

## UNE APPROCHE NEURONALE POUR LA STEREOVISION LINEAIRE

Y.RUICHEK et J.-G.POSTAIRE

Université des Sciences et Technologies de Lille  
Centre d'Automatique de Lille  
59655 Villeneuve d'Ascq Cedex

### RESUME

Cet article décrit une nouvelle méthode pour la détection d'obstacles à l'avant des véhicules en utilisant la vision linéaire. Nous nous sommes intéressés à résoudre le problème de l'appariement stéréoscopique en utilisant une approche neuronale. Ce problème est ramené, dans un premier temps, à un problème d'optimisation. Une fonction d'énergie, construite pour représenter toutes les contraintes du problème, est ensuite minimisée en utilisant un réseau de neurones de Hopfield.

### 1. INTRODUCTION

Dans le cadre du programme européen **PROMETHEUS** (**PRO**gramme for an **E**uropean Traffic with **H**ighest **E**fficiency and **U**nprecedented **S**afety), le Centre d'Automatique de l'Université des Sciences et Technologies de Lille a participé au programme **ProLab2** en développant un système neuronal pour détecter les obstacles à l'avant d'un véhicule en utilisant un stéréoscope à base de caméras linéaires.

En stéréovision, l'un des problèmes les plus délicats à résoudre est l'appariement des indices visuels associés aux mêmes objets physiques présents dans la scène observée par les caméras. L'extraction de ces indices visuels et leur appariement en vue de reconstituer l'environnement 3-D peuvent rarement être effectués à des cadences suffisantes pour analyser des scènes dynamiques, sauf dans des cas très simples.

Pour palier ces limitations des systèmes stéréoscopiques classiques, nous avons remplacé les caméras matricielles par des caméras linéaires [1]. Celles-ci ont en commun un plan de visée unique de telle sorte que la taille et la position de tout objet coupé par ce plan peuvent être déterminées par triangulation. Un tel système peut servir à détecter les obstacles à l'avant des véhicules en temps réel. En effet, l'emploi de ce type de capteur permet d'obtenir une résolution bien supérieure à celle des caméras matricielles, tout en limitant la quantité d'information à traiter.

Dans cet article, nous nous intéressons au problème de l'appariement des images linéaires en utilisant une approche neuronale. Ce problème est ramené, dans un premier temps, à un problème d'optimisation. Une fonction d'énergie, construite

### ABSTRACT

This paper describes a new method for achieving obstacle detection in front of moving cars by linear computer vision. We are interested in solving the stereo correspondence problem using a neural approach. The problem is first formulated as an optimisation problem. An energy function which represents the constraints on the solutions is mapped onto a two-dimensional Hopfield neural network for minimisation.

pour représenter toutes les contraintes du problème, est ensuite minimisée en utilisant un réseau de neurones de Hopfield [2].

### 2. STEREOVISION LINEAIRE

Deux caméras linéaires sont montées à l'avant du véhicule prototype. Leurs axes optiques parallèles et séparés d'une distance  $E$  définissent un plan optique, légèrement incliné par rapport à l'horizontale, qui coupe le plan de la chaussée à l'avant de véhicule (Cf. Figure 1).

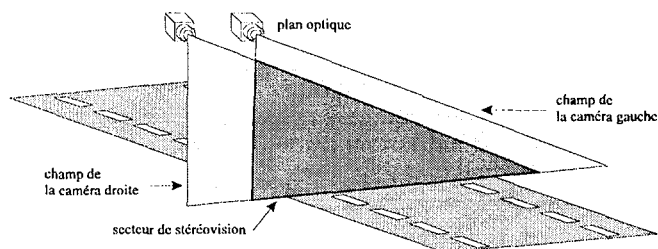


figure 1: Principe de la stéréovision linéaire.

Tout objet pénétrant dans la partie commune aux deux champs de vision dans le plan optique est vu simultanément par les deux caméras, gauche et droite, et peut ainsi être localisé par triangulation.

On définit l'axe  $OX$  par la ligne de base reliant les centres optiques des deux objectifs. L'axe  $OZ$ , perpendiculaire à  $OX$ , est parallèle aux deux axes optiques des caméras. L'origine  $O$  est située au milieu du segment joignant les centres des deux objectifs (Cf. Figure 2).



On note  $x_g$  et  $x_d$  les coordonnées image dans les images lignes, de gauche et de droite, sur lesquelles est projetée l'intersection du plan optique avec la scène.

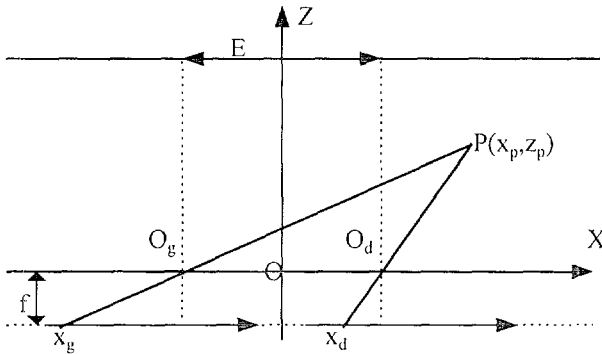


figure 2: Formation des images gauche et droite.

En utilisant le modèle du sténopé pour les deux caméras, on peut aisément calculer les coordonnées d'un point  $P(x_p, z_p)$  de la scène réelle, de coordonnées  $x_p$  et  $z_p$  dans le plan optique:

$$z_p = \frac{E \cdot f}{d} \quad (2.1)$$

$$x_p = \frac{x_g \cdot z_p}{f} - \frac{E}{d} = \frac{x_d \cdot z_p}{f} - \frac{E}{d} \quad (2.2)$$

où  $f$  est la distance focale des deux objectifs et  $d = |x_g - x_d|$  est la disparité entre les images de gauche et de droite du point  $P$ .

Ce calcul n'est possible que si les axes optiques des deux caméras sont rigoureusement parallèles. Une méthode spécifique pour calibrer les deux caméras est présentée dans [3].

### 3. EXTRACTION DES POINTS CARACTERISTIQUES

La première étape de la stéréovision est destinée à extraire, de chaque image, un ensemble de points caractéristiques susceptibles d'être les reflets des éléments physiques les plus saillants de la scène observée. Ces points doivent correspondre aux variations locales significatives du niveau de gris et doivent aussi être localisés avec précision, car ils serviront à la reconstruction 3-D des éléments de la scène. Pour satisfaire ces exigences, nous faisons appel à l'opérateur de Deriche qui utilise une famille de filtres récursifs exponentiels pour lisser le signal vidéo avant d'en détecter les fronts avec précision [4].

Le lissage des images monodimensionnelles est obtenu en convoluant les signaux issus des caméras linéaires, notés  $x(n)$ , où  $n$  indique la position du point considéré le long de l'image-ligne, avec l'opérateur:

$$L(n) = k \cdot (\alpha \cdot |n| + 1) e^{-\alpha \cdot |n|} \quad (3.1)$$

L'implantation récursive de cette convolution est réalisée au moyen de deux filtres discrets du second ordre, utilisés en parcourant le signal du début vers la fin, puis en sens inverse.

La détection des fronts est réalisée en convoluant le signal résultant de ce lissage avec la dérivée de l'opérateur de lissage lui-même:

$$D(n) = k \cdot n \cdot e^{-\alpha \cdot |n|} \quad (3.2)$$

dont l'implantation récursive est analogue à celle du filtre de lissage.

La réponse de cet opérateur le long de chaque image-ligne indique la valeur de l'amplitude et le signe de la dérivée du signal vidéo. Les très faibles réponses sont éliminées par seuillage de l'amplitude de la réponse de l'opérateur. Une procédure de sélection est ensuite appliquée pour ne garder que les réponses susceptibles de représenter les fronts significatifs: pour chaque intervalle où le signe de la dérivée reste constant, on sélectionne le maximum de l'amplitude de la réponse. Si ce maximum dépasse le seuil, sa position indique la présence d'un front qui sera retenu comme candidat à l'appariement.

En appliquant cette procédure aux images-lignes, gauche et droite, on obtient deux listes de fronts qu'il s'agit maintenant d'apparier, sachant que chacun de ces fronts est caractérisé par sa position ainsi que par l'amplitude et le signe de la réponse de l'opérateur de Deriche.

## 4. MISE EN CORRESPONDANCE

Le problème de l'appariement des fronts sélectionnés sur les deux images-lignes, gauche et droite, est ramené à un problème d'optimisation. Une fonction d'énergie représentant toutes les contraintes du problème est construite pour que les bons appariements correspondent à des minima locaux de cette fonction. Pour la définir, nous introduisons cinq contraintes compétitives: unicité des appariements, cohérence des pentes des fronts appariés, position des fronts appariés, conservation de l'ordre des appariements et la contrainte de la continuité de la disparité.

Cette fonction d'énergie est ensuite minimisée en utilisant un réseau de neurones de Hopfield [2]. Ce réseau est représenté par un tableau de  $N_g \times N_d$  neurones, où  $N_g$  et  $N_d$  sont respectivement les nombres de fronts de l'image gauche et l'image droite. L'état d'un neurone représente la mesure de la qualité d'appariement entre un front de l'image gauche et un front de l'image droite.

### 4.1. Fonction d'énergie du problème

L'objectif est de trouver une fonction d'énergie représentant les cinq contraintes telle que les bons appariements correspondent à des minima locaux de cette fonction. Pour cela, nous allons traduire chacune des contraintes en terme d'énergie de telle sorte que sa satisfaction corresponde à un minimum de la fonction qui la représente.

*Contrainte d'unicité:* chaque front d'une image ne peut être apparié qu'avec au plus un front de l'autre image. Pour satisfaire cette contrainte, le terme d'énergie qui la représente doit être minimal lorsque la somme des états des neurones de chaque ligne et de chaque colonne vaut 1. Ce terme d'énergie s'écrit comme suit:

$$H_u = \sum_{i=1}^{N_g} \left( 1 - \sum_{k=1}^{N_d} E_{ik} \right)^2 + \sum_{k=1}^{N_d} \left( 1 - \sum_{i=1}^{N_g} E_{ik} \right)^2 \quad (4.1.1)$$

où  $E_{ik}$  (état du neurone  $(i,k)$ ) représente la mesure de la qualité d'appariement entre le front gauche  $i$  et le front droit  $k$ .  $N_g$  et  $N_d$  sont respectivement les nombres de fronts de l'image gauche et l'image droite.

*Contrainte des pentes:* comme chaque front est caractérisé par le signe du gradient, nous apparions que ceux

qui ont le même signe du gradient. La fonction d'énergie représentant cette contrainte peut s'écrire:

$$H_p = \sum_{i=1}^{i=N_g} \sum_{k=1}^{k=N_d} P_{ik} E_{ik} \quad (4.1.2)$$

$$P_{ik} = \begin{cases} P & \text{si les fronts } i \text{ et } k \text{ ont des pentes différentes} \\ 0 & \text{sinon} \end{cases}$$

où  $P$  est une pénalité au cas où les fronts  $i$  et  $k$  n'ont pas la même pente.

*Contrainte des positions:* les positions  $x_g$  et  $x_d$  des fronts appariés doivent satisfaire la contrainte  $x_d < x_g$ . L'expression en terme d'énergie de cette contrainte peut s'écrire comme suit:

$$H_g = \sum_{i=1}^{i=N_g} \sum_{k=1}^{k=N_d} G_{ik} E_{ik} \quad (4.1.3)$$

$$G_{ik} = \begin{cases} G & \text{si } x_i < x_k \\ 0 & \text{sinon} \end{cases}$$

où  $G$  est une pénalité au cas où la contrainte des positions n'est pas satisfaite.

*Contrainte d'ordre:* si un front  $x_{1g}$  est candidat pour être apparié avec un front  $x_{1d}$ , alors il est impossible pour un front  $x_{2g}$  ( $x_{2g} < x_{1g}$ ) d'être apparié avec un front  $x_{2d}$  ( $x_{2d} > x_{1d}$ ). La fonction d'énergie représentant cette contrainte peut être exprimée de la manière suivante:

$$H_o = \sum_{i=1}^{i=N_g} \sum_{k=1}^{k=N_d} \sum_{j=1}^{j=N_g} \sum_{l=1}^{l=N_d} O_{ikjl} E_{ik} E_{jl} \quad (4.1.4)$$

$$O_{ikjl} = T_{ikjl} (1 - \delta_{ij}) (1 - \delta_{kl})$$

$$T_{ikjl} = |s(i-j) - s(k-l)|$$

$$s(a) = \begin{cases} 1 & \text{si } a > 0 \\ 0 & \text{sinon} \end{cases}$$

$$\delta_{ij} = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{sinon} \end{cases}$$

*Contrainte de continuité:* cette contrainte exprime la continuité de la disparité dans la scène. Elle est représentée par le terme d'énergie suivant:

$$H_c = \sum_{i=1}^{i=N_g} \sum_{k=1}^{k=N_d} \sum_{j=1}^{j=N_g} \sum_{l=1}^{l=N_d} C_{ikjl} E_{ik} E_{jl} \quad (4.1.5)$$

$$C_{ikjl} = C(X) = \frac{2}{1 + e^{(X-\theta)}} - 1$$

où  $C_{ikjl}$  est une mesure de compatibilité entre les deux paires de fronts appariés  $(i,k)$  et  $(j,l)$ .  $X$  est la différence des disparités entre les deux paires de fronts appariés  $(i,k)$  et  $(j,l)$ . La valeur du paramètre  $\theta$  doit être choisie de telle sorte qu'une compatibilité maximale (+1) est obtenue pour un bon appariement ( $X=0$ ) et qu'une compatibilité minimale (-1) est obtenue pour un mauvais appariement ( $X \gg 0$ ).

En combinant (4.1.1), (4.1.2), (4.1.3), (4.1.4) et (4.1.5), la fonction d'énergie totale pour le problème de l'appariement des images linéaires a l'expression suivante:

$$\begin{aligned} H = & \sum_{i=1}^{i=N_g} \left( 1 - \sum_{k=1}^{k=N_d} E_{ik} \right)^2 + \sum_{k=1}^{k=N_d} \left( 1 - \sum_{i=1}^{i=N_g} E_{ik} \right)^2 \\ & + \sum_{i=1}^{i=N_g} \sum_{k=1}^{k=N_d} P_{ik} E_{ik} + \sum_{i=1}^{i=N_g} \sum_{k=1}^{k=N_d} G_{ik} E_{ik} \\ & + \sum_{i=1}^{i=N_g} \sum_{k=1}^{k=N_d} \sum_{j=1}^{j=N_g} \sum_{l=1}^{l=N_d} O_{ikjl} E_{ik} E_{jl} \\ & + \sum_{i=1}^{i=N_g} \sum_{k=1}^{k=N_d} \sum_{j=1}^{j=N_g} \sum_{l=1}^{l=N_d} C_{ikjl} E_{ik} E_{jl} \end{aligned} \quad (4.1.6)$$

## 4.2. Minimisation de la fonction d'énergie

Un réseau de Hopfield est utilisé pour minimiser la fonction d'énergie proposée. Pour cela, il suffit de la mettre sous la forme quadratique suivante:

$$\begin{aligned} H = & \frac{-1}{2} \sum_{i=1}^{i=N_g} \sum_{k=1}^{k=N_d} \sum_{j=1}^{j=N_g} \sum_{l=1}^{l=N_d} W_{ikjl} E_{ik} E_{jl} \\ & - \sum_{i=1}^{i=N_g} \sum_{k=1}^{k=N_d} I_{ik} E_{ik} \end{aligned} \quad (4.2.1)$$

où  $W_{ikjl}$  représente le poids de connexion entre le neurone  $(i,k)$  et le neurone  $(j,l)$  et  $I_{ik}$  représente l'entrée externe du neurone  $(i,k)$ :

$$W_{ikjl} = C_{ikjl} - \delta_{ij} - \delta_{kl} - O_{ikjl} (1 - \delta_{ij}) (1 - \delta_{kl}) \quad (4.2.2)$$

$$I_{ik} = \frac{4 - G_{ik} - P_{ik}}{2} \quad (4.2.3)$$

A partir d'un état initial, le réseau se relaxe jusqu'à ce qu'il atteigne un état stable. Le réseau est stable lorsqu'aucun des neurones ne peut changer d'état. Durant la relaxation du réseau, les états des neurones évoluent selon le schéma itératif asynchrone suivant:

$$\frac{du_{ik}}{dt} = -\frac{u_{ik}}{\tau} + \sum_{j=1}^{j=N_g} \sum_{l=1}^{l=N_d} W_{ikjl} E_{ik} E_{jl} + I_{ik} \quad (4.2.4)$$

où  $u_{ik}$  représente l'état interne du neurone  $(i,k)$ .  $\tau$  est un paramètre fixé à 1 qui représente un temps constant.

La relation entre l'état interne et la sortie d'un neurone  $(i,k)$  est donnée par la fonction sigmoïde suivante:

$$E_{ik} = g(u_{ik}, \alpha) = \frac{1}{2} \left( 1 + \tanh \left( \frac{u_{ik}}{\alpha} \right) \right) \quad (4.2.5)$$

où  $\alpha$  est un paramètre qui peut être vu comme le paramètre de température pour un système physique.

Pour déterminer le résultat de l'appariement à partir de l'état final du réseau, on sélectionne les neurones dont les états sont maximum, selon la stratégie suivante. Dans un premier temps, on cherche, dans chaque ligne du réseau, le neurone dont l'état est maximal. On cherche ensuite, dans chaque colonne, parmi les neurones sélectionnés dans la première étape, le neurone qui a une réponse maximale. Chacun des neurones sélectionnés représente une paire de fronts appariés.

## 5. APPLICATION A LA DETECTION D'OBSTACLES A L'AVANT DES VEHICULES

Une séquence acquise avec le dispositif stéréoscopique installé à bord d'un véhicule est présentée sur la Figure 3. Les images vidéo (a) et (b) correspondent à un empilement des images-lignes acquises par les deux caméras linéaires du stéréoscope. L'origine du temps correspond au haut des images.



Figure 3-a: Séquence gauche.

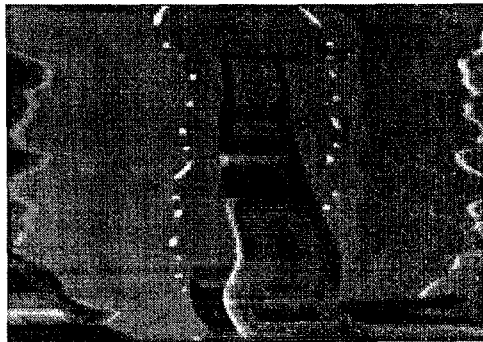


Figure 3-b: Séquence droite.

Le véhicule instrumenté roule plus vite que la voiture qui le précède (R4). Les plans de visée croisent l'arrière de la R4 à une hauteur croissante à mesure que la distance décroît, ce qui explique l'inversion de l'image de cette voiture, dont on commence par voir les roues pour remonter jusqu'au toit.

Plus en avant se trouve un autre véhicule, en train de se rabattre. La brièveté de l'apparition de ce second véhicule est due au mouvement de tangage du véhicule instrumenté.

Par ailleurs, les traces laissées à gauche des images (a) et (b), à la fin de la séquence, proviennent d'une troisième voiture qui dépasse le véhicule instrumenté en troisième position. Sur ces mêmes images, on distingue parfaitement les traces blanches dues à la signalisation au sol.

Cette séquence a été traitée par l'algorithme de mise en correspondance décrit ci-dessus. Sur la scène reconstituée (Cf. Figure 4), on reconnaît bien la forme de la R4. On notera que, contrairement à ce que l'on constatait sur les images du stéréoscope, sa largeur apparente est ici constante dans le temps. La teinte des contours de la R4, de plus en plus foncée, traduit son rapprochement au cours du temps. En effet, la profondeur des contours est codée en niveaux de gris, du blanc qui représente la distance maximale au noir qui représente la distance minimale.

On voit également, en pointillé, les deux lignes blanches qui séparent les voies de l'autoroute.

Le véhicule qui se rabat devant la R4 est clairement visible en bas de la Figure 4, sous la forme d'une ligne claire et continue. L'extrémité de son ombre est visible en bas, à l'extrême droite de l'image.

La voiture qui double le véhicule instrumenté est visible, en bas et à gauche de la Figure 4, comme une ligne oblique, très sombre en raison de sa proximité.

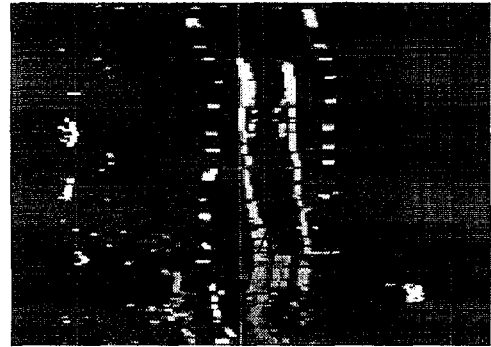


Figure 4: Scène reconstituée.

## 6. CONCLUSION

Dans cet article, nous avons présenté une approche neuronale pour la reconstitution de l'environnement 3-D à l'avant d'un véhicule. Nous nous sommes intéressés plus particulièrement au problème de l'appariement stéréoscopique des images linéaires que nous avons ramené à un problème d'optimisation. Nous avons utilisé un réseau de neurones de Hopfield pour minimiser une fonction d'énergie caractérisant ce problème.

Le principal objectif est la détection en temps réel d'obstacles à l'avant d'un véhicule routier. L'utilisation d'un stéréoscope à base de caméras linéaires permet un gain de temps au niveau de l'analyse des couples d'images stéréoscopiques.

Les résultats obtenus sont satisfaisants et ont montré la bonne robustesse de la méthode.

## REMERCIEMENTS

Les auteurs désirent remercier le G.I.E. PSA-RENAULT pour son aide précieuse, sa contribution financière et sa confiance au cours de ce travail qui s'inscrit dans PROLAB2, un sous-programme du projet EUREKA-PROMETHEUS.

## REFERENCES

- [1] J.-L. Bruyelle and J.-G. Postaire, "Direct range measurement by linear stereovision for real time obstacle detection in road traffic", Robotics and Autonomous Systems, Vol. 11, pp. 261-268, 1994.
- [2] J.J. Hopfield and D.W. Tank, "Neural Computation of decisions in optimization problems", Biological Cybernetics, 52, pp. 141-152, 1985.
- [3] J.-L. Bruyelle and J.-G. Postaire, "Disparity analysis for real time obstacle detection by linear stereovision" Proceeding IEEE Intelligent Vehicles '92, Detroit, pp. 51-56, 1992.
- [4] R. Deriche, "Fast algorithms for low-level vision", IEEE Trans. On Patt. Anal. Mach. Intel., Vol. PAMI-12, n° 1, pp. 5-13, Jan. 1992.