIGURE 1 – Exemples de catégories de limites de vitesse.

classer les scènes en catégories urbaines / non urbaines. Une analyse approfondie des résultats de classification est ensuite effectuée afin d'établir un lien entre la perception de la *VMA* par des sujets et la catégorie de scène obtenue de automatiquement. Dans cette contribution, nous proposons d'aller plus loin, puisque outre le fait que nous proposons un nombre de catégories plus important, nous effectuons un lissage des résultats de classification, afin d'obtenir une segmentation de l'itinéraire.

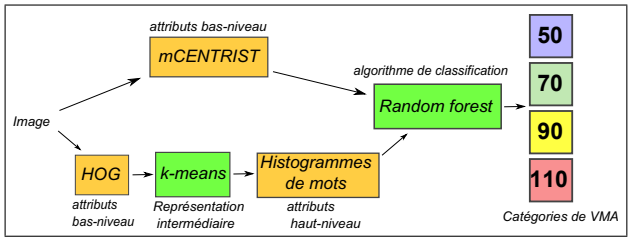

FIGURE 2 – Algorithmes de classification de scènes : approche bas-niveau (haut) et haut-niveau (bas).

3 Méthodologie

La méthode proposée suit un processus en deux étapes : la première phase permet de classer chaque image dans une catégorie de *VMA*. Un second traitement est effectué afin de lisser les résultats de classification en considérant l'aspect séquentiel.

3.1 Classification de scènes

La classification de scènes routières repose soit sur un algorithme bas-niveau soit sur un algorithme haut-niveau (voir Fig. 2). Dans les deux cas, la catégorie finale est obtenue à partir du classifieur *Random Forest* [1]. Ce méta-classifieur est un ensemble d'arbres décisionnels, qui permet d'obtenir une probabilité d'appartenance à une catégorie. Dans l'approche bas-niveau, les descripteurs mCentrist sont calculés directement sur l'image initiale. Dans la méthode haut-niveau, les attributs utilisés sont issus d'une représentation simplifiée de l'image.

3.1.1 Caractéristiques bas-niveau

Le descripteur Centrist [8] correspond à un histogramme des valeurs obtenues après la transformation Census [10], semblable au *Local binary Pattern* [5]. Cette transformée a pour but d'encoder, sous la forme d'un octet, les relations entre un pixel et ses huit voisins. Ainsi, un bit de l'octet est codé à la valeur 1 si le pixel central est supérieur ou égal au voisin considéré et à 0 dans le cas contraire. La comparaison avec chaque voisin se fait de gauche à droite puis de haut en bas :

$$\begin{array}{|c|c|c|}\hline 156 & 168 & 172 \\ \hline 156 & 156 & 157 \\ \hline 153 & 150 & 155 \\ \hline \end{array} \Rightarrow (10010111)_2 \Rightarrow CT = 151 \qquad (1)$$

L'information spatiale est également considérée en calculant les descripteurs Centrist sur différentes cellules, obtenues par division de l'image à différentes résolutions. La pyramide spatiale utilisée comprend $n_b = 31$ cellules. Le descripteur mCentrist [9] permet de prendre en compte l'information couleur, ignorée dans le descripteur Centrist. Dans ce descripteur, les composantes colorimétriques sont associées par paire pour former deux histogrammes de $d_{mCT} = 254$ niveaux, les valeurs extrêmes n'étant pas considérées (voir [9] pour plus de détails). Dans ces travaux, nous considérons une image à 4 composantes (3 composantes colorimétriques et la composante Sobel), soit $n_{pairs} = 6$ combinaisons possibles. La dimension du vecteur de caractéristiques est : $d_{mCentrist} = n_b \times 2 \times d_{mCT} \times n_{pairs}$.

3.1.2 Caractéristiques haut-niveau

Dans cette approche, une représentation simplifiée de l'image est générée. En reprenant le principe des méthodes *Bag-of-words* (BOW), l'image est dans un premier temps divisée en cellules rectangulaires de taille $n_c \times n_c$ et un histogramme de gradients orientés (*HOG*) [2] est calculé pour chaque cellule. L'algorithme des k-moyennes est utilisé pour associer à chaque cellule un mot en fonction du vecteur de caractéristiques. La classification finale dans une des quatre catégories de vitesse est obtenue en prenant l'histogramme des mots, de dimension n_mots en entrée du classifieur *random forest*. La taille de la cellule n_c, le nombre de mots n_{mots} et le nombre de cases n_{bins} dans l'histogramme sont déterminés empiriquement.

3.2 Segmentation d'itinéraires

La probabilité de changement de catégorie entre deux images successives étant faible, excepté pour les quelques transitions entre zones, il apparaît pertinent de lisser les résultats de la classification en considérant les images avoisinantes. Pour cela, nous considérons l'ensemble des probabilités *a posteriori* d'appartenance à une catégorie obtenue pour chaque image de la séquence et appliquons successivement un filtre moyenne et un filtre morphologique (fermeture et ouverture) à ces résultats. La taille du filtre, centré sur l'image courante, est $N_{filtre} =$

21 images. La classe maximisant la probabilité d'appartenance après filtrage donne la catégorie retenue.

4 Protocole expérimental

Les images sont acquises sous forme de séquences (1 image/5 mètres), de jour, au moyen de caméras frontales situées sur le toit d'un véhicule. La taille des images est 1920×1080 pixels. Deux bases d'images avec vérité-terrain sont établies :

- un lot de 640 images, avec 4 catégories de 160 images indépendantes les unes des autres, permet d'effectuer l'apprentissage et l'optimisation des paramètres.
- une séquence de 11689 images avec des tronçons homogènes de *VMA* est utilisée pour évaluer l'algorithme. La catégorie des images étant, dans de nombreux cas, difficile à déterminer, trois vérités-terrain (*VT*) sont initialement établies par trois opérateurs. Pour l'évaluation, une unique vérité-terrain est ensuite générée en combinant les résultats des trois VT selon le vote majoritaire ou selon la catégorie intermédiaire lorsque les 3 votes diffèrent.

5 Résultats et discussion

L'apprentissage de l'algorithme et l'optimisation des différents paramètres propres à chaque approche sont effectués sur le lot de 640 images individuelles selon une procédure de validation croisée. Les résultats montrent que, pour la méthode bas-niveau, l'ajout de la couleur (mCentrist au lieu de Centrist) et de la pyramide spatiale a un impact fort sur le taux de vrais positifs (*TVP*) quelle que soit la catégorie de vitesse. Le gain observé est en moyenne de 11,4% pour l'ensemble des catégories lorsqu'on utilise les descripteurs mCentrist calculés sur la pyramide spatiale par rapport aux attributs Centrist extraits de l'image globale. Dans l'approche haut-niveau, les paramètres ont été optimisés de manière empirique sur cette même base de 640 images. Les valeurs optimales sont $n_{mots} = 160$ mots, $n_c = 20$ pixels et $n_{bins} = 18$.

Sur ce premier lot, Les résultats finaux pour les deux approches sont regroupés dans le tableau 1. Quel que soit l'algorithme utilisé, le meilleur *TVP* est obtenu pour la catégorie 110 avec 96,2% (resp. 96,9%) pour l'approche bas-niveau (resp. haut-niveau). La catégorie 70 donne les moins bons résultats avec respectivement $TVP = 79,4\%$ et $TVP = 81,2\%$ pour les approches bas-niveau et haut-niveau. On observe que les résultats sont significativement meilleurs pour la catégorie 90 avec l'approche haut-niveau alors que la catégorie 50 est mieux identifiée avec la méthode bas-niveau. Enfin, on remarque des confusions importantes entre les catégories 50, 70 et 90.

Dans un deuxième temps, les deux algorithmes proposés sont évalués sur la séquence de 11689 images avec et sans lissage. Les résultats de la segmentation sont montrés sur la figure 3. Une première observation visuelle montre que, sans filtrage, les sections sont très grossièrement identifiées. Il existe ainsi de nombreux changements de catégories entre deux images successives, contrairement à ce que l'on peut voir sur la vérité-terrain. Il apparaît donc indispensable d'effectuer un filtrage des résultats en prenant en compte les résultats des images avoisinantes. En appliquant le filtre moyenne puis le filtre morphologique, on note sur la figure 3 que les sections sont plus homogènes. Par ailleurs, la méthode « haut-niveau » semble plus fidèle à la vérité-terrain que l'approche basée sur les descripteurs mCentrist. Il y a moins de changements de catégorie d'une image à l'autre. Pour cette séquence, les résultats quantitatifs sont donnés dans le tableau 2. De manière générale, les *TVP* mesurés sur cette séquence sont beaucoup plus faibles que ceux observés sur le premier lot de 640 images. L'algorithme semble toujours très performant pour la catégorie 110 (avec $TVP = 100\%$ pour l'algorithme bas-niveau et $93,5\%$ pour la méthode *BOW*). En revanche, pour les autres catégories, les *TVP* sont sensiblement moins élevés, en particulier pour la catégorie 70 (approche haut-niveau). Visuellement, la catégorie 110 ($2\times$ 2 voies) est celle qui se distingue le plus des autres, ce qui peut expliquer un meilleur TVP. A l'opposé, la catégorie 70 est la plus difficile à interpréter et de nombreuses confusions existent entre cette catégorie et les catégories voisines. On remarque également que la classe 90, nettement majoritaire dans la séquence d'images, est beaucoup mieux identifiée par l'approche haut-niveau, ce qui confirme l'impression visuelle

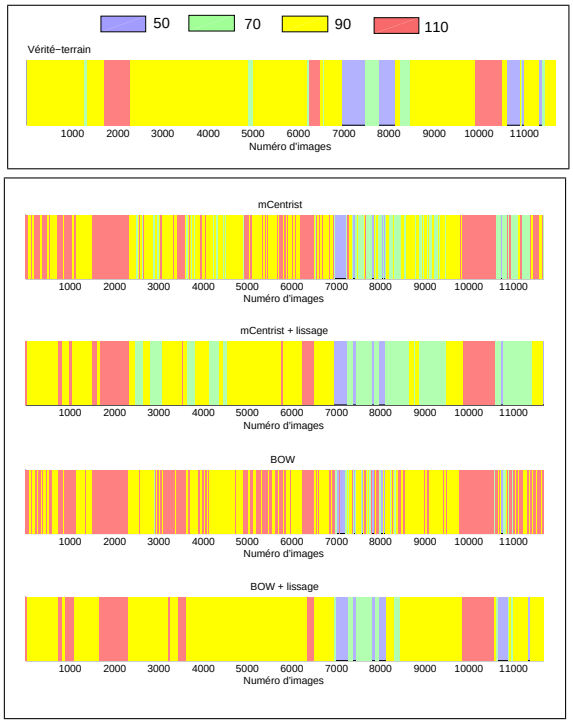

FIGURE 3 – Résultats de segmentation sur un itinéraire de 11689 images réelles. L'axe des abscisses représente le numéro d'image dans la séquence. Les couleurs correspondent aux 4 catégories. La première ligne est la vérité-terrain. Les résultats de segmentation avant et après lissage sont donnés lignes 2 et 3 pour la méthode bas-niveau et lignes 4 et 5 pour la méthode haut-niveau.

donnée par la figure 3. On observe de nombreuses erreurs au niveau des zones de transition entre sections, par exemple au niveau des images numérotées 2000 ou 8000 où la largeur des sections obtenues par l'algorithme diffère par rapport à la largeur des mêmes sections de la VT (voir Fig 3). Les erreurs se traduisent aussi par l'apparition de tronçons erronés. L'approche bas-niveau a ainsi tendance à classer, à tort, plusieurs sections dans la catégorie 70 (au lieu de 50 ou 90). Dans l'approche haut-niveau, des sections de la catégorie 90 dans la *VT* sont classés en 110 par l'algorithme.
En termes de temps de calcul, l'extraction des descripteurs est la phase la plus longue du processus. Dans l'approche bas-niveau, la dimension du vecteur attribut est très élevé (94448 valeurs) et son temps d'extraction est de 4,2012 secondes par image. Avec la méthode haut-niveau, le vecteur attribut correspond à la taille du vocabulaire $n_{mots} = 160$ mais le processus d'extraction nécessite 10,12 secondes. Notons que tous les algorithmes ont été développés et optimisés avec le logiciel Matlab®. Les objectifs de l'application n'étant pas temps réel, les temps de calcul observés ne sont donc pas rédhibitoires.

TABLE 1 – Matrice de confusion (en %) pour l'approche bas-niveau (haut) et haut-niveau (bas) sur le lot de 640 images.

		Algorithme			
		50	70	90	110
VT	50	95	5	0	0
	70	10	79.4	10	0
	90	1.9	9.4	83.8	5
	110	0	1.9	1.9	96.2

		Algorithme			
		50	70	90	110
VT	50	90.6	8.1	0.6	0.6
	70	7.5	81.2	10	1.2
	90	1.2	5.6	88.8	4.4
	110	0.6	0.6	1.9	96.9

6 Conclusion et perspectives

Nous nous sommes intéressés à la problématique de la segmentation d'itinéraires routiers en catégories de limites de vitesse. Un algorithme en deux étapes a été proposé avec, en premier lieu, une classification des images à partir de descripteurs bas-niveau ou haut-niveau, puis une seconde étape de lissage sur l'ensemble de la séquence. L'analyse des résultats montre que l'algorithme n'est pas généralisable puisque les *TVP* sont significativement plus faibles sur la base de validation. En revanche, cette étude a montré que la prise en compte de l'information séquentielle améliore les performances de classification. Elle apparaît donc indispensable dans les travaux futurs.
Parmi les perspectives, l'amélioration des performances peut être envisagée à la fois par une meilleure catégorisation des images avec par exemple l'introduction de l'information couleur dans l'approche haut-niveau et à la fois par l'utilisation de méthodes de lissage plus robustes telles que les modèles markoviens . Enfin, la présence de données déséquilibrées dans la séquence test (majorité d'images dans la classe 90) devra être prise en compte.

TABLE 2 – Matrice de confusion (en %) pour la segmentation d'itinéraires sur la séquence de 11689 images : approche mCentrist (haut) ; approche BOW (bas). $N_{filtre} = 21$.

		Algorithme			
		50	70	90	110
VT	50	55.7	44.3	0	0
	70	5.1	67.6	27.1	0.1
	90	0.4	27.6	62.9	9
	110	0	0	0	100

		Algorithme			
		50	70	90	110
VT	50	64.4	25.3	6.3	4
	70	4.3	44.1	51	0.1
	90	0.3	4	84.5	11.1
	110	0	0	6.5	93.5

Références

[1] L. BREIMAN : Random forests. *Machine learning*, 45(1): 5–32, 2001.

[2] N. DALAL et B. TRIGGS : Histograms of oriented gradients for human detection. *In Proceedings of International Conference on Computer Vision and Pattern Recognition*, pages 886–893, San Diego, Etats-Unis, juin 2005.

[3] P. FOUCHER, E. MOEBEL et P. CHARBONNIER : Route segmentation into speed limit categories by using image analysis. *In Proceedings of International Conference on Computer Vision Theory and Applications*, volume 2, pages 416–423, Berlin, Allemagne, mars 2015.

[4] G. IVAN et C. KOREN : Recognition of built-up and non-built-up areas from road scenes. *In Proceedings of Transport Research Arena Conference*, Paris, France, 2014.

[5] T. OJALA, M. PIETIK AINEN et D. HARWOOD : A comparative study of texture measures with classification based on feature distributions. *Pattern Recognition*, 29(1): 51–59, 1996.

[6] I. SIKIRIC, K. BRKIC, J. KRAPAC et S. SEGVIC : Image representations on a budget : Traffic scene classification in a restricted bandwidth scenario. *In Proceedings of IEEE Intelligent Vehicles Symposium*, pages 845–852, Dearborn, Etats-unis, mars 2014.

[7] I. TANG et T.P. BRECKON : Automatic road environment classification. *IEEE transactions on Intelligent Transportation systems*, 12(2):476–484, décembre 2010.

[8] J. WU et J.M. REHG : Centrist : A visual descriptor for scene categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1489–1501, 2011.

[9] Y. XIAO, J. WU et Y. YUAN : mCentrist : A multi-channel feature generation mechanism for scene categorization. *IEEE Transactions on Image Processing*, 23(2):823–836, 2014.

[10] R. ZABIH et J. WOODFILL : Non-parametric local transforms for computing visual correspondence. *In Proceedings of European Conference on Computer Vision*, pages 151–158, Stockholm, Suède, 1994. Springer.