

Système de Vision Temps Réel Embarqué sur Robots Mobiles Quadrupèdes dans le Cadre de la « SONY Legged RoboCup »

Patrick BONNIN^{1,2}, Didier SOLHEID^{1,2}, Vincent HUGEL¹, Pierre BLAZEVIC¹, Kamel BOUCHEFRA²

¹Laboratoire de Robotique de Paris LRP,
10-12 Av de l'Europe, 78140 VELIZY

²IUT de Villeteuse (Université Paris XIII)
Av JB Clément, 93430 Villeteuse

bonnin@robot.uvsq.fr, hugel@robot.uvsq.fr, pierre@robot.uvsq.fr

Résumé – Nous présentons la programmation d'un Système de Vision Temps Réel Embarqué sur un robot quadrupède mobile et autonome, conçu et réalisé par SONY dans le contexte de la RoboCup (compétition de robots jouant au football en équipe). Nous analysons les problèmes rencontrés par le système de vision temps réel : l'éclairage, la puissance de calcul disponible, ainsi que le contexte temps réel dans le cas d'un robot à pattes.

Abstract – We present a software implementation of a real time vision system which is inboard a mobile and autonomous four legged robot, designed and realized by SONY. This implementation was realized for the RoboCup context (competition between robot teams playing soccer). The encountered problems are analyzed : the lighting conditions, the available inboard computing power, the real time context in the case of legged robots.

1. Introduction

1.1 Les Compétitions de Robots Mobiles

Les compétitions de robots mobiles internationales telles que la RoboCup, Bourges et nationales telle que E=M6 sont un cadre propice à l'expérimentation et à la confrontation (coté compétition) des belles théories de recherche « amont » que nous développons dans nos laboratoires. De plus elles favorisent la création d'équipes pluridisciplinaires, car la moindre application de robotique mobile et autonome nécessite une équipe de chercheurs avec les compétences dans les domaines de la Robotique, de l'Automatique, de l'Informatique Temps Réel, de la Vision Robotique (Traitement d'Image classique + contraintes : Temps Réel, embarquabilité etc...) et de l' Intelligence Artificielle.

1.2 La RoboCup

La RoboCup est une fédération internationale proposant des compétitions internationales de robotique mobile ou en simulation, où dans plusieurs catégories de taille et / ou de types, des robots s'affrontent en jouant au football en équipe. La seconde manifestation, organisée à La Villette en Juillet 98, a regroupé 80 Universités à travers le monde. La troisième édition se tiendra à Stockholm en Suède au début Août cet été.

Dans le cadre de la « SONY Legged RoboCup », les équipes sont composée de 3 robots mobiles et totalement autonomes : ils ne communiquent pas ni entre eux, ni avec un

ordinateur hors terrain. Le seul capteur permettant la perception de l'environnement du robot est une caméra couleur. La Vision (que nous différencions volontairement du Traitement d'Image classique, compte tenu des énormes contraintes temporelles de l'application) assure le contrôle du robot : elle assure l'asservissement tout au long de la marche.

1.3 Suite de l'Article

L'objectif de ce papier est de présenter le système de Vision réalisé ou plus exactement programmé, permettant l'interprétation de scènes temps réel à partir d'images couleur en composantes YUV acquises à la cadence de 15 Hertz, intégré sur un robot mobile et autonome de petite taille. Ainsi, le cadre applicatif : le robot et le jeu, est décrit dans le § 2, l'organisation des logiciels développée dans le § 3, les problèmes rencontrés : présentation et solutions apportées dans le § 4, et la présentation des résultats dans le § 5. Enfin, nous concluons sur l'expérience dans le § 6.

2. Le Cadre Applicatif

2.1 Le Robot : Mécanique et Informatique

Le robot (cf Prototype de la figure n°1), sur lequel nous avons implanté le système de vision temps réel, pour l'édition 98 de la compétition est un robot mobile et autonome de petite taille : 25 cm de haut sur 25 cm de long et 10 cm de large, ressemblant à un animal familier de petite taille tel qu'un chat ou un petit chien. C'est bien évidemment

un robot à 4 pattes, chacune étant dotée de 3 moteurs, avec une tête comportant 3 degrés de liberté commandés par 3 moteurs, et d'une queue mobile dotée d'un moteur. La vitesse de déplacement que nous obtenons est à l'heure actuelle d'environ 5 cm par seconde. Il a été conçu et réalisé par SONY [1], et mis à notre disposition dans le cadre de la RoboCup.

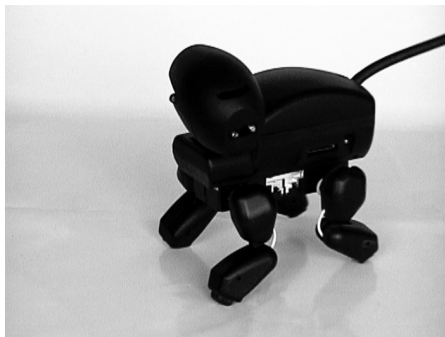


Figure n° 1 : Le Robot « Prototype »



Figure n° 2 : Le Robot « Commercialisé »

La partie matérielle informatique est bâtie autour d'une architecture MIPS série R 4000 [1] et de circuits ASICs pour contrôler les périphériques. La tête comporte une micro-caméra avec un objectif intégré, le tout de taille 23x16x4 mm avec une résolution de 362 x 492 pixels. Un système matériel de détection des couleurs permet de détecter jusqu'à 8 couleurs préalablement définies dans l'espace des couleurs YUV en un temps de 50 ms.

La partie logicielle informatique est constituée d'un Système d'Exploitation Temps Réel Orienté Objet, APERIOS développé par SONY. Les divers objets développés pour la programmation du robot peuvent dialoguer entre eux par plusieurs types de communications par échange de message [1]. Ils peuvent avoir des niveaux de priorité différents.

Le robot est doté d'un gyromètre et d'un accéléromètre, tous deux trois axes. L'accéléromètre permet de détecter la chute du robot et d'enclencher la procédure de relevage.

A l'heure où nous écrivons ce papier, nous « prenons en mains » le nouveau robot, version commercialisée par SONY (3000 exemplaires ont été vendus en 20 mn sur internet au prix de \$2500 : un succès commercial !). D'un « design » totalement nouveau (cf figure n° 2), il est beaucoup plus performant que le prototype décrit ci-dessus. Au niveau de la vision, outre des performances accrues du processeur, il est possible de piloter par logiciel des réglages de la caméra, tel que le contrôle du gain, ce que nous allons tester.

2.2 Le Jeu et le Rôle du Système de Vision

Le rôle du Système de Vision est de permettre au robot de jouer au football en équipe de 3 joueurs, dans le cadre de la RoboCup [2], donc de **détecter**, de **reconnaître** et de **localiser** dans la scène 3D les divers éléments distinguables grâce à leur couleur : la balle (orange), les balises pour se localiser sur le terrain (rose et soit bleu clair, soit vert ou soit jaune), les buts (bleu clair ou jaune) et les joueurs partenaires et adversaires (rouge et bleu foncé, couleurs inversées à chaque mi-temps).

La Vision est seul capteur permettant l'autonomie du robot. En effet, l'odométrie est certes employée, mais elle n'est pas utilisable seule à ce propos, du fait des chutes fréquentes du robot dues aux collisions entre joueurs en cours de matchs, ou lorsque l'arbitre enlève un joueur ayant commis une faute en cours de match.

3. L'Organisation des Logiciels Développés

3.1 Paris 98

L'ensemble des développements logiciels embarqués, intégrés au système temps réel APERIOS, comprenant 3 objets, ainsi que les informations échangées et les synchronisations entre les objets est schématisé figure n° 3 :

- l'objet « supervision » réalisant la tâche principale de supervision qui gère les tâches de récupération des informations de l'objet vision, d'élaboration d'une stratégie de jeu puis de déplacement du robot, et la génération des consignes à envoyer aux moteurs des pattes pour la marche et de la tête pour suivre la balle,
- l'objet « relevage » réalisant une tâche qui préempte la tâche de supervision lorsque l'accéléromètre détecte une chute du robot, et qui génère aux moteurs des pattes les consignes de relevage,
- l'objet « vision » dont la tâche est essentielle pour - d'une part la localisation des éléments fixes : balises et buts, ou mobiles : autres joueurs et la balle, - et d'autre part la localisation du robot lui même sur le terrain. Toutes ces informations, fournies à la tâche de supervision, lui permettent d'élaborer la stratégie de jeu puis de générer le déplacement du robot.

Dans cette première version (Paris 98), l'objet vision, le moins prioritaire, bouclait sur lui même, réalisant acquisition et traitements. Toutes les informations : balle, balises, buts, joueurs étaient simultanément extraites et l'objet « Locomotion » contenant la partie « intelligente » ne venait récupérer que la partie de l'information nécessaire. Cette option n'était pas pénalisante du fait de la faible priorité de l'objet vision, dont la tâche ne s'accomplissait que lorsque le processeur n'était pas occupé, et que nous n'avions intégré sur le robot que des stratégies réactives, peu coûteuses en temps de calcul.

3.2 Modifications pour Stockholm 99

Pour la seconde version, en cours de réalisation (Stockholm 99), les stratégies sont de plus haut niveau. Il est donc nécessaire de gagner du temps, notamment au niveau de la vision. Pour ce faire, nous rapprochons la vision et la partie intelligente pilotant la vision. Par exemple, pour se localiser, il est nécessaire uniquement de trouver des balises, pour aller à la balle, il est uniquement nécessaire de trouver la balle etc... Le système de vision va fonctionner uniquement par requête de l'objet locomotion. Un agent [3] va avoir pour rôle de mettre en œuvre la séquence adéquate de traitements assurant la requête (cf figure n° 4).

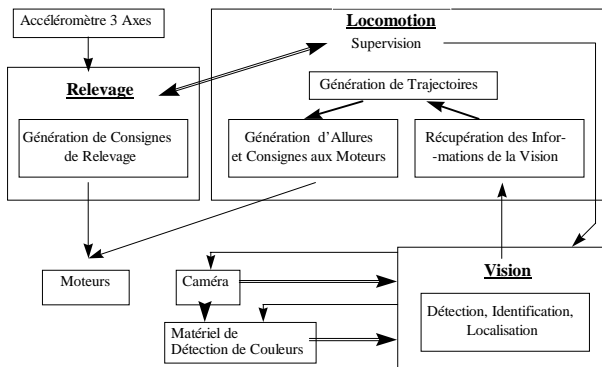


Figure n°3 : Organisation des Développements Logiciels : Les 3 Objets et leurs Interactions.

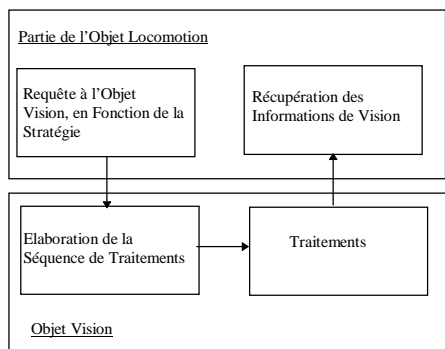


Figure n°4 : Vision par Requêtes

4. Les Problèmes Rencontrés

4.1 Leur Enumération

L'intérêt d'une telle manipulation, hors du contexte « douillet » de son laboratoire favori, est d'affronter des problèmes réels, à mon avis proches de ceux que l'on pourrait avoir sur site industriel

Ainsi, nous avons rencontré trois principaux problèmes lors de la conception et de la réalisation (en terme de programmation, la partie matérielle étant fournie par SONY) du système de vision :

- les **conditions d'éclairage** (découvertes sur place à la Cité des Sciences...), bien que maîtrisées car les compétitions ont lieu en intérieur sur une scène de spectacle, sont loin d'être idéales : projecteurs trop

puissants qui génèrent des reflets sur les objets, ainsi que des ombres portées etc...des conditions similaires se retrouvent dans le milieu industriel.

- la **puissance de calcul** souhaitable, comparée à la puissance de calcul disponible et embarquée dans un système autonome. En effet, les algorithmes de segmentation coopérative multi-spectrale [4], que nous avons développé sur connexion machine CM5 (d'un volume de 1m³) hors contexte temps réel qui pourraient tout à fait être transposés du domaine multi-spectral à la segmentation d'images couleurs, en assurant une segmentation d'image « quasi automatique » (c'est-à-dire utilisant un nombre très restreint de paramètres de contrôle) ne sont raisonnablement pas exploitables, car trop « gourmands » en puissance de calcul (une idée réaliste des architectures temps réel pour le traitement de l'image est présentée dans [5]),
- l'**intégration** du système d'Interprétation de scènes dans une architecture temps réel, dans un contexte de robotique à pattes : la tâche de vision n'étant pas prioritaire devant la tâche de locomotion (contrôler les moteurs des pattes est primordial pour la sécurité du robot), ni le délai entre deux acquisitions d'images successives, ni le temps de latence (incluant le temps de traitement des images) ne peuvent être garantis.

4.2 Solutions apportées

4.2.1 Qualité « Roboticienne » de la Segmentation

Pour **détecter, reconnaître** dans l'image, puis **localiser** les divers éléments la scène 3D, il n'est pas nécessaire d'employer une méthode de segmentation complexe produisant des résultats précis dans l'image. Au contraire, le but est obtenir le plus rapidement possible une localisation 3D « correcte » c'est-à-dire permettant au robot d'aller dans la bonne direction, sans forcément avoir une extraction complète des éléments dans l'image, point que nous soulignerons lors de la présentation.

4.2.2 Détection Temps Réel des Couleurs

Le système de vision du robot, doté d'un matériel spécifique de détection temps réel des couleurs, conçu et réalisé par SONY, nous permet d'effectuer les tâches précédentes à la cadence vidéo de la caméra (15 Hz). En effet, la classification des pixels de l'image en 8 couleurs (au maximum) prédéfinies est obtenu en 50 ms par une procédure à base de seuillages, réalisés dans l'espace des couleur YUV, qui effectue une fusion des informations de type « fusion pixel » [6]. Les paramètres de réglage sont nombreux : 128 valeurs de seuil par couleur prédéfinie, et demandent un réglage précis de manière à ce qu'il n'y ait pas d'ambiguïté de classification entre les couleurs (jaune et orange) à cause des problèmes d'éclairage. Nous avons ainsi été amené à réaliser une interface conviviale permettant d'obtenir automatiquement les seuils de manière précise, en un temps suffisamment court (environ 30 minutes, pour la version Paris 98) pour le contexte de la compétition. Celle-ci constitue, en tant que telle, une recherche sur l'ergonomie des outils à

développer dans le cadre d'une expérimentation sur site. Pour la version Stockholm 99, nous pensons réduire de moitié le temps de réglage, tout en diminuant le risque d'erreur de manipulation en automatisant la recherche des zones de calcul des paramètres. Ceci sera possible en utilisant des algorithmes de segmentation coopérative contour / région sur images YUV, adaptés de [4].

4.2.3 Algorithmes Embarqués

Concernant les algorithmes embarqués proprement dits, qui sont, comme tout algorithme de traitement d'image, gourmands en temps de calcul, nous avons cherché à les optimiser à deux niveaux différents :

- leur choix et / ou leur conception,
- leur implantation.

Au niveau de leur conception, nous tentons de diminuer le nombre de balayages de l'image en regroupant les traitements locaux (en chaque pixel) qui peuvent l'être. Une implantation efficace temporellement remet généralement en cause le découpage « historique » en étapes d'une chaîne d'opérateurs de traitement d'image. De plus, nous choisissons des structures de données simples et d'accès rapide.

Au niveau de l'implantation, les balayages de l'image sont effectués linéairement comme en modèle de programmation par flot de données. En fait, nous concevons les algorithmes « bas niveau » implantés sur le robot de manière à pouvoir les implanter sans trop de modifications sur architecture à base de FPGA, toujours dans le cadre de la RoboCup, mais dans la catégorie des « Middle Size », c'est-à-dire moins de 50 cm. Dans cette catégorie, l'équipe conçoit le robot sur tous les plans : mécanique, informatique matérielle et logicielle.

Les diverses étapes du traitement embarqué sur le robot, constituant le Système Temps Réel d'Interprétation de Scènes à partir d'Images Couleurs sont :

- la détection des couleurs, utilisant le matériel spécifique, produisant, au bout d'un délai de 50 ms, une image de profondeur un octet où chaque bit représente la détection ou non de l'une des 8 couleurs prédéfinies,
- l'« ouverture » simultanée sur chacun des 8 bits de l'image résultant de la détection des couleurs, par un élément structurant 3x3 centré 8-connexe,
- l'Extraction des Composantes Connexes,
- la suppression de composantes connexes trop petites,
- la fusion de composantes connexes « proches » et de même couleur en ensembles de composantes connexes (non connexes, bien évidemment !),
- l'identification des divers éléments : la balle, les balises, les buts, les autres joueurs,
- la localisation des éléments précédemment identifiés,

Les résultats obtenus sont corrects aussi bien dans l'image (cf figure n°4) que dans l'optique de la robotique mobile : la balle est suivie durant la marche par la tête du robot avec mouvement fluide. La cadence des traitements mesurée de deux manières différentes est de 15 Hz, c'est-à-

dire la cadence vidéo. A cette cadence, le robot suit la balle de la tête, mais également détermine la présence et la localisation de tous les autres éléments de la scène : balises, buts, joueurs, ceci tout en marchant.

5. Présentation des Résultats

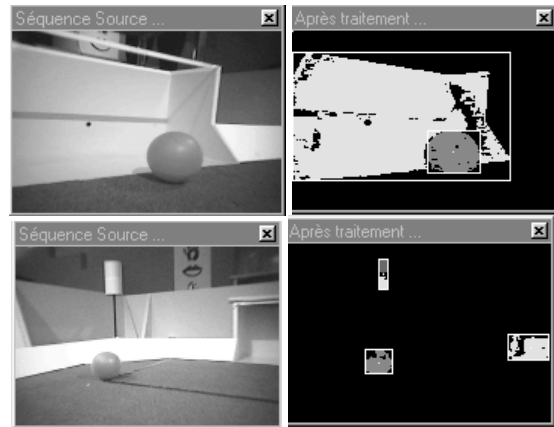


Figure n°4 : Résultats en Image ...

6. Conclusion

En guise de conclusion, nous tenons à souligner l'intérêt d'une expérimentation (implantation dans un système physique et non en simulation!) qui nous a permis de nous confronter à de nombreux problèmes concrets, du système de vision, mais également de l'intégration des différents objets locomotion / stratégie, vision et relevage, pour fonctionner ensemble et assurer au robot mobile et autonome un comportement satisfaisant. Nous nous sommes également confrontés à d'autres équipes dans le cadre du challenge de la RoboCup.

7. Bibliographie

- [1] M. Fujita, S. Zrehen, H.Kitano., *A Quadruped Robot for RoboCup Legged Robot Challenge, Proceedings of the second RoboCup Workshop, Paris Juillet 98*
- [2] M. Asada, *Proceedings of the second RoboCup Workshop, RoboCup-98 : Robot Soccer World Cup II, Paris Cité des Sciences et de l'Industrie Juillet 98*
- [3] K.Bouchebra, V.Hugel, P.Blazevic, P.Bonnin, D.Duhaut, A.El Fallah Seghrouchini, *Situated Agents with Reflexive Behavior, IJCAI RoboCup Workshop Stockholm 99*
- [4] P.Bonnin, B.Hoeltzner, E.Pissaloux, *A new way of image data fusion : the multi-spectral cooperative image segmentation, IEEE Int Conf on Image Processing ICIP 95 Washington*
- [5] E.Pissaloux, P.Bonnin, *On the evolution of parallel computers dedicated to image processing through examples of french computers, Digital Signal Processing, Academic Press, vol 7, n°1, January 1997*
- [6] M. Mangolini, *Apport de la Fusion d'images satellitaires multiscapteur au niveau pixel en télédétection et photo-interprétation, Thèse de l'Université de Nice Sophia Antipolis 1995.*