



Traitement, Synthèse, Technologie et Applications

BIARRITZ - Mai 1984 -

ANIMA 2 : UN SYSTEME GENERAL DE VISION POUR LA ROBOTIQUE

ANIMA 2 : A GENERAL VISION SYSTEM FOR ROBOTICS

Didier JUVIN - Bertrand de COSNAC

Centre d'Etudes Nucléaires de Saclay - DEIN/SIR - 91191 GIF-SUR-YVETTE CEDEX

RESUME

La grande variété des problèmes que doit résoudre un système de vision dans les applications robotiques lui impose d'être réalisé sous forme d'une structure modulaire et évolutive.

Le système ANIMA 2 permet à l'utilisateur de privilégier :

- la rapidité d'exécution (nombre de formes/sec)
- les performances du prétraitement (adaptativité au mauvais éclairage)
- les performances à la reconnaissance (précision)
- le prix de revient de la machine

en choisissant la combinaison des modules logiciels et matériels d'ANIMA 2 répondant pour le mieux au problème posé.

Au cours de la première partie, nous présentons deux méthodes de pré traitement délivrant un codage des silhouettes de tous les objets et trous présents dans l'image.

La première méthode exige une image de bonne qualité : après une binarisation de l'image par seuillage et un codage par plages réalisé par un matériel spécifique temps réel, un algorithme de liaison très rapide (60 millisecondes par objet) recherche les connectivités de chaque plage.

La 2ème méthode utilise l'information apportée par les niveaux de gris dans l'image : nous avons développé un algorithme original de suivi de contours basé sur la notion d'arborescence de lignes combinée à une "fonction de coût" quantifiant la qualité du contour détecté. Nous présentons une heuristique de parcours à coût "raisonnable" de l'arborescence en un temps minimal, ce qui équivaut à suivre les lignes de contour probable dans l'image. Cette méthode permet d'extraire les contours dans le cas d'image peu contrastées ou bruitées, mais est plus lente que la précédente (de 0,15 à 0,5 s).

La 2ème partie décrit deux techniques de reconnaissance compatibles avec les modules précédents.

Si, un seul tri rapide sur les objets à reconnaître est nécessaire, un algorithme classique de sélection sur des caractéristiques globales de l'objet (périmètre, surface, moments invariants) sera utilisé.

La 2ème méthode se propose de segmenter le contour et de le coder d'une manière invariante en translation et en rotation. nous présentons ensuite un algorithme de reconnaissance basé sur l'utilisation d'une distance définie sur ce codage invariant, associée à une heuristique d'accélération par tri-sélection. Cet algorithme donne une estimation de la déformation de l'objet présenté et est assez rapide (environ 0,4 s pour reconnaître un objet parmi 10).

Nous présentons en conclusion des applications industrielles du système ANIMA 2, bâti sur le microprocesseur INTEL 8086 commandé, côté utilisateur, à l'aide d'un langage évolué : le ROL.

SUMMARY

In robotics applications, there is a large variety of problems that users would like to solve with a vision system. This leads to a vision system working with a modular and evolutive structure.

The ANIMA 2 system allows the user to give particular attention to :

- the processing time (number of patterns/sec)
- pre-processing performances (adaptativity to poor and non-uniform lighting)
- recognition performances (precision)
- cost

by picking up from ANIMA 2 at his choice the best suited algorithm combination for each particular problem.

In the first part, two pre-processing techniques are presented, giving an outline encoding of every objects and holes in the picture.

The first method requires a good picture quality : a threshold is applied to the image and a strip encoding is performed by a specific hardware. A fast tying algorithm (60 ms/object) looks up for the connectivities between each strip.

The second method deals with the grey level interpretation : we have developed an original edge tracking algorithm based on a line arborescence concept in combination with a "cost" function giving a quantitative assessment of contour quality. We present heuristics for moderate cost minimum time arborescence coverage, which yields the probable contour lines in the image. This method allows contour extraction even in low contrasted and noisy images, but is more time consuming than the first one (from 0.15 to 0.5 second per object).

The second part describes two recognition techniques which are implemented with the preceding modules.

If a fast object classification is needed, a classical selection algorithm based on global object characteristics (perimeter, area, invariant moments) will be used but the second method is more accurate being based on the outline itself.

We first present a contour segmentation encoding which is invariant in the translational and in the rotational modes. We then describe a recognition method based on a distance specified in terms of the above invariant encoding, joined with heuristics accelerated by pre-selection and sorting. This algorithm allows to measure the distortion between the presented object and a reference object and is quite fast (about 0,4 sec to recognise and inspect one object among 10).

We conclude with the presentation of some industrial applications of the system ANIMA 2. This system implemented in an INTEL 8086 microprocessor supports a high-level language : ROL.



ANIMA 2 : UN SYSTEME GENERAL DE VISION POUR LA ROBOTIQUE

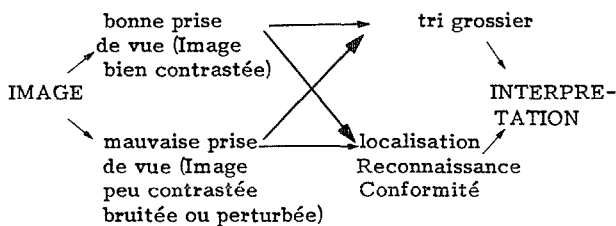
ANIMA 2 : A GENERAL VISION SYSTEM FOR ROBOTICS

D. JUVIN - B. de COSNAC

I - INTRODUCTION

Dans le contexte de la robotique, la forte diversité des demandes au niveau des applications en traitement d'images impose au concepteur d'un système de vision, de prévoir une grande modularité du logiciel, afin de pouvoir s'adapter facilement aux besoins précis des utilisateurs.

La plupart des problèmes posés correspondent à un parcours particulier dans ce schéma :



CRITERE DE TEMPS : critique ou non
 CRITERE DE COUT : critique ou non

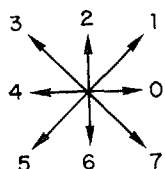
Le système industriel de vision ANIMA 2 possède une panoplie de modules logiciels et matériels permettant à l'utilisateur de privilégier un ou plusieurs critère(s) en fonction de son besoin particulier.

Nous proposons, dans cette communication, quelques solutions techniques retenues et implantées sur le système ANIMA 2 bâti à partir d'un microprocesseur INTEL 8086.

II - PRE-TRAITEMENT : EXTRACTION DU CONTOUR

a) But

Les informations pertinentes à extraire de l'image sont les lignes de contour, que nous codons selon le codage de FREEMAN,



par des déplacements élémentaires selon les 8 directions de base. Un tel codage permet d'enregistrer d'une manière compacte, une ligne quelconque dans l'image.

En fonction de la qualité de la prise de vue, l'extraction des lignes de contour nécessite l'interprétation plus ou moins fine des niveaux de gris présents dans l'image.

b) Images bien contrastées

Dans le cas où la prise de vue est de bonne qualité et où un simple seuillage est suffisant, il reste à regrouper les points appartenant à la même "tache", afin de pouvoir isoler les objets présents dans l'image, à l'aide d'un algorithme de connexité rapide. Celui-ci se décompose en 2 parties :

. connexité par ligne

2 points d'une ligne sont dits connexes s'ils sont dans un même état binaire et adjacents. Les zones connexes dans une ligne sont constituées d'une série de points adjacents dans le même état binaire. Il suffit de stocker les coordonnées de début et de fin de ces zones pour restituer la ligne sans ambiguïté. On a réalisé ainsi un "codage par plages" qui peut facilement être câblé au standard vidéo.

. connexité par colonne

En adaptant la même définition de connexité sur les colonnes, l'algorithme revient à chercher à relier des plages adjacentes entre elles et à créer des blocs connexes correspondant aux taches présentes dans l'image. Cependant, une telle réalisation

nécessite beaucoup de manipulations de tableaux coûteuses en temps de calcul et rend délicate la reconstitution des enveloppes de ces taches (qui sont les lignes de contours) dès que les objets présentent des épaules ou des jambages,

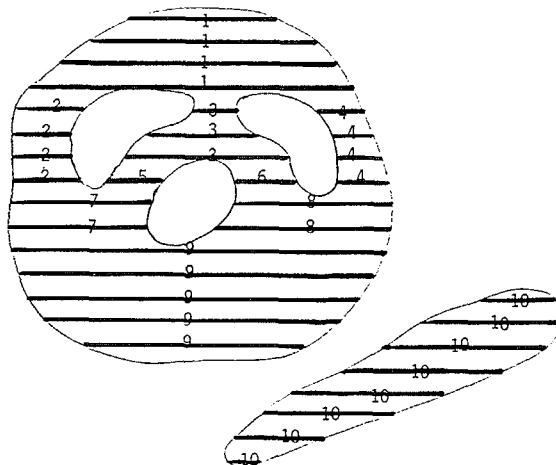


ce qui est pratiquement toujours le cas.

L'algorithme proposé agit en 2 temps :

- dans un premier temps, on "relie" les plages connexes entre elles en les enregistrant dans des tables, avec pour convention d'ignorer la connexité entre des plages présentant des épaules ou des jambages. On établit simultanément une table de liaison entre ces blocs.

Exemple :



Notre algorithme délimite :

Nom	Liens	Provenance
1	2,3,4	NEANT
2	7	1
3	5,6	1
4	8	1
5	7	3
6	8	3
7	9	2,5
8	9	6,4
9	NEANT	7,8
10	NEANT	NEANT

- dans un deuxième temps, on "relit" les tables et l'on génère le codage de Freeman des enveloppes des taches (objets et trous) à l'aide d'un algorithme de décision réentrant. Cet algorithme est très rapide (60 ms par objet) car il évite toute recopie de tableaux.

c) Images faiblement contrastées ou bruitées

Dans ce cas, l'examen des niveaux de gris se révèle nécessaire, les lignes de contour étant les lignes à forte variation de niveau.

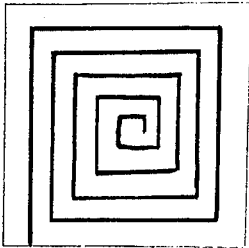
Une analyse systématique de tous les points de l'image est très coûteuse en temps de calcul si elle est faite séquentiellement, c'est pourquoi nous avons développé un algorithme de poursuite dynamique des contours. Il consiste à rechercher dans l'image des points de contours, puis suivre ces contours en examinant les propriétés

ANIMA 2 : UN SYSTEME GENERAL DE VISION POUR LA ROBOTIQUE
 ANIMA 2 : A GENERAL VISION SYSTEM FOR ROBOTICS

D. JUVIN - B. de COSNAC

locales de l'image à l'aide d'heuristiques permettant de poursuivre même en cas de bruit ou de perturbations locales :

- dans un premier temps, on cherche une "amorce" de contour en calculant en chaque point d'un parcours arbitraire sur l'image, le gradient $G(J, x, y)$. Un point est considéré comme amorce de contour s'il est à gradient localement maximal.



Un exemple de parcours arbitraire sur l'image

- dans un deuxième temps, on cherche à suivre le contour trouvé à l'aide de notre algorithme de poursuite dynamique. Cet algorithme consiste, partant d'un point de contour P, à essayer de déterminer un nouveau point de contour Q en examinant l'environnement local de P dans les trois directions adjacentes à la direction donnée par le vecteur élémentaire de contour en P. Cet examen local se fait en calculant une "fonction de coût" exprimée entre le point P et chacun des 3 points "candidats" Q_i :

$$F^c(J_p, G(J_p, x_p, y_p), J_{Q_i}, G(J_{Q_i}, x_{Q_i}, y_{Q_i}))$$

Cette fonction de coût indique la "qualité" du point Q_i à appartenir à la ligne de contour [3].

L'algorithme prend alors pour nouveau point de contour le candidat ayant le coût minimal si néanmoins :

- l'écart entre la nouvelle direction du contour et les précédentes n'est pas trop brutal (pour éviter de suivre des perturbations trop importantes dans une zone bruitée)

- aucun des 3 points n'est un bord d'image ou un contour déjà traité.

Si l'un de ces cas est rencontré, l'algorithme abandonne la poursuite et retourne à la 1ère phase d'exploration.

Un tel algorithme est assez rapide : il met environ 300 ms pour extraire le contour d'un objet dans une image peu contrastée et présente de bonnes performances au niveau de l'insensibilité au bruit (voir figure 1).

III - RECONNAISSANCE

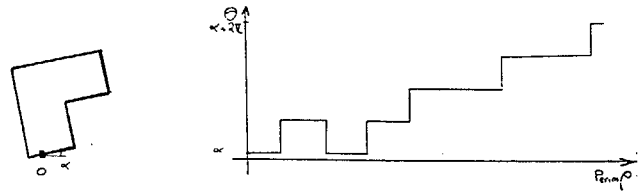
Les critères de reconnaissance pour un système de vision dépendent de l'application envisagée.

Dans les cas où un tri rapide est suffisant (par exemple vérification de la présence d'une pièce déterminée sur la scène), un algorithme simple de reconnaissance basé sur le calcul des paramètres invariants en translation et rotation (périmètre, surface, moments invariants) [4] peut suffire et est disponible dans le logiciel d'ANIMA 2. Cependant, la description d'un objet par ces seuls paramètres de nombre très limité (une dizaine), peut amener des confusions à la reconnaissance entre des objets ayant des aspects visuels très différents.

C'est pourquoi, nous avons développé un algorithme de reconnaissance performant, travaillant directement sur la ligne de contour segmentée, et donc très précis (probabilité de confusion théoriquement nulle pour 2 objets d'aspect visuel différent).

Cet algorithme agit en 3 étapes :

- il consiste à coder le contour segmenté d'une manière invariante en translation et en rotation par un codage polaire curviligne qui consiste à enregistrer l'abscisse curviligne des extrémités de chaque segment, ainsi que l'orientation correspondante. Ce codage peut être reporté sur un repère cartésien ; on appelle alors une telle représentation codage développé de la forme.

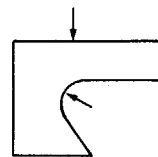
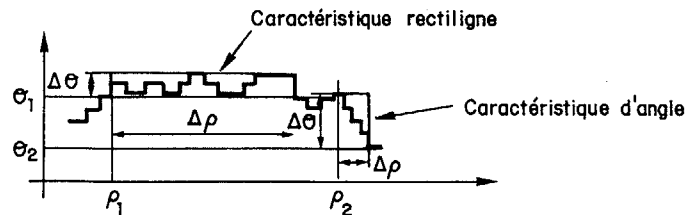


Une forme et son codage développé

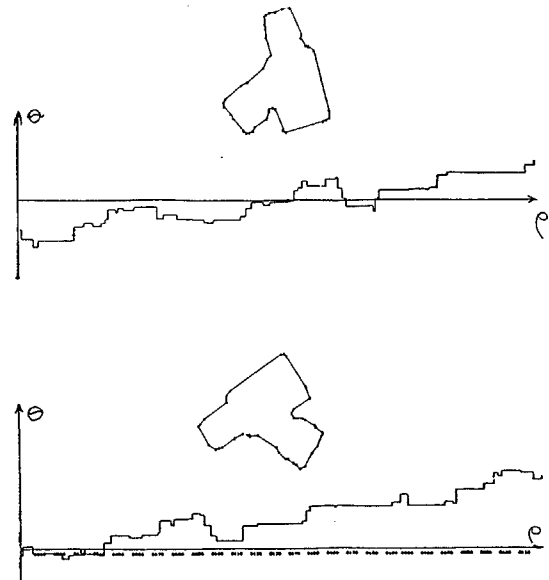
Dans un deuxième temps, l'algorithme cherche à appairer des caractéristiques grossières issues de ce codage : il s'agit de succession de segments dont le codage développé s'inscrit dans un rectangle tel que :

. $\Delta\theta \ll \Delta\rho$, ce qui correspond sur l'image à une caractéristique rectiligne

. $\Delta\theta \gg \Delta\rho$, ce qui correspond sur l'image à une variation d'angle

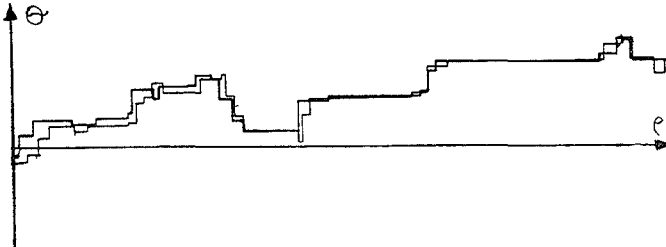


L'appariement permet d'effectuer un tri-sélection sur les objets les plus probables, et dans un 3ème temps, l'algorithme évalue la différence entre le codage développé de l'objet inconnu et celui de l'objet sélectionné, par superposition des deux codages développés.





Superposition la plus favorable donnée par la corrélation :



L'utilisation d'une distance entre formes à partir d'un codage de la ligne de contour segmenté associé à un algorithme de tri-sélection permet d'obtenir de très bonnes performances à la reconnaissance, et permet d'estimer la déformation de la forme présentée par rapport à une forme de référence. Les possibilités de confusion pour deux objets d'aspect différent sont théoriquement nulles ; cette méthode permet, en outre, de situer des points caractéristiques directement sur la ligne de contour (points de prise). [6]

IV - CONCLUSION

Nous avons présenté quelques algorithmes de pré-traitement et de reconnaissance implantés sur le système ANIMA 2. Ces modules sont appelables à l'aide d'un langage évolué ROL comprenant des fonctions de haut niveau en vision et robotique.

La grande souplesse de ce système en fait un outil adapté à la résolution de nombreux problèmes réels de traitement d'image en vue d'automatisation.

Nous présentons en annexe 3 applications de vision industrielles traitées par le système ANIMA :

- saisie automatique d'objets sur un tapis roulant
- inspection de conformité des touches d'un clavier de terminal téléphonique
- inspection visuelle d'un cordon de soudure en milieu hostile.

BIBLIOGRAPHIE

- (1) JC SIMON, A. ROSENFELD
Digital Image Processing and analyses
- (2) W. GEISLER
A vision system for shape and position recognition.
Proc. of the 2nd International Conference on Robot Vision, Stuttgart, 1982
- (3) B. DUPEYRAT, D. JUVIN
ANIMA (Analyses of Images) a quasi real time system
Proceeding of PRIP 82, LAS VEGAS
- (4) D. JUVIN
Contribution à la reconnaissance automatique des images appliquée à la robotique. Thèse de D.I. (janvier 83) Paris XI
- (5) T. PAVLIDIS
Structural Pattern Recognition - Springer Verlag, Berlin 1977
- (6) D. JUVIN
Une méthode heuristique de classification et d'inspection automatique d'objets : le système de reconnaissance d'ANIMA - 4ème congrès Reconnaissance des Formes et I.A. (Paris janvier 1984)

ANIMA 2 : UN SYSTEME GENERAL DE VISION POUR LA ROBOTIQUE
ANIMA 2 : A GENERAL VISION SYSTEM FOR ROBOTICS
D. JUVIN - B. de COSNAC

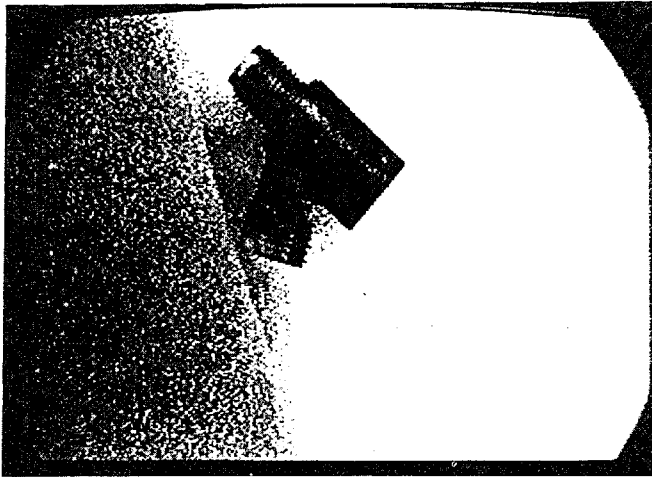
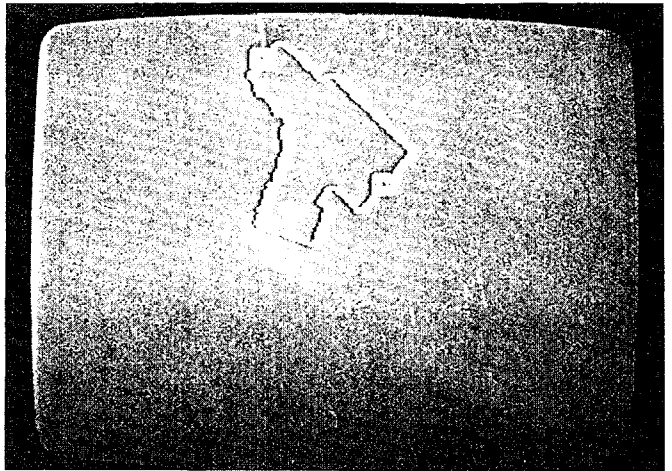


Image présentée



Reconnaissance : points de saisie

FIGURE 1 : Saisie automatique d'objets sur un tapis roulant

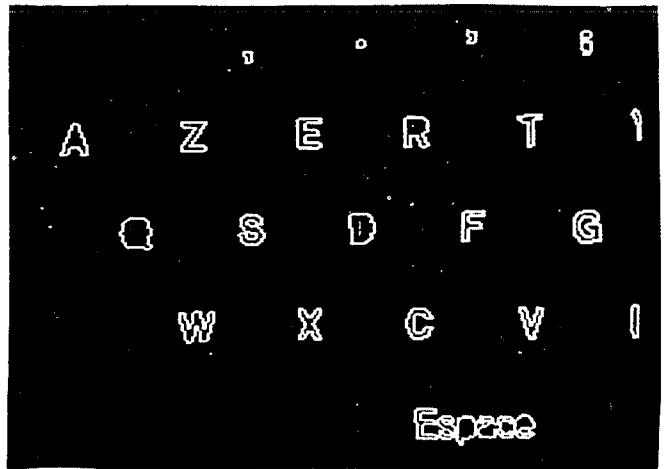
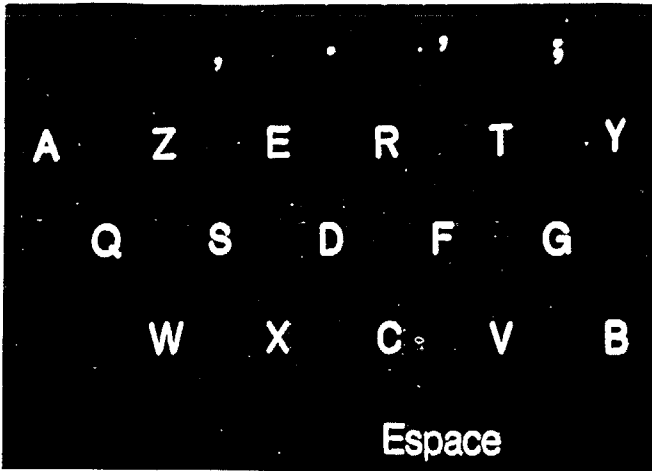


FIGURE 2 : Inspection de conformité d'un clavier de terminal téléphonique

Figure 3 : Inspection visuelle d'un cordon de soudure

