

# ***DETECTION DE VEHICULES DANS UN ENVIRONNEMENT AUTOROUTIER A L'AIDE DE DONNEES TELEMETRIQUES ET VISUELLES***

---

**Sonia Izri,**  
Doctorante en Robotique  
[Izri@iut-amiens.fr](mailto:Izri@iut-amiens.fr) + 33 3 22 53 40 88

**Eric Brassart,**  
Maître de conférence en Robotique  
[Eric.brassart@iut.u-picardie.fr](mailto:Eric.brassart@iut.u-picardie.fr) + 33 3 22 53 40 80

**Laurent Delahoche**  
Maître de conférence en Robotique  
[Laurent.delahoche@iut.u-picardie.fr](mailto:Laurent.delahoche@iut.u-picardie.fr) + 33 3 22 53 40 83

**Arnaud Clémentin**  
Docteur en Robotique  
[Arnaud.clémentin@iut.u-picardie.fr](mailto:Arnaud.clémentin@iut.u-picardie.fr) + 33 3 22 53 40 85

**Adresse professionnelle**  
IUT d'Amiens : Département informatique Université de Picardie Jules–Verne,  
Avenue des facultés, 80025 Amiens CEDEX 1

**Résumé :** Nous présentons dans cet article les principes de traitements de données mis en œuvre dans le cadre d'un projet de recherche sur l'amélioration de la sécurité automobile, intitulé **SAACAM** (**S**ystèmes **A**ctifs d'**A**ide à la **C**onduite pour **A**méliorer la sécurité automobile), qui est contractualisé dans le cadre du pôle régional **DIVA** (**D**iagnostic et **V**éhicules **A**vancés) de la région Picardie.

Ce projet est basé sur l'utilisation conjointe de deux systèmes de perception extéroceptive :

- Un télémètre laser (le SICK LMS 200)
- Un capteur de vision omnidirectionnel.

Nous montrons dans cet article les résultats obtenus dans un environnement structuré de type autoroutier, et nous indiquons plus spécifiquement les temps de calcul des différents processus mis en œuvre.

Nous terminons en présentant quelques perspectives qui envisagent de définir un système de perception bi-sensoriel avec les capteurs extéroceptifs et tenter de fusionner deux types de primitives issues de ces derniers dans le but de localiser de façon relative les véhicules avoisinant.

**Summary :** We will present in this article the principles of data processing put into operation within the framework of a research project on the improvement of automobile safety, entitled **SAACAM** (**S**ystems **A**ctifs d'**A**ide à la **C**onduite pour **A**méliorer la sécurité automobile : *Active Systems of Driving Assistance for Improvement of Motorcar Safety*) which is contracted within the framework of the **DIVA** (**D**iagnosis and **V**ehicles **A**vancés : *Diagnosis and Advanced Vehicles*) regional pole of the Picardy region.

This project is based on the simultaneous use of two systems of exteroceptive perception :

- A telemetric laser (the SICK LMS 200)
- An omnidirectional vision sensor.

In this article we will show the results obtained in a structured motorway-type environment and indicate more specifically the calculation times employed.

We will end by presenting some perspectives, which attempt to define a bi-sensory system of perception with the exteroceptive sensors and try to merge two types of primitive results of these latter with the aim of localizing the neighbouring vehicles in a relative way

**Mots clés :** Perception, Vision omnidirectionnelle, contours actifs, segmentation.

# Détection de véhicules dans un environnement autoroutier a l'aide de données télémétriques et visuelles

## 1. Introduction

Les projets concernant l'assistance à la sécurité routière par intégration de capteurs dits 'intelligents' sont extrêmement nombreux [CARSENSE, RADARNET, DENSETRAFFIC, EAST-EEA, etc.] et correspondent à une nécessité de plus en plus perceptible dans notre vie quotidienne. Cette problématique est, de surcroît, fortement encouragée par les constructeurs automobiles, les conducteurs de véhicules, la vie politique, le corps médical et les associations.

Quelque soit le niveau d'intégration des systèmes sécuritaires, quelque soit l'architecture fonctionnelle déployée, le problème récurrent reste la quantité d'information à traiter par rapport à la dynamique mise en jeu. Ainsi la synchronisation des traitements, les temps de traitement, l'acquisition temps réel sont autant de contraintes qui, rendent l'objectif final difficile à atteindre. Dans ce contexte on cherchera évidemment à privilégier des 'solutions capteurs' qui permettront soit d'obtenir un maximum d'informations en une seule acquisition, soit une information pré-traitée, qui de plus pourra être facilement embarquée sur un véhicule. Les systèmes de vision omnidirectionnelle [10], [11] sont à ce titre très intéressants à utiliser puisque lorsqu'ils sont embarqués sur un véhicule, ils permettent de détecter des dangers proches sur 360 degrés et cela en une seule acquisition. L'intégration d'un tel capteur sur un véhicule pour la détection des dangers proches est originale puisqu'elle n'a pas encore été faite à ce jour. En outre des données visuelles, nous utilisons dans le projet un autre système de perception extéroceptive qui est le télémètre laser. Ce dernier permet d'obtenir des nuages de points issus de mesures télémétriques, sur lesquels, nous cherchons à identifier les objets présents dans l'image.

Cette solution que nous présentons fait parti du projet intitulé SAACAM (Système Actifs d'Aide à la Conduite pour AMéliorer la sécurité automobile) et est contractualisée dans le cadre du pôle régional DIVA (DIagnostic et Véhicules Avancés) de la région Picardie.

Dans cet article nous détaillerons d'une part, les mécanismes des différents traitements effectués sur les données télémétriques. D'autre part, nous présenterons l'étude d'extraction d'indices visuels dans les images panoramiques, basée sur l'utilisation des modèles déformables de type 'snakes'. Nous terminerons cet article par la présentation des premiers résultats obtenus dans un environnement structuré de type autoroutier. Finalement nous donnerons quelques perspectives à la suite de nos travaux.

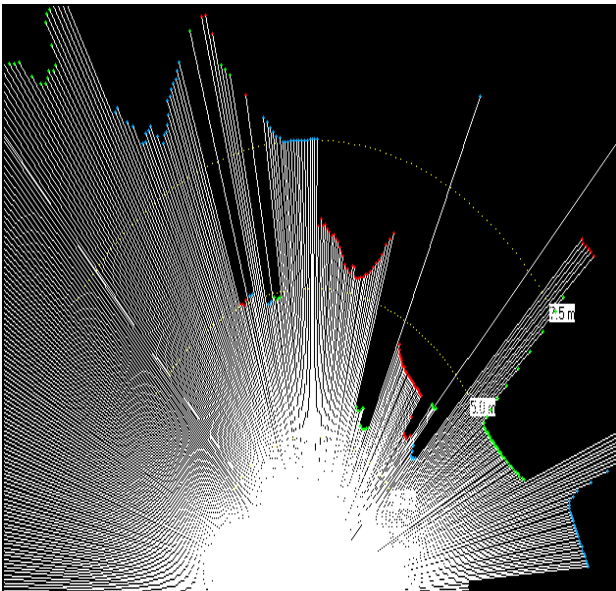
## 2. Traitement des données télémétriques

### 2.1. Présentation générale du SICK LMS 200

Le SICK LMS 200 est un scanner de mesure à balayage 2 dimensions : il scrute son environnement et récupère les coordonnées polaires de celui-ci. C'est une combinaison d'un télémètre à temps de vol avec un système de rotation du faisceau de mesure, grâce à cette technique on obtient une vision sous forme de radar.



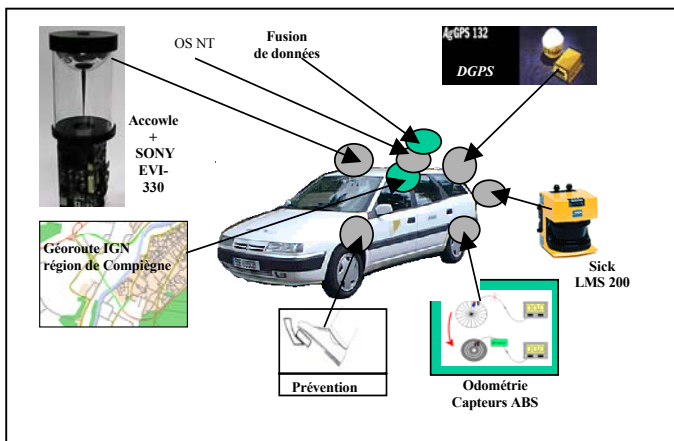
Figure 1. SICK LMS\_200



**Figure 2** Image laser acquise avec le télémètre

## 2.2. Fonctionnement du SICK LMS 200

Une impulsion (un tir) est émise par une diode laser et simultanément, une horloge est démarrée. Cette impulsion lumineuse sera renvoyée par le premier obstacle rencontré sur son chemin. L'impulsion lumineuse, renvoyée par un obstacle, arrive sur un récepteur qui déclenche l'arrêt de l'horloge. A partir de ces informations, nous déduisons la distance séparant le télémètre et l'obstacle. Et connaissant la position angulaire du laser, nous arrivons à déterminer la position de l'objet dans le plan.



**Figure 3** Schéma de fonctionnement

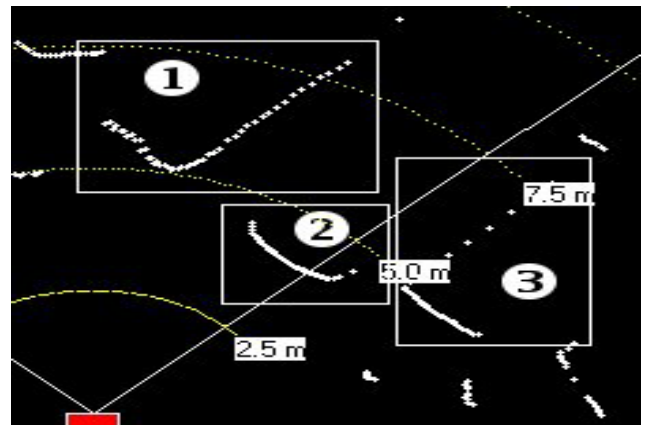
## 2.3. Intégration du SICK LMS 200 dans le projet

Le télémètre a été placé à l'arrière du véhicule pour des raisons sécuritaires, ce qui permet de récupérer une image des véhicules suiveurs, comme le montre la figure ci-dessous (**figure 3**).

## 2.4. Traitement des données du télémètre laser

### 2.4.1. Le "clustering"

Le "clustering" permet de regrouper des points consécutifs afin de répertorier l'ensemble des objets présents. Pour cela, nous avons calculé la distance qui sépare un point N de son voisin immédiat N+1. Dans l'exemple suivant, nous montrons, par les encadrements, ce que nous devrions obtenir à l'issue de cette étape.



Dans cet extrait de carte nous pouvons distinguer :

- ❶ La face avant et le côté gauche d'un véhicule.
- ❷ La face avant d'un véhicule.
- ❸ La face avant et le côté droit d'un véhicule.

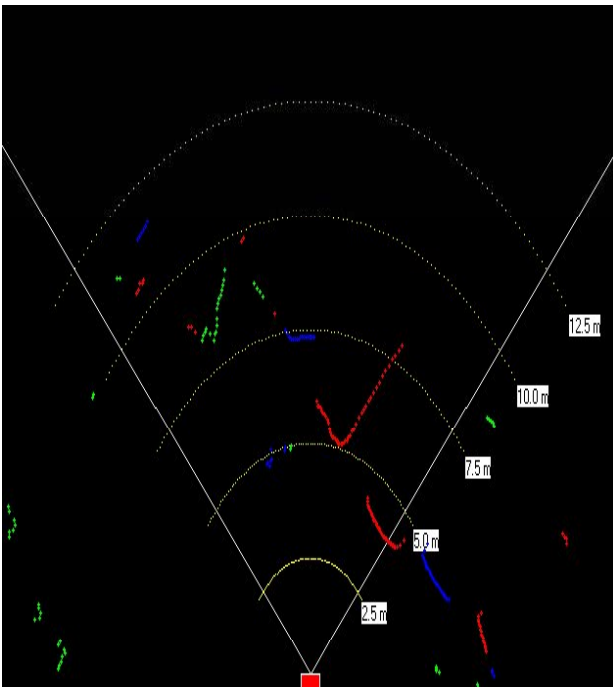
**Figure 4.** Extrait d'une image avec identification des objets caractéristiques

Les alignements de points sont caractéristiques de la détection d'arêtes appartenant à des objets. Quand la continuité n'est pas respectée, il faut donc créer une séparation de cluster.

La détermination des seuils de rupture dans l'identification des différents objets se fait actuellement de manière heuristique. Ce seuil est pour l'instant fixé à 50 cm.

Pour pouvoir mieux identifier les résultats, nous les avons affichés en 3 couleurs différentes (RVB) qui se répéteront successivement.

Le schéma ci-dessous illustre le traitement effectué sur les données télémétriques en l'occurrence la lecture des données et l'affichage des clusters coloriés.



**Figure 5.** Lecture des données et affichage des clusters coloriés

### 2.4.2. La segmentation

Après avoir "clusteriser" les données, vient l'étape de segmentation, celle-ci permet de déterminer les différentes parties d'un cluster. Ce dernier est composé d'un ou plusieurs points représentant une partie d'un objet ou sa totalité.

Les points n'étant pas tous alignés, la segmentation va permettre de tracer les segments de droites correspondant à un sous-ensemble de points.

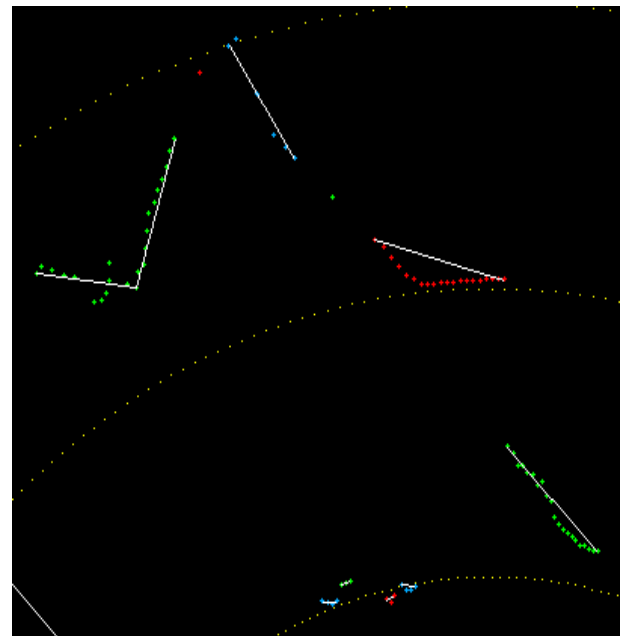
L'algorithme de recherche utilisé est celui de Duda-Hart [13]. Il consiste à regrouper

récurivement des ensembles de points alignés par rapport à un critère de distance point / droite support. L'algorithme est stoppé lorsqu'il n'y a plus de points qui vérifient la condition de distance. En appliquant cette méthode sur les différents clusters, on pourra déterminer les différentes droites qui composent chaque cluster.

### 2.4.3. Filtrage

#### 2.4.3.1. Suppression des « petits » segments

Les 'petits' clusters ne permettent pas de conclure quant à la nature de l'objet qui est représenté. Et les clusters composés d'un seul point ne sont pas utilisables non plus.



**Figure 6 .** Suppression des clusters non significatifs

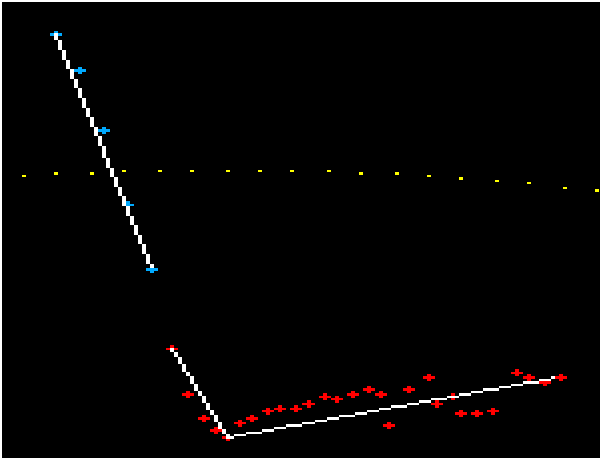
Pour supprimer ces éléments, un seuil minimum de points est fixé.

En parcourant toute la liste des clusters, nous recherchons le nombre de points qui compose chaque cluster et si ce nombre est inférieur au minimum imposé, alors il est supprimé de la liste.

### 2.4.3.2. Fusion de droites alignées

La fusion de droites alignées permet de fusionner deux droites qui sont :

- Alignées avec un angle  $\theta$  de  $\pm 10^\circ$ ,
- Une distance.

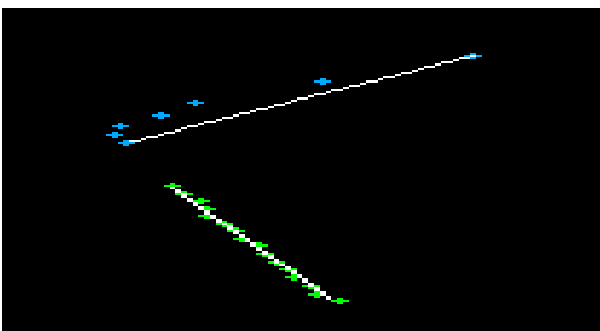


**Figure 7.** Exemple de fusion de droites alignées

### 2.4.3.3. Fusion de droites orthogonales

Pour la fusion de droites orthogonales, on va supposer que ces deux éléments n'en forment qu'un seul suivant des critères définis et modifiables par l'utilisateur.

Dans le schéma ci-contre, nous pouvons remarquer que si on prolonge les deux droites, elles se coupent en un point I. Nous pourrions ainsi dire que cet élément forme un objet et non deux objets.



**Figure 8.** Exemple de fusion de droites orthogonales

## 3. Traitement des données omnidirectionnelles

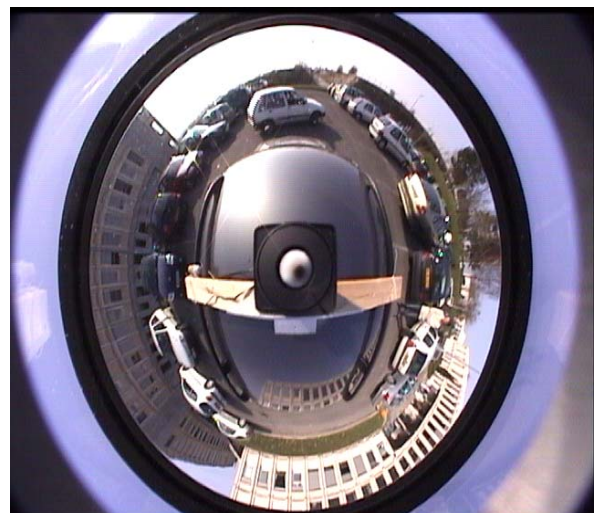
### 3.1. Les systèmes de vision omnidirectionnelle

Ces systèmes de perception associent une caméra CCD et un miroir convexe (sphérique) permettant d'obtenir une vue sur 360 degrés de l'environnement, comme le montrent les figures ci-dessous (**figure 9, 10**). Nous comprendrions aisément son intérêt dans le cas d'une utilisation sur un véhicule dans un milieu autoroutier, pour une application de type 'dangers proches'.

Voici un exemple de ce l'on peut obtenir :



**Figure9 .** Capteur sphérique Accowle



**Figure 10.** Image obtenue par le capteur sphérique

### 3.2. Modélisation par contours actifs

La première utilisation des contours actifs date de 1987, lorsque Kass, Witkin et Terzopoulos [2] présentent leurs *snakes* comme un outil permettant de résoudre plusieurs sujets majeurs de la vision par ordinateur, à savoir la détection de contours et de segments [8], la mise en correspondance en stéréovision, et surtout le suivi d'objets. La compacité et l'efficacité du concept [9] lui ont assuré un succès rapide et on trouve actuellement une multitude de publications sur la variation de sa forme originelle.

Le principe consiste à placer dans l'image un *snake* qui 'se colle' au contour de la cible. Une énergie lui est associée, fonction de sa déformation et de sa position dans l'image. Le *snake* cherche en permanence à minimiser cette énergie. Celle-ci est modélisée de telle manière que le comportement résultat est une suite de déplacements et de déformations jusqu'à ce que le *snake* se stabilise sur un minimum local de son énergie qui correspond alors à la forme de l'objet suivi. Un *snake* se comporte comme une corde élastique qui serpente vers les contours de l'image grâce à un ensemble de forces locales gérées par un processus itératif. Les forces internes conservent la forme et assurent une continuité spatio-temporelle. Les forces externes tirent et guident le *snake*. L'initialisation et la normalisation des paramètres du *snake* sont difficiles. Le *snake* doit être placé assez proche du contour pour atteindre le minimum global de la fonction d'énergie. Ce système est utilisé dans de nombreux cas où le traitement de l'image est fait en temps réel comme en imagerie médicale, mais également pour détecter des contours déjà connus comme en botanique ou en paléontologie pour l'identification des espèces.

Le choix de l'utilisation de cette méthode se justifie par diverses raisons :

- méthode extensible
- interaction de l'utilisateur
- intégration de caractéristiques très variantes ( informations géométriques et physiques )

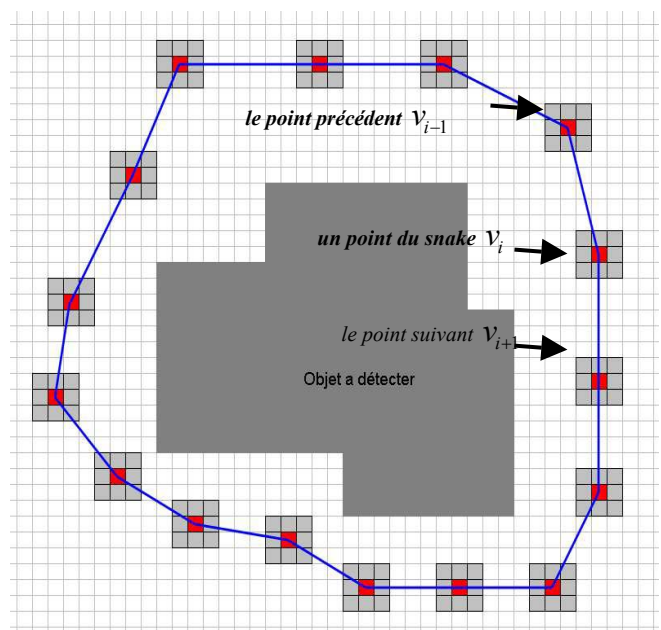
Cependant, ce modèle présente certains défauts :

- l'influence de l'initialisation du *snake*
- le choix des paramètres élastiques est dépendant de l'image de travail
- il conserve sa topologie initiale au cours de ses déformations.

#### 3.2.2 Les différentes forces du snake

La méthode de détection par *snake* ou contour actif, consiste à placer aux alentours de la forme à détecter une liste de points voisins deux à deux, représentée par une courbe.

Le *snake* peut être représenté (figure 11) de la manière suivante :



**Figure 11.** Schématisation d'un snake autour de l'objet à détecter

Comme le montre la figure 12, on ne s'intéresse pas qu'à chaque point du *snake* mais également à chaque voisinage de ces points, ce voisinage a une dimension de 3 x 3 pixels.

- Le *snake* est constitué d'un ensemble de  $n$  points,  $v_i = (x_i, y_i)$ .
- Chaque point  $v_i$ , est caractérisé par ses voisins proches : Le point précédent  $v_{i-1}$  et

le point suivant  $v_{i+1}$ . Un voisinage  $\mathcal{N}_i$  de taille  $m$  (dans notre cas  $m=8$ ), est défini pour chaque point  $v_i$ .

•

La courbe ainsi obtenue que l'on peut formuler dans le plan par une forme paramétrique :  $v(s) = (x(s), y(s))$ ,  $s \in [0, 1]$ , va se déformer de manière itérative selon l'action de plusieurs forces qui vont la tirer vers la forme.

Les forces impliquées dérivent de trois énergies associées au *snake* :

- Une énergie propre, due uniquement à la forme du contour, dite énergie interne  $E_{int}$ ,
- Une énergie potentielle imposée par l'image  $E_{image}$ . C'est elle qui va attirer le *snake* vers les contours réels présents,
- Une énergie de contraintes  $E_{cont}$  qui exprime certaines contraintes supplémentaires qui peuvent être imposées par l'utilisateur vu le *snake* qu'il veut obtenir.

L'énergie totale du *snake* est donc donnée par :

$$E_{snake} = \int_0^1 E_{snake}(v(s)) ds = \int_0^1 E_{int}(v(s)) + E_{image}(v(s)) + E_{cont}(v(s)) ds. \quad (1)$$

• **L'énergie interne :**

Dépend uniquement de la forme du *snake*. C'est une contrainte de régularité sur la courbe. Elle s'écrit :

$$E_{int} = (\alpha(s) \|v_s(s)\|^2 + \beta(s) \|v_{ss}(s)\|^2) / 2 \quad (2)$$

$\alpha$  et  $\beta$  sont les poids respectifs accordés à la dérivée première  $v_s$  et à la dérivée seconde  $v_{ss}$ .

Le premier coefficient contrôle le terme de premier ordre, qui agit sur la continuité (longueur) de la courbe, et le deuxième contrôle le terme de deuxième ordre qui agit sur sa courbure.

• **L'énergie potentielle liée à l'image :**

$$E_{image} = \gamma(s) E_{image} \quad (3)$$

Elle caractérise les éléments vers lesquels on veut attirer le *snake* sur l'image. Pour la détection de contours, on va s'intéresser aux lignes de fort gradient, on choisit, donc, le gradient de l'image au point considéré.

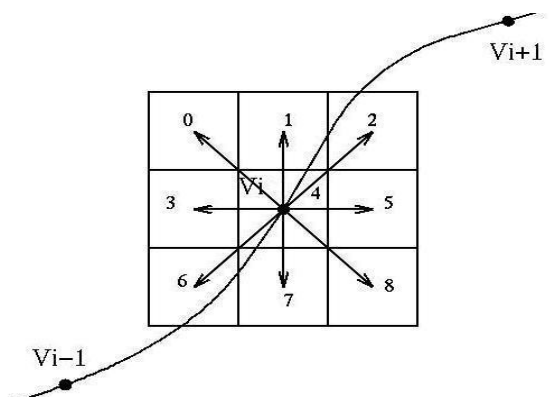
$$E_{image} = - \int_0^1 |\nabla I(v(s))|^2 ds \quad (4)$$

• **L'énergie de contraintes :**

Elle est définie par l'utilisateur selon les spécificités du problème. Par exemple imposer une distance minimale ou maximale entre deux points consécutifs du contour actif.

Une fois les énergies de chaque point du voisinage calculées, on est à même de décider vers quel pixel de l'image le point du *snake* va migrer (vers le point du voisinage dont la somme des 3 énergies est la plus faible) tel qu'on peut le voir sur la (figure 12).

Voici la manière dont sont codés les emplacements des points dans le voisinage.



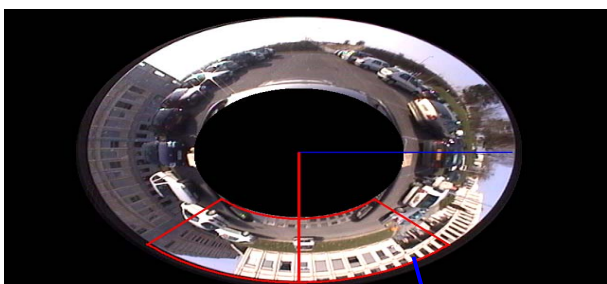
**Figure 12.** Déplacements possibles d'un point dans son voisinage

**4. Résultats obtenus**

**4.1. Transformation omnidirectionnelle en panoramique**

Les résultats obtenus initialement avec les images omnidirectionnelles ne permettent pas d'avoir des détections de contours correctes sur les objets. De ce fait, nous avons développé une méthode permettant la

conversion des images omnidirectionnelles en images panoramiques ; plus facilement interprétables d'un point de vue humain, et permettant de rejeter les zones de l'image où l'information visuelle ne subit pas le bruit du capteur. Aussi, nous avons décidé de ne travailler que sur certaines portions d'images en définissant des « imagettes » qui présentent l'avantage de ne pas trop pénaliser en temps de calcul le processus de traitement. Le procédé mis en œuvre pour réaliser cette transformation est basé sur la méthode de l'interpolation bilinéaire [10], qui permet d'obtenir des images redressées et non pixélisées.



**Figure 13.** Application d'un masque sur l'image omnidirectionnelle et tracé d'une imagette



**Figure 14.** Imagette panoramique résultante

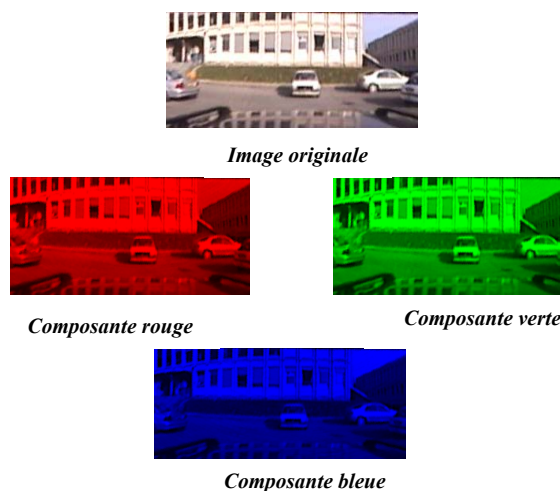
#### 4.2. Décomposition de l'imagette en trois composantes RVB

La caméra utilisée dans le projet nous a permis d'obtenir des images couleurs. Ce type d'images nous offre une information supplémentaire par rapport aux images en niveaux de gris et qui est apportée par la chrominance. On peut imaginer, par exemple que le changement de couleur entre deux

objets de luminance équivalente définira un contour que seule la donnée de luminance n'aurait pas permis de détecter. Nous avons d'abord, procédé à la décomposition de chaque imagette couleur en trois composantes RVB, ensuite calculer le gradient de chaque composante et avons choisi parmi les méthodes permettant de déterminer les contours d'une image couleur celle basée sur le calcul du Max des trois gradients sur les trois composantes couleurs. C'est ce dernier, qui va attirer le *snake* vers les contours présents.

Cela permettra de constituer une image finale moins volumineuse et plus adaptée à un traitement temps réel.

La figure ci-dessous illustre la décomposition d'une imagette panoramique en trois composantes RVB.



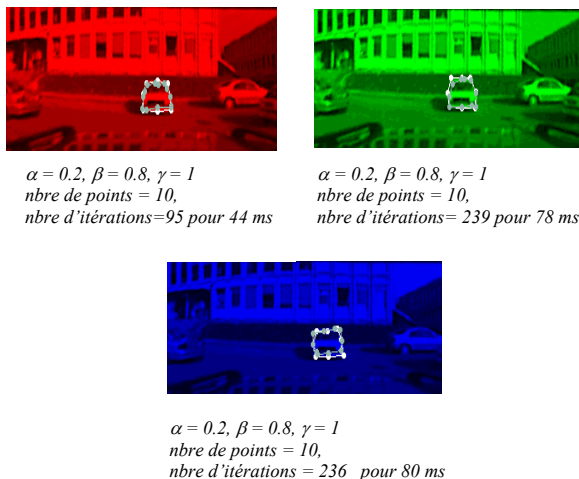
**Figure 15.** Décomposition de l'imagette couleur en composantes RVB

#### 4.3. Résultats des contours actifs

La mise en place du processus dans son ensemble se fait par un ajustement de plusieurs paramètres liés à l'utilisation des *snakes*. Pour notre application nous avons, en fonction de différents tests effectués au préalable, positionné les valeurs des paramètres,  $\alpha$ ,  $\beta$  caractérisant l'énergie interne respectivement à 0.2, 0.8, les



traitements images de l'énergie externe sont basés sur un calcul de gradient localisé de type Sobel (gradient qui fournit le meilleur résultat dans ces types d'images), le paramètre  $\gamma$  varie de 1 à 4 selon l'image et le gradient choisi. L'initialisation du *snake* s'effectue pour l'instant, à la main, et nous nous sommes limités ponctuellement au repérage d'un seul objet.



**Figure 16.** Résultats obtenus par application des snakes sur les trois composantes RVB

Nous montrons dans la figure 16, des exemples de résultats que nous avons obtenus dans un environnement d'extérieur avec un capteur omnidirectionnel sphérique. Ces derniers sont très encourageants puisque, après initialisation du *snake* [9], nous arrivons à suivre, sur une séquence d'images, un objet spécifique avec des temps de traitement variant de 40 ms à 100 ms.

## 5. Conclusion

Dans cet article, nous avons étudié et mis au point le mécanisme de détection des objets à partir de données télémétriques et visuelles. Après avoir détaillé le principe des différents traitements effectués sur les données télémétriques, ainsi que le formalisme des contours actifs, appliqué sur les images omnidirectionnelles, différentes

expérimentations ont été menées sur des environnements réels, hautement structurés de type autoroutier. Une analyse des résultats obtenus montre la compacité et l'efficacité de la méthode. Elle intègre des caractéristiques très différentes, telles que des informations géométriques, de contraste et d'intensité. Par ailleurs, le modèle travaille indépendamment de l'orientation de l'image, ce qui permet de réaliser le suivi d'objets dans l'image.

## 6. Perspectives

Nous nous sommes attachés à définir un système de perception et des méthodes d'extraction de primitives par rapport à un contexte autoroutier.

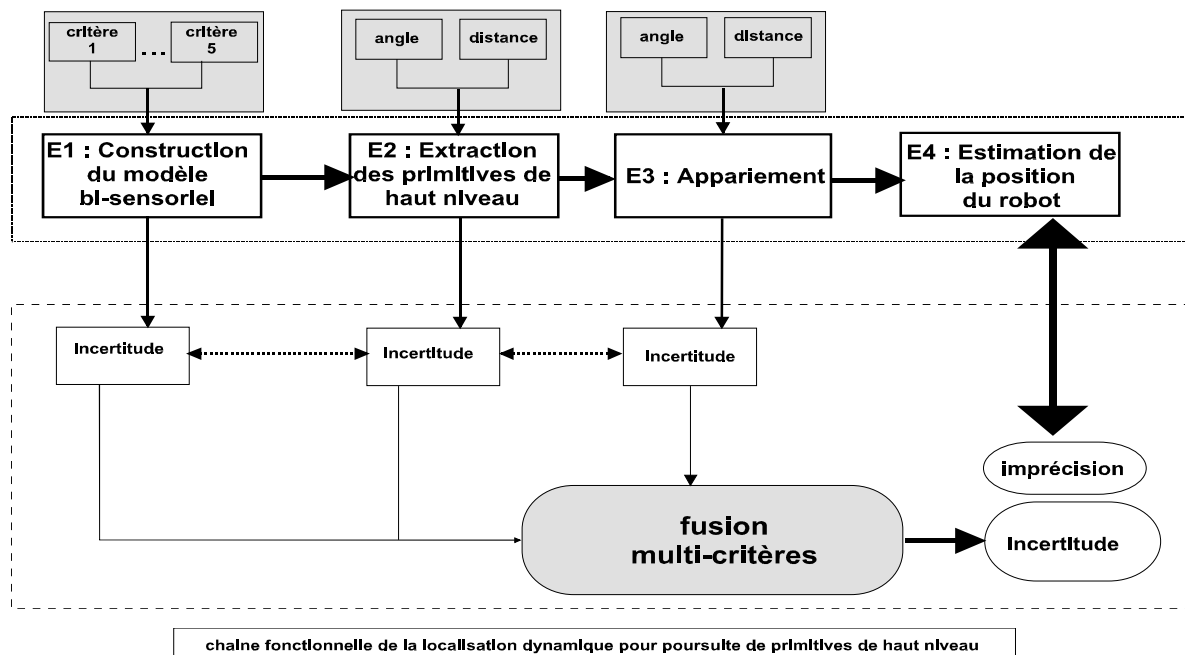
Notre stratégie est basée sur le suivi des véhicules proches, pour se faire, nous nous proposons d'utiliser une architecture de fusion de données multi-niveaux et multi-critères (**figure 17**).

Rappelons que l'objectif est de localiser de façon relative les véhicules avoisinant. Notre démarche consistera donc à propager une mise en correspondance d'une acquisition  $n$  à une acquisition  $n + 1$  tant que le ou les véhicules proches apparaissent comme des situations de danger potentiel.

Nous souhaitons nous appuyer sur la théorie de l'évidence de Dempster-Shafer [14] pour réaliser une fusion de plusieurs critères qui devront caractériser de façon robuste la croyance accordée à chaque primitive sensorielle détectée, issue du modèle bi-sensoriel. Cette théorie permet de bien manipuler des données incertaines.

En effet, contrairement à la théorie de Bayes, on peut porter de la masse sur les hypothèses singleton, mais aussi sur les unions d'hypothèses.

A chaque niveau, on manipule un cadre de discernement composé de deux hypothèses :



**Figure 17.** Schéma fonctionnel de la fusion de données multi-critères

OUI et NON. Par exemple, à l'étape E1, l'hypothèse OUI correspond à l'affirmation « OUI, le segment existe », et NON à « NON, le segment n'existe pas ». Ainsi, à l'issue de la fusion des différents critères, l'incertitude du segment sera dénotée par trois valeurs :  $m(\text{OUI})$ ,  $m(\text{NON})$ , et  $M (\text{OUI} \cup \text{NON})$ , ces trois valeurs nous permettent d'obtenir une quantification de l'incertitude sur le segment.

La redondance et la complémentarité des données sensorielles seront exploitées pour ne garder que des primitives qui correspondront aux véhicules proches.

L'étape E2 de classification de ces primitives sera de déterminer l'incertitude liée à l'extraction des primitives de 'haut niveau' (en l'occurrence des véhicules) par propagation des incertitudes associées à chaque segment.

Une fois les cibles étiquetées au sein du modèle sensoriel obtenu à un instant  $t$ , la phase de suivi multi-cibles E3 consistera à mettre en correspondance ces amers de type voiture avec ceux obtenus à l'instant  $t-1$  : nous souhaitons développer une solution basée sur la fusion multi-critères, les critères seront liés à des stratégies d'association de type 'prédiction-observation'.

La démarche permettant de réaliser le suivi multi-cibles sera donc de nouveau basée sur l'utilisation des règles de Dempster-Shafer. Nous souhaitons évidemment traiter le problème de la « non observabilité momentanée » dans le cadre du suivi et de la mise à jour des incertitudes. Ainsi, l'incertitude gérée dans les niveaux précédents sera prise en compte dans cette dernière étape. Nous avons montré dans [11][12] que l'utilité de gérer ces incertitudes est multiple. En effet, cette approche nous permet d'écarter les pistes non fiables et ainsi travailler avec des données robustes ce qui permet notamment de ne pas supprimer

brutalement une piste dès qu'un appariement n'est pas réalisé. Enfin, avec cette stratégie il nous sera possible de déduire l'incertitude globale sur la localisation par propagation des incertitudes calculées sur chacune des pistes. Cette gestion orientée sur la cohérence des incertitudes aura pour conséquence de ne déclencher des alarmes que dans des configurations à fort niveau de dangerosité.

## 7. Bibliographie

- [1] Laurent Cohen. "Modèles déformables", CEREMADE, URA CNRS 749, Université Paris-9 Dauphine.
- [2] M. Kass, A. Witkin and D. Terzopoulos. "Snakes: Active contour models". Proc. 1st Int. Conference on Computer Vision, London, 1987, pp. 259-268.
- [3] F. Leymarie and M. Levine. "Tracking deformable objects in the plane using an active contour model". IEEE Trans. on PAMI, Vol. 16 (1993), pp. 617-634.
- [4] S. de Pablo, S. Lorenzo, S. Cáceres and J.A. Domínguez. "Dynamic Image Segmentation based on DSP". Proc. of Int. Conference on Signal and Image Processing (SIP'95), Las Vegas (Nevada-USA), 1995, pp. 109-112.
- [5] Donna J. Williams and Mubarak Shah - "A Fast Algorithm For Active Contours and Curvature Estimation" – Image Understanding, Vol55, N°1, January 1992, pp14-26.
- [6] Lavanya Viswanathan – « Equations For Active Contours », Novembre 1998
- [7] J.F. Canny - "A Computational Approach to Edge Detection", IEEE, Trans. Pattern Anal. Mach. Intelligence, PAMI-8(1986), pp679-698
- [8] A.Y. Yuille, D.S. Cohen and P.W. Hallinan - "Feature Extraction From Faces Using Deformables templates", in CVPR, 1989, pp104-109
- [9] V. Caselles and B. Coll. "Snakes in Movement". Preprints of AERFAI (Córdoba-Spain) 1995, pp. 683-684.
- [10] Cyril Cauchois. "Modélisation et Calibration du Capteur Omnidirectionnel SYCLOP : Application à la localisation Absolue en Milieu Structuré". Université de Picardie Jules Verne. Décembre 2001.
- [11] Arnaud Clémentin. "Localisation d'un robot mobile par coopération multi-capteurs et suivi multi-cibles", Université de Picardie Jules Verne. Décembre 2001.
- [12] Arnaud CLERENTIN, Laurent DELAHOUCHE, Eric BRASSART, Cyril CAUCHOIS – "Mobile robot localization based on multi target tracking" - proc. of the IEEE International Conference on Robotics and Automation (ICRA 2002), Washington, USA, Mai 2002.
- [13] J. Crowley, "World modelling and position estimation for a mobile robot using ultrasonic ranging", Proc. Of IEEE Conference on Robotics and Automation, Scottsdale, May 1989, p. 674-680.
- [14] SHAFER G.A. "A mathematical theory of evidence", Princeton university press, 1976.