

# Mise en correspondance couleur par programmation dynamique pour les véhicules intelligents

Iyadh CABANI, Gwenaëlle TOULMINET, Abdelaziz BENSRAHAI

Laboratoire d'Informatique, de Traitement de l'Information et des Systèmes - EA 4108  
Avenue de l'Université, 76801 Saint-Etienne-du-Rouvray Cedex, France  
iyadh.cabani@insa-rouen.fr, gwenaëlle.toulminet@insa-rouen.fr,  
abdelaziz.bensrahair@insa-rouen.fr

**Résumé** – Dans cette communication, nous proposons un opérateur de détection de contours couleur rapide et auto-adaptatif ainsi qu'un algorithme de mise en correspondance couleur par programmation dynamique pour les véhicules intelligents. Le but étant d'intégrer ce processus dans notre système de stéréo vision couleur en cours de conception afin d'extraire les contours 3D des obstacles. Dans une première étape, les points de contours sont extraits de façon auto-adaptative en utilisant l'opérateur déclivité couleur. Dans une deuxième étape, ces points de contours sont mis en correspondance par une méthode de programmation dynamique, en tenant compte des contraintes géométriques, colorimétriques, d'ordre et d'unicité. Les résultats de mise en correspondance sont des points de contours 3D.

**Abstract** – In this article, we present a fast and self-adaptive color stereo vision matching algorithm which is a first step in the conception of a color stereo vision system for road obstacle detection. Our color matching approach associates vertical edges points using dynamic programming. Edges points are extracted using the operator color-declivity. Performance of segmentation and matching are discussed. Experimental results are shown.

## 1 Introduction

Dans le cadre de l'aide à la conduite automobile, deux techniques existent pour la perception de l'environnement d'un mobile, suivant qu'elles font appel à des capteurs actifs ou passifs. Des radars [1], des lidars [2], des systèmes de vision monoculaire [3] et stéréo [4][5][6] sont utilisés pour la détection des obstacles. Parfois l'information provenant de capteurs actifs et passifs sont fusionnées pour la détection des obstacles [7][8].

Souvent les systèmes de vision proposés sont basés sur des caméras en niveaux de gris. Malheureusement, quand les conditions de visibilité sont réduites (nuits, brouillards, etc.), ces systèmes sont presque aveugles. Comme conséquence, la détection des obstacles est moins robuste et moins fiable. Face à ce problème de visibilité réduite, des caméras couleur ou infrarouge peuvent être utilisées. Par exemple, à partir d'images couleur, *Cheng* propose d'extraire les primitives appartenant à la route [9], *Maldonado-Bascon* propose d'extraire les panneaux de signalisation [10], nous nous avons extrait les feux arrière des véhicules [11], alors que d'autres utilisent des caméras infrarouge pour détecter les piétons [12].

En partant de notre expérience sur les systèmes de stéréo vision en niveaux de gris [13] et afin d'améliorer la robustesse et la fiabilité de la détection des obstacles, un système de stéréo vision couleur est en cours de conception. L'approche couleur se compose de trois étapes. La première étape consiste à extraire les contours verticaux des objets. L'opérateur de détection de contours doit être robuste et fiable quelques soient les conditions de visibilité. Pour cela, nous avons défini un opérateur de détection de

contours auto-adaptatif pour les images couleurs appelé déclivité couleur (cf. section 2). La deuxième étape est la mise en correspondance des contours verticaux extraits sur la paire stéréoscopique par la programmation dynamique. La mise en correspondance doit être suffisamment rapide pour pouvoir envisager son utilisation en temps réel pour la détection des obstacles (cf. section 3). Cette carte 3D éparse alimente un module d'extraction des contours 3D des obstacles [13].

## 2 Segmentation couleur

Notre équipe a développé un opérateur d'extraction de points de contour monodimensionnel, rapide et auto-adaptatif, appelé déclivité [14]. Cet opérateur est robuste et fiable particulièrement pour les scènes extérieures mais malheureusement il n'est applicable qu'aux images en niveaux de gris. Dans cette communication, nous proposons une extension de cet opérateur aux images couleurs appelé déclivité couleur.

Une image couleur se compose de trois bandes. La première étape pour la détection du contour de l'image couleur est l'extraction du contour de chacune des trois bandes en utilisant l'opérateur déclivité défini dans [14].

Chaque déclivité est caractérisée par ses attributs (cf. figure 1) :

- la coordonnée de son premier pixel, notée  $x_i$ , sur une ligne image,
- la coordonnée de son dernier pixel, notée  $x_{i+1}$ , sur une ligne image,
- sa largeur, définie par :  $x_{i+1} - x_i$ ,

- son amplitude, définie par :  $d_i = I(x_{i+1}) - I(x_i)$ ,
- sa position sur une ligne image est obtenue avec une erreur maximale de 1 pixel en calculant la position moyenne des points appartenant à la déclivité pondérée par le carré des gradients, noté  $X_i$ , et défini par :

$$X_i = \frac{\sum_{x=x_i}^{x_{i+1}-1} [I(x+1) - I(x)]^2 (x+0.5)}{\sum_{x=x_i}^{x_{i+1}-1} [I(x+1) - I(x)]^2} \quad (1)$$

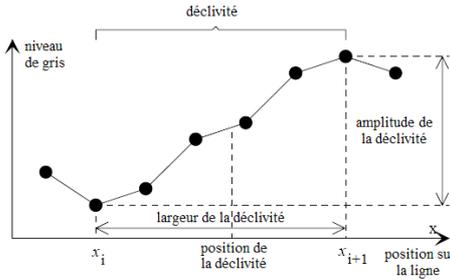


FIG. 1 – Paramètres caractéristiques d'une déclivité.

Cette forme quadratique est bien adaptée aux contours étendus irréguliers, c'est-à-dire étendus sur plusieurs pixels avec des pentes variables provoquées par l'effet du bruit non filtré : le carré des gradients est utilisé de manière à privilégier les pentes de déclivité les plus importantes pour une meilleure localisation.

L'opérateur d'extraction de déclivités caractéristiques a pour spécificité d'être auto-adaptatif. Le seuil de décision  $d_{t_c}$  ( $c \in \{1, 2, 3\}$ ) est fonction de la ligne observée et il est définie par :

$$d_{t_c} = \alpha \times \sigma_c \quad (2)$$

avec  $\alpha = 7, 6$  et  $\sigma_c$  l'écart type de l'histogramme des variations des pixels sur la bande  $c$  d'un profil ligne image, en supposant que ces variations sont un bruit blanc gaussien à moyenne nulle et sont caractérisées par leurs écarts-types  $\sigma_c$ . La valeur du coefficient  $\alpha$  a été déterminée de manière à éliminer 99.98% des accroissements dus au bruit ou non significatifs.

En respectant l'équation 2, trois seuils de décision sont calculés pour chaque ligne  $l$  de l'image couleur, et notés  $d_{t_1}$ ,  $d_{t_2}$  et  $d_{t_3}$ . Ces seuils de décision sont respectivement associés à la ligne  $l$  de la couche 1, couche 2 et couche 3.

Dans une première étape, la déclivité de la couche  $c$  ( $c \in \{1, 2, 3\}$ ) est extrait dans le cas où son amplitude  $d_{j_c}$  vérifie l'inégalité suivante  $d_{j_c}^2 \geq d_{t_c}^2$ .

En deuxième étape, les déclivités couleurs sont construites par fusion des déclivités déjà obtenues sur chacune des trois couches en utilisant l'opérateur arithmétique/logique OR.

Ainsi, chaque déclivité couleur est caractérisée par ses attributs :

- l'ensemble des couches sur lesquelles la déclivité couleur a été obtenue, noté  $\Omega_i$ . Pour une déclivité couleur obtenue à partir de la couche 1 et la couche 3,  $\Omega = \{1, 3\}$ ,

- la coordonnée de son premier pixel, notée  $u_i$ , sur une ligne image, et défini par :

$$u_i = \max_{c \in \Omega_i} \{x_{j_c}\}$$

- la coordonnée de son dernier pixel, notée  $u_{i+1}$ , sur une ligne image, et défini par :

$$u_{i+1} = \min_{c \in \Omega_i} \{x_{j+1_c}\}$$

- sa largeur, définie par :  $u_{i+1} - u_i$ ,
- sa position.

La définition de la déclivité couleur présente quelques avantages. Elle est applicable sur des images en niveaux de gris, sur des images couleurs codées sur différents espaces couleurs ou même sur un espace hybride, et est facilement extensible aux images multi-spectrales.

### 3 Mise en correspondance couleur par programmation dynamique

On peut résumer le problème de mise en correspondance comme étant la recherche d'un chemin optimal sur un graphe (2D), où les axes verticaux et horizontaux représentent respectivement les déclivités couleurs gauches d'une ligne image et les déclivités couleurs droites de la ligne image stéréo correspondante. Les intersections de ces axes sont des nœuds, qui représentent des associations hypothétiques de déclivités couleurs. La mise en correspondance globale des déclivités couleurs droites avec les déclivités couleurs gauches est obtenue en sélectionnant un chemin dont la valeur du gain global qui lui est associée est maximale. Dans les approches classiques de mise en correspondance par programmation dynamique, un coût est associé à chaque appariement pour optimiser le processus de mise en correspondance. L'inconvénient est qu'il peut augmenter indéfiniment ce qui affecte le temps de calcul de l'algorithme. Pour cette raison, la valeur du gain global est calculée en utilisant une fonction de gain locale non linéaire qui prend ses valeurs entre 0 et une valeur maximale égale à  $3 \times \max_{c \in \Omega_{i,j}} \{g_{max_c}\}$ .  $\Omega_{i,j}$  est égale à  $\Omega_i \cup \Omega_j$  et  $g_{max_c}$  est définie par :

$$g_{max_c} = 3 \times (d_{t_{Rc}} + d_{t_{Lc}}), \quad c \in \{1, 2, 3\} \quad (3)$$

Pour  $c \in \{1, 2, 3\}$ ,  $d_{t_{Rc}}$  et  $d_{t_{Lc}}$  sont respectivement les valeurs des seuils auto-adaptatifs utilisées lors de la détection des déclivités couleurs caractéristiques des lignes droites et gauches sur chacune des trois couches couleurs. En outre, la fonction de gain local est obtenue de la façon suivante :

Cas 1.

Si  $\forall c \in \{1, 2, 3\}$  ( $l_{phdist_c} < g_{max_c}$  et  $r_{phdist_c} < g_{max_c}$ ) alors

$$gain = \frac{1}{Card(\Omega_{i,j})} \sum_{c \in \Omega_{i,j}} (3 \times g_{max_c} - l_{phdist_c} - r_{phdist_c}) \quad (4)$$

Cas 2.

Si  $\forall c \in \{1, 2, 3\}$  ( $l_{phdist_c} < g_{max_c}$  et  $r_{phdist_c} \geq g_{max_c}$ )

alors

$$gain = \frac{1}{Card(\Omega_{i,j})} \sum_{c \in \Omega_{i,j}} (g_{max_c} - l_{phdist_c}) \quad (5)$$

Cas 3.

Si  $\forall c \in \{1, 2, 3\}$  ( $l_{phdist_c} \geq g_{max_c}$  et  $r_{phdist_c} < g_{max_c}$ ) alors

$$gain = \frac{1}{Card(\Omega_{i,j})} \sum_{c \in \Omega_{i,j}} (g_{max_c} - r_{phdist_c}) \quad (6)$$

avec

$$l_{phdist_c} = \sum_{k=0}^{k=2} |I_{R_c}(u_i - k) - I_{L_c}(u_j - k)|, \quad c \in \{1, 2, 3\}$$

et

$$r_{phdist_c} = \sum_{k=0}^{k=2} |I_{R_c}(u_{i+1} + k) - I_{L_c}(u_{j+1} + k)|, \quad c \in \{1, 2, 3\}$$

Le gain est calculé :

- s'il y a ressemblance colorimétrique globale (cas1), gauche (cas2) ou droite (cas3),
- si la déclivité couleur gauche et sa correspondante droite respectent la même monotonie sur chacune des couches de  $\Omega_{i,j}$ .

Cette fonction de gain local non linéaire a pour avantage de rendre l'algorithme de mise en correspondance auto-adaptatif, robuste et rapide. En effet, elle prend en compte la nature des deux lignes épipolaires car elle utilise la valeur de  $g_{max_c}$  qui est calculée à partir de l'écart type du bruit estimé sur chacune des trois couches de chaque ligne image. Elle prend ses valeurs entre 0 et  $3 \times \max_{\forall c \in \Omega_{i,j}} \{g_{max_c}\}$  ce qui permet la convergence du gain global dans tous les cas. Enfin, elle limite les distances colorimétriques à  $g_{max_c}$  pour la couche  $c$  ce qui réduit le nombre de candidats à l'appariement et augmente la rapidité de l'algorithme.

## 4 Résultats expérimentaux

Dans TAB. 1, nous prouvons que la mise en correspondance couleur est plus robuste qu'en niveaux de gris. En utilisant la couleur, le taux de fausse mise en correspondance est moins important, de plus, le nombre de primitive extrait est beaucoup plus important. L'apport de la couleur a été quantifié à partir de la base d'images MARS/PRESCAN [15]. En moyenne 5200 déclivités sont extraites avec le processus niveaux de gris et 5700 déclivités couleur avec le processus couleur, ce qui représente un apport de 10 % au niveau points de contours. Ces points de contours sont mis en correspondance par programmation dynamique et l'apport de la couleur est de 33 % au niveau des points de contours mis en correspondance. Notre processus couleur améliore l'extraction des contours et la mise en correspondance par ce qu'il résout en partie le problème du métamérisme (cf. FIG. 2) et des couleurs qui ont une intensité très proche. La localisation des contours est améliorée dans le cas particulier où les couleurs adjacents ont une intensité monotone en niveaux de gris (cf. FIG. 3). Tout cela fait que la carte 3D éparsée est plus

robuste en utilisant le processus couleur (cf. FIG. 4). Sur la base d'images MARS/PRESCAN, on montre qu'on diminue le taux des mauvaises mise en correspondance de 40 %. Sur le tableau TAB. 2, le temps de calcul moyen pour l'extraction des contours de la paire stéréo par l'opérateur déclivité couleur et de la mise en correspondance sont présentés pour la base d'images Middlebury [16]. Il est à noter que notre processus est totalement parallélisable, le traitement de l'extraction des contours et de la mise en correspondance couleur par programmation dynamique se fait ligne par ligne. L'utilisation d'une architecture dédiée [17] permet cette parallélisation ce qui revient à diviser le temps de calcul par le nombre de lignes.



FIG. 2 – Métamérisme : (a) Image composée de deux rectangles couleur. (b) Image contour obtenue à partir de la déclivité couleur. (c) Le correspondant en niveaux de gris de (a); le métamérisme est observé. (d) Image contour obtenue à partir de la déclivité; les contours ne sont pas détectés.



FIG. 3 – Couleurs adjacents ayant une intensité monotone en niveaux de gris. (a) Image couleur. (b) Image contour obtenue à partir de la déclivité couleur. (c) Le correspondant en niveaux de gris de (a). (d) Image contour obtenue à partir de la déclivité.

TAB. 1 – Performance de la mise en correspondance couleur par rapport à une mise en correspondance niveaux de gris.

Nom de l'image (type de la mise en correspondance)	Déclivités mis en correspondance	Fausse mise en correspondance
Teddy (couleur)	12470	939 (7,53 %)
Teddy (niv. de gris)	9335	876 (9,38 %)

## 5 Conclusion

Dans cette communication, nous avons présenté une méthode d'extraction des contours d'images et un algorithme de mise en correspondance basé sur la programmation dynamique qui sont auto-adaptatives, rapides, robustes et fiables. Cette carte 3D éparsée alimente un module d'extraction des contours 3D des obstacles.

TAB. 2 – Temps de calculs moyen pour l'extraction des contours et la mise en correspondance sur la base Middlebury.

Détection des contours pour la paire stéréoscopique	32 ms
Mise en correspondance par programmation dynamique	54 ms

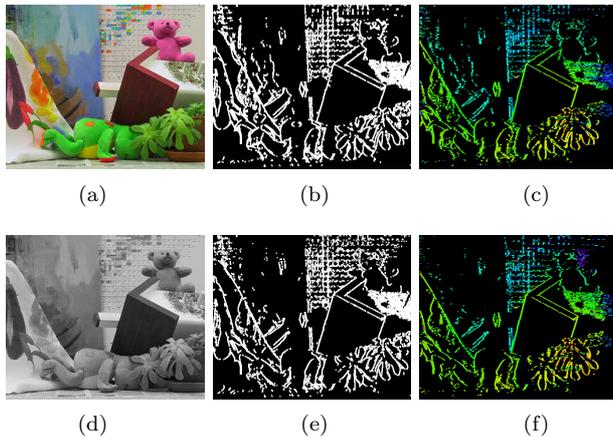


FIG. 4 – Résultats expérimentaux d'extraction des contours et de la construction de la carte de disparité (la disparité est codée en fausse couleur). (a) Image couleur (Teddy [16]). (b) Image contour obtenue à partir de la déclivité couleur. (c) Résultats de la mise en correspondance couleur. (d) Le correspondant en niveaux de gris de (a). (e) Image contour obtenue à partir de la déclivité. (f) Résultat de la mise en correspondance niveaux de gris.

## Références

- [1] F. Kruse, F. Follster, and M. Ahrholdt, "Target classification based on near-distance radar sensor," in *Proc. of the IEEE Intelligent Vehicles Symp., Parma, Italy*, June 2004.
- [2] B. Gao and B. Coifman, "Vehicle identification and gps error detection from a lidar equipped probe vehicle," in *Proceedings of the IEEE International Conference on Intelligent Transportation Systems*, 2006, pp. 1537–1542.
- [3] P. Lombardi and B. Zavidovique, "A context-dependent vision system for pedestrian detection," in *Proceedings of the IEEE Intelligent Vehicles Symposium, Parma, Italy*, June 2004.
- [4] U. Franke, "Real-time stereo vision for urban traffic scene understanding," in *Proc. of the IEEE Intelligent Vehicles Symp., Dearborn, USA*, Oct. 2000.
- [5] A. Benschair, M. Bertozzi, A. Broggi, A. Fascioli, S. Mousset, and G. Toulminet, "Stereo vision-based feature extraction for vehicle detection," in *Proceedings of the IEEE Intelligent Vehicles Symposium, Versailles, France*, June 2002.
- [6] J. I. Woodfill, R. Buck, D. Jurasek, G. Gordon, and T. Brown, "3d vision : Developing an embedded stereo-vision system," *Computer*, vol. 40, no. 5, pp. 106–108, May 2007.
- [7] G. Alessandretti, A. Broggi, and P. Cerri, "Vehicle and guard rail detection using radar and vision data fusion," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 1, pp. 95–105, March 2007.
- [8] H. Cheng, N. Zheng, X. Zhang, and H. van de Wetering, "Interactive road situation analysis for driver assistance and safety warning systems : Framework and algorithms," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 1, pp. 157–167, 2007.
- [9] H.-Y. Cheng, B.-S. Jeng, P.-T. Tseng, and K.-C. Fan, "Lane detection with moving vehicles in the traffic scenes," *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 4, pp. 571–582, December 2006.
- [10] S. Maldonado-Bascon, S. Lafuente-Arroyo, P. Gil-Jimenez, H. Gomez-Moreno, and F. Lopez-Ferreras, "Road-sign detection and recognition based on support vector machines," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 2, pp. 264–278, June 2007.
- [11] I. Cabani, G. Toulminet, and A. Benschair, "Color-based detection of vehicle lights," in *Proceedings of IEEE intelligent Vehicle Symposium, Las Vegas, USA*, June 2005.
- [12] M. Bertozzi, A. Broggi, C. Caraffi, M. Del Rose, M. Felisa, and G. Vezzoni, "Pedestrian detection by means of far-infrared stereo vision," *Computer Vision and Image Understanding*, vol. 106, no. 2-3, pp. 194–204, May-June 2007.
- [13] G. Toulminet, M. Bertozzi, S. Mousset, A. Benschair, and A. Broggi, "Vehicle detection by means of stereo vision-based obstacles features extraction and monocular pattern analysis," *IEEE Transactions on Image Processing*, vol. 15, no. 8, pp. 2364–2375, August 2006.
- [14] P. Miché and R. Debrie, "Fast and self-adaptive image segmentation using extended declivity," *Annals of telecommunication*, vol. 50, no. 3-4, pp. 401–410, 1995.
- [15] W. van der Mark and D. Gavrila, "Real-time dense stereo for intelligent vehicles," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 1, 2006, pp. 38–50.
- [16] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, [www.middlebury.edu/stereo/](http://www.middlebury.edu/stereo/), vol. 47, pp. 7–42, 2002.
- [17] A. Greiner, F. Petrot, M. Carrier, M. Benabdenbi, R. Chotin-Avot, and R. Labayrade, "Mapping an obstacles detection, stereo vision-based, software application on a multi-processor system-on-chip," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, 2006, pp. 370–376.