

Traitement des contours en reconnaissance de formes visuelles: Application en robotique

Thèse N° 387 (1980)

PRÉSENTÉE AU DÉPARTEMENT D'ÉLECTRICITÉ
ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES TECHNIQUES

PAR

JEAN-DANIEL DESSIMOZ

ingénieur-électricien EPFL
originaire de Conthey (Valais)

Acceptée sur proposition du jury

Prof. F. de Coulon, rapporteur
Prof. C. W. Burckhardt, corapporteur
Dr. G. Giral, corapporteur

à Nisa,

sans sa compréhension et
sa patience, cette thèse
n'aurait pas été possible

TABLE DES MATIERES

REMERCIEMENTS	iv
RESUME	v
1. RECHERCHES ACTUELLES EN ROBOTIQUE	1
2. ACQUISITION ET FILTRAGE D'IMAGE	5
2.1 Acquisition d'image	5
2.2 Problèmes d'éclairage	6
2.2.1 Episcopie	6
2.2.2 Pièces en mouvement	7
2.2.3 Eclairage fluorescent	7
2.2.4 Mesures de la troisième dimension	8
2.3 Numérisation d'image	9
2.4 Filtrage d'une image	11
2.4.1 Filtres linéaires	11
2.4.2 Filtres non-linéaires	13
2.4.3 Fenêtres	15
2.5 Dépendance, causalité et récursivité	16
2.6 Forme et Texture	18
3. DETECTION DE CONTOURS	20
3.1 Segmentation d'image	20
3.1.1 Analyse de connexité	20
3.1.2 Morcellement par agglomération et division	21
3.1.3 Poursuite de contours	22
3.2 Poursuite et squelettisation des contours	23
3.2.1 Poursuite de bord	24
3.2.2 Squelettisation de lignes	25
3.2.3 Raffinements possibles	27
4. ECHANTILLONNAGE ET LISSAGE DES COURBES	31
4.1 Intervalle d'échantillonnage minimum	31
4.1.1 Courbes paramétrées	32
4.1.2 Pas d'échantillonnage et de quantification	32
4.1.3 Contenu spectral	35
4.1.4 Exemple commenté	39

4.2 Lissage de courbes bruyantes	41
4.2.1 Interpolation entre quelques points	44
4.2.2 Approximations par splines	45
4.2.3 Approximation par segments de cercles	48
4.2.4 Amélioration à l'aide de tables	48
4.2.5 Filtrage des courbes	49
4.3 Lissage de courbes unidimensionnelles	54
5. EXTRACTION DES CARACTERISTIQUES	57
5.1 Caractéristiques simples.	57
5.2 Mesures composites	59
5.3 Mesures statistiques	60
5.4 Combinaisons complexes	62
5.4.1 Transformation polaire	62
5.4.2 Transformation de Fourier	63
5.4.3 Caractéristiques structurelles	64
5.5 Caractéristiques locales ou globales	65
5.6 Propriété d'invariance	65
6. IDENTIFICATION	68
6.1 Methodes corrélatives	68
6.1.1 Fonction de corrélation	68
6.1.2 Substituts de la fonction de corrélation	70
6.2 Classification	71
6.2.1 Espace des caractéristiques	72
6.2.2 Différence entre événements	72
6.2.3 Définition de classes (apprentissage)	73
6.2.4 Identification ou classification	74
6.3 Methodes structurales	74
6.3.1 Arbres	75
6.3.2 Grammaires	75
6.4 Relations entre techniques de comparaison	77
7. APPLICATION EN ROBOTIQUE	79
7.1 Problèmes... et solutions	79
7.1.1 Contraintes industrielles	83
7.1.2 Pièces qui ne se touchent pas	84
7.1.3 Pièces plates avec recouvrement partiel	87

7.2 Montage experimental	88
7.2.1 Capteurs	89
7.2.2 Unites de traitement	92
7.2.3 Deplacement des objets	93
7.3 Acquisition d'images	95
7.3.1 Quantification de l'intensité lumineuse	95
7.3.2 Echantillonnage de l'image	97
7.3.3 Extraction des contours	98
7.4 Poursuite et squelettisation des contours	101
7.4.1 Images binaires	101
7.4.2 Images à niveaux de gris multiples	102
7.6 Mesures de caractéristiques	108
7.6.1 Caractéristiques globales	108
7.6.2 Estimation de courbure	110
7.6.3 Vecteur d'orientation	112
7.7 Reconnaissance des pièces et des repères	116
7.7.1 Pièces complètement visibles	116
7.7.2 Contours partiels	117
7.7.3 Repères d'orientation	120
7.8 Test de superposition	121
8. PROBLEMES OUVERTS	124
REFERENCES	127
ANNEXE 1	135
ANNEXE 2	141
BIOGRAPHIE	142

REMERCIEMENTS

Je tiens à remercier le Professeur Frédéric de Coulon, chef du Laboratoire de traitement des signaux et mon directeur de these, pour avoir guidé mes pas dans le domaine du traitement du signal et pour avoir créé un environnement si favorable à mes recherches en robotique.

Je tiens également à remercier le Professeur C.W. Burckhardt. Non seulement il m'a fait profiter de sa vision très eclectique de l'art d'être ingénieur, mais de plus, dans le cadre de mon travail de doctorat, il a fait des suggestions, très tôt sur l'utilisation de la courbure le long des contours, et encore récemment, sur la rédaction de la these.

Merci aux autres membres du jury et en particulier à Monsieur G. Giralt, directeur de recherches au Laboratoire d'automatique et d'analyse des systèmes à Toulouse, pour des critiques très stimulantes et constructives.

Mes remerciements vont encore à mes collègues du Laboratoire de traitement des signaux à l'EPFL. En particulier, je voudrais souligner les innombrables discussions (parfois passionnées!) et les nombreux services que je dois à Panayotis Kammenos. Je suis reconnaissant aussi à Murat Kunt pour m'avoir intégré au réseau impressionnant de ses relations scientifiques; à Jean-Marc Zurcher pour la conception de circuits spécialisés qui ont grandement facilité la partie expérimentale de mes recherches; à Eric Gruaz pour ses solutions astucieuses aux problèmes mécaniques occasionnels et pour son aide quant aux illustrations de cette these.

J'exprime enfin ma gratitude à la Commission d'Encouragement des Recherches Scientifiques, qui a partiellement subventionné mes travaux.

RESUME

Cette thèse présente un travail de recherche en robotique concernant particulièrement l'interprétation d'informations visuelles pour l'automatisation de postes de production industriels.

Elle peut schématiquement se partager en deux sections. La première présente un exposé assez général sur le traitement d'images, traitant particulièrement de techniques compatibles avec les contraintes industrielles de vitesse, de précision, de simplicité et de robustesse. La seconde partie traite d'un système automatique pour le tri et le rangement en magasins de pièces industrielles plates, même sous l'hypothèse de recouvrement partiel.

On a particulièrement développé les paragraphes où, sur des matières déjà connues, des remarques originales sont faites, et les chapitres présentant des contributions nouvelles en traitement d'images. Ainsi les fenêtres de mesures, qui sont largement répandues, répondent ici à un critère précis pour leur taux de superposition; les filtres moyenneur et médian sont analysés pour leur comportement statistique; le modèle des courbes quantifiées, pour la représentation des formes, se compare favorablement à celui des images échantillonnées; la segmentation d'image est redéfinie pour signifier "division de l'image en segments de contours", plutôt que "mosaïque de régions (parfois arbitrairement) fermées".

Les contributions les plus substantielles, semble-t-il, sont les suivantes:

- Un nouvel algorithme de poursuite et de squelettisation des contours. La technique décrite permet une connexité maximale, une assez grande adaptabilité et une rapidité extrême.
- Un filtrage curviligne itératif. Il permet l'extraction ultérieure très précise des caractéristiques (de formes) d'une image. Basé sur des considérations similaires à celles qui conduisent au filtre de Wiener, il permet d'échanger de l'insensibilité au bruit contre de la définition des contours dans les hautes fréquences spatiales.

- La conception et la réalisation sur miniordinateur d'un système de reconnaissance et de localisation, qui permet d'automatiser une opération de tri et de mise en magasins pour des pièces industrielles plates entièrement visibles. On utilise en particulier l'information de courbure le long des contours pour reconnaître le type et pour estimer l'orientation des pièces de la scène.
- L'adaptation et l'extension du système ci-dessus pour traiter les pièces qui se chevauchent et sont visibles partiellement (pièces en vrac). La résolution des niveaux de gris est affinée, les contours doivent être amincis, et après reconnaissance et localisation de chaque segment de contour un test de superposition vérifie que la pièce correspondante est accessible.

1. RECHERCHES ACTUELLES EN ROBOTIQUE

La robotique est une discipline qui traite de systèmes autonomes, appelés "robots", comprenant schématiquement trois parties: Des capteurs d'information, des unités de traitement de l'information acquise, et des actuateurs mécaniques qui modifient l'environnement en conséquence. Bien qu'autonomes, les robots incluent généralement une importante fonction de communication avec les opérateurs et les autres machines.

La robotique fait respectivement appel à la physique, à l'électronique, et à la mécanique pour chacune de ces parties, mais elle emprunte principalement ses concepts à la théorie du contrôle et à la psychologie traditionnelle: C'est une discipline "cybernétique" en cela qu'elle organise des systèmes selon le modèle du corps humain, imitant bien des mécanismes régulateurs, de la simple boucle de contre-réaction jusqu'à l'intelligence (Saridis /1/ montre bien comment l'intelligence artificielle s'intéresse à la théorie du contrôle).

La robotique se développe maintenant très vite. D'une part la technologie adéquate existe, permettant à l'outil robotisé de produire autant qu'avant, pour une quantité globale de travail humain moindre. D'autre part un surcroît de production est nécessaire pour satisfaire les besoins des pays moins développés ... et ceci sans que les pays industrialisés en pâtissent.

La robotique émerge pour, en principe, remplacer des travailleurs (au moins partiellement) par des moyens de production non-humains. On a donc la satisfaction éthique de réduire l'asservissement de l'homme à son travail. Cependant dans la mesure où la robotique atteint son but, d'autres problèmes, moindres, apparaissent sur les terrains politique et social. D'une part il faut étudier une nouvelle loi de répartition des biens produits, moins basée sur la notion de travail fourni, et d'autre part, il faut créer des "jeux", évoluer vers une société des "travaux-loisirs". La différence entre le travail traditionnel et le jeu me paraît résider, aujourd'hui, dans la satisfaction à très court terme que le jeu apporte par opposition aux satisfactions indirectes et lointaines auxquelles le salaire donne droit. La situation doit se généraliser, où les notions de travail et de jeu se confondent.

Ceci n'est pas utopiste, il existe déjà des privilèges qui, bien que leur part de biens soit absolument garantie, n'en vont pas moins "travailler".

Bien que conceptuellement partis du modèle humain, les robots ne sont guère antropomorphiques. Ils se sont rapidement spécialisés, améliorant les capteurs (sensibilité, précision, diversité) et les actuateurs (puissance, précision, diversité). Même les capacités logiques et arithmétiques de l'unité de traitement d'information sont parfois décuplées (calculs, démonstration de théorème, régulation...). Les robots s'imposent d'abord en environnement non viable ou dangereux pour l'homme tel que l'espace ou les ateliers de fonderie. Depuis quelques années, ils sont étudiés par de nombreux centres de recherches en vue d'applications industrielles et ils commencent à envahir les unités de production /2,3,4,5/.

Commencés en 1976 dans le cadre pluridisciplinaire d'un "Projet d'Ecole" sur l'automatisation industrielle, nos recherches ont pris un caractère plus théorique dès 1978, s'orientant vers une thèse de doctorat. Spécifiquement en robotique, nos travaux s'apparentent à certaines recherches développées à l'université d'Edimbourg /6/, et aux Etats-Unis /7,8/. Cependant, ils s'inscrivent dans le domaine plus vaste du traitement des signaux maintenant classique, et plus particulièrement du traitement d'images en passe de le devenir.

Notre étude concerne un type de capteur particulier et une unité de traitement de l'information. Elle se restreint à l'acquisition et à l'analyse d'informations visuelles. Le but est de connaître sous la forme la plus abstraite possible l'environnement d'un robot, en utilisant la lumière comme support d'information. Les moyens mis en oeuvre sont limités par des exigences industrielles de rentabilité.

A côté de remarques originales sur des matières connues (cf. paragraphes traitant des fenêtres de mesures, de la parenté entre filtres médian et moyenneur, de la dualité entre échantillonnage d'images et quantification de courbes, de la segmentation d'images, du groupement des caractéristiques et des méthodes correspondantes de comparaison), les contributions nouvelles les plus substantielles rapportées dans cette thèse concernent:

- Un nouvel algorithme de poursuite et de squelettisation séquentielle des contours,
- un filtrage curviligne itératif pour les images numérisées,
- la conception et la réalisation sur miniordinateur d'un système de reconnaissance et de localisation de pièces industrielles multiples qui ne se touchent pas,
- l'adaptation et l'extension du système précédent pour l'identification et la localisation de pièces partiellement cachées (pièces en vrac).

Afin de valider par l'expérience diverses approches et pour illustrer les résultats obtenus, un petit système robotique complet a été mis sur pied, comprenant une caméra de télévision à vidicon, un miniordinateur et un manipulateur cartésien à quatre degrés de liberté.

Cette étude comprend schématiquement deux parties. La première est théorique, présentant des techniques classiques ou originales en traitement d'images.

Le chapitre 2 traite de l'image, de son contenu en formes et en textures, de son acquisition par différents capteurs, à vidicon ou à semi-conducteurs, de son amélioration par des filtres linéaires, passe-bas, passe-bande, passe-haut, et par des filtres non-linéaires, tables de conversions, seuils, filtre médian... Le chapitre 3 décrit différentes méthodes de segmentation d'images, par analyse de connexité, par transformées, et par poursuite de contours. Il présente aussi un algorithme original de squelettisation séquentielle des contours. Au chapitre 4, plusieurs algorithmes de lissage des courbes se comparent sous un éclairage de la théorie de l'information. L'extraction de caractéristiques visuelles, longueurs, surfaces, moments d'inertie, courbures, transformées, font l'essentiel du chapitre 5. Le chapitre suivant présente divers aspects de la reconnaissance des formes. En particulier, des fonctions de corrélation et des mesures de distances dans l'espace des caractéristiques sont décrites, puis les notions d'analyse structurelle, de variables floues et de grammaires avec attributs sont introduites.

La seconde partie de ce rapport discute les principaux résultats expérimentaux tirés d'essais en automatisation industrielle (chapitre 7).

On s'est attaqué à la reconnaissance et à la localisation de pièces mécaniques. Le problème s'est cantonné à des scènes essentiellement bidimensionnelles, sans texture. Si des pièces tridimensionnelles se présentent, elles doivent offrir un nombre fini de profils (en particulier pile et face pour les pièces plates). De plus, le profil ne doit pas trop varier lorsque les pièces se superposent.

Deux solutions sont apparues, l'une pour le cas où les pièces ne se touchent pas, et la seconde pour celui où des objets plats peuvent se superposer à quelques niveaux.

2. ACQUISITION, NUMERISATION ET FILTRAGE D'IMAGE

Le chapitre 2 traite d'abord de l'acquisition des images. Le contenu de l'image, le choix d'un capteur, les problèmes d'éclairage et de numérisation sont brièvement examinés.

Ensuite l'attention se porte sur les opérateurs de filtrages, linéaires et non-linéaires, qui permettent une amélioration ou facilitent l'interprétation du signal lumineux perçu. Les mesures opérées dans des régions (fenêtres) de l'image (histogrammes, moments statistiques ...) sont interprétées dans une perspective de théorie de l'information.

Enfin la division de l'image en informations de texture et de forme dévoile les limites posées à notre étude.

2.1 Acquisition d'image

Une image est essentiellement la projection d'une scène lumineuse à trois dimensions sur un plan. Il y a donc, en principe, perte d'information.

Dans une certaine mesure, en multipliant les projections selon des angles différents il est possible de connaître la scène dans ses trois dimensions. Cette approche est utilisée en vision stéréoscopique par exemple /9,10/.

Une autre manière, indirecte, de connaître la troisième dimension consiste à projeter un éclairage latéral sur la scène.

Pour notre étude, nous nous sommes limités à des scènes quasi-planes, parallèles au plan de l'image. Une image peut être définie mathématiquement comme une fonction analogique d'intensité lumineuse b_a exprimée par rapport à deux variables orthogonales x et y .

Il existe de nombreux capteurs opto-electroniques capables d'assurer la traduction électrique de scènes visuelles /11/. Tubes (vidicon, plumbicon...), dissecteurs d'image, matrices à semi-conducteurs (diodes, CCD, CID...) diffèrent par beaucoup de

caractéristiques dont les principales sont: Les distorsions géométriques, la résolution en x et y, la sensibilité aux différentes couleurs, la gamme dynamique, le temps de latence de l'information perçue.

De plus en plus, les cameras "solid state" à couplage de charge ou à diodes s'imposent, remplaçant les tubes. Les distorsions géométriques ne sont plus alors limitées que par l'objectif. L'échantillonnage de l'image est assuré par le capteur lui-même. De plus, la gamme dynamique est bien plus grande.

2.2 Problèmes d'éclairage

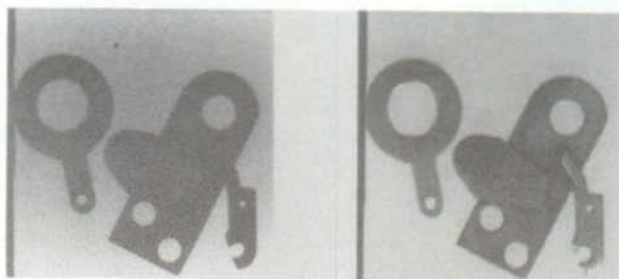


Fig.2.1 Scène perçue en diascopie (gauche) et en épiscopie (droite).

2.2.1 Episcopie

Deux options sont parfois possibles pour éclairer une scène (Fig.2.1). Les petites pièces horlogères par exemple, peuvent facilement se voir sur un fond lumineux (diascopic). Par opposition, la méthode courante utilise les réflexions de la scène (épiscopie). La diascopie présente les avantages d'un excellent contraste entre objet et fond, et d'une absence totale de réflexions parasites sur l'objet. Par contre un plan de travail translucide est généralement mal adapté aux conditions industrielles, surtout si on exige le mouvement (bande transporteuse). D'autre part les informations sur les premiers plans de l'image (obscur en diascopie) sont nécessaires pour traiter les pièces qui se superposent ou qui ont une texture particulière. Nos

expériences montrent que même si on choisit la solution par épiscopie, le contraste entre fond et objet est généralement suffisant. C'est donc la solution que nous avons adoptée.

2.2.2 Pièces en mouvement

Il est bien connu en photographie que si un objet se déplace, le temps de capture de l'information doit être très court pour éviter un "hougé". Deux méthodes sont possibles pour geler le mouvement: l'emploi d'un stroboscope, ou celui d'un obturateur rapide [12,13]. L'obturateur rapide conduit à un temps de capture ($1/2'000$ s) plus grand que le stroboscope ($1/40'000$ s). Il exige un éclairage continu et donc beaucoup plus d'énergie que le stroboscope. Par contre, on peut l'obtenir dans une zone du spectre lumineux où les capteurs sont généralement plus sensibles.

2.2.3 Eclairage fluorescent

Le type d'éclairage nécessaire, continu ou variable, dépend du capteur utilisé; les capteurs intègrent la lumière qu'ils reçoivent durant un certain temps. Ils réagissent donc comme des filtres passe-bas à l'illumination de la scène.

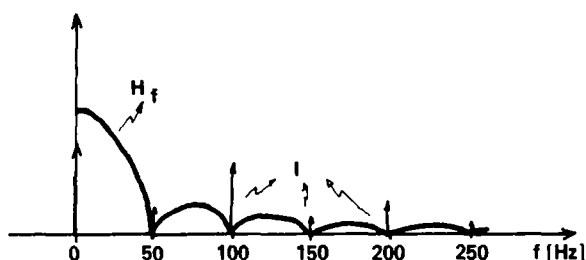


Fig.2.2 L'illumination I (tube fluorescent) a ses composantes périodiques (scintillement) filtrées par la caméra de TV, dont le temps d'intégration est de 20 ms. H représente la bande passante de la cible vidicon balayée selon les normes CCIR.

En particulier pour les cameras de television (tubes a vidicon) l'image due a un eclairage fluorescent est pareille a celle que fournirait un eclairage continu (lampe a filament et tension continue): Les composantes fréquentielles de l'illumination ont des périodes multiples du temps d'integration de la camera (Fig.2.2).

2.2.4 Mesures de la troisieme dimension

La profondeur d'une scene est rarement mesurée directement. Duda et al. /14/ ont utilisé un laser. Par contre une manière indirecte de connaître la troisième dimension consiste a projeter un eclairage latéral sur la scene.

Si la lumiere incidente est formée de lignes ou d'un reseau de points lumineux, la troisième dimension de la scene est alors trahie par leur déformation /15,16,17,18/.

Si l'eclairage latéral est uniforme, on peut néanmoins déduire des ombres portées la position relative des différents plans de l'image. La Fig.2.3 montre comment l'usage d'eclairages rasants, peut faciliter la détection des bords et de la position relative de pièces partiellement superposées. Outre le temps de traitement ralenti, les difficultés additionnelles d'un eclairage pulse, il faut encore remarquer que les ombres portées donnent lieu a deux bords, dont l'un seulement est significatif. C'est pourtant le prix a payer pour percevoir des contours la où les autres méthodes échouent.

Un eclairage latéral uniforme peut aussi donner des renseignements sur l'orientation des surfaces. La quantité de lumière diffusée par une surface varie avec le cosinus carré de l'angle défini par l'eclairage incident et une normale a la surface.

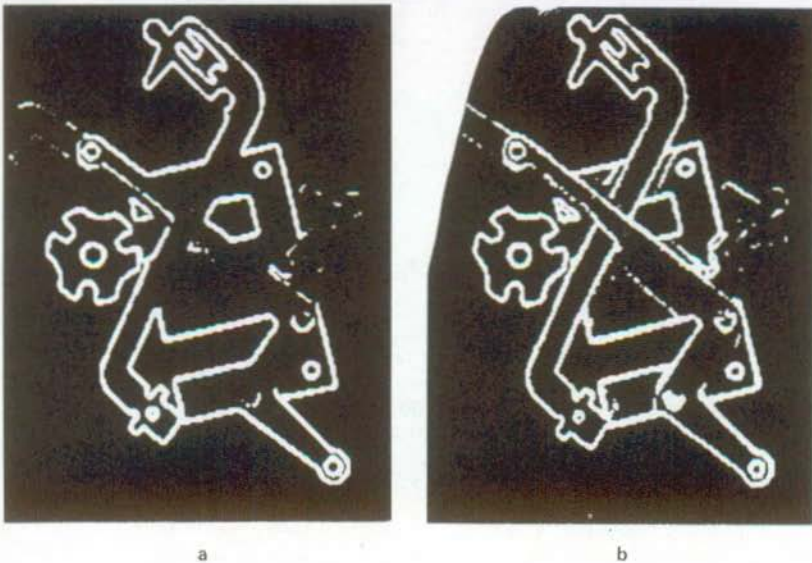


Fig.2.3 Extraction de contours. Une scène est éclairée uniformément (a). Les contours correspondant à quatre éclairages rasants successifs sont additionnés (b). On remarque la sensibilité accrue, au prix d'ombres portées.

2.3 Numérisation d'image

Le théorème de l'échantillonnage est fondamental en traitement des signaux: Un signal à bande limitée peut être représenté par un ensemble de valeurs discrètes (Fig.2.4). Le théorème s'applique aussi à des signaux bidimensionnels. Lorsqu'on le généralise à plusieurs dimensions, différentes stratégies sont possibles /19/. Deux stratégies sont principalement suivies, utilisant une grille d'échantillonnage rectangulaire (la plus courante) ou une grille hexagonale /20,21/. Les hypothèses faites sur le contenu spectral de l'image et les restrictions que l'on pose au traitement ultérieur des échantillons permettent en pratique de résoudre l'alternative. La première grille conduit à une plus grande simplicité des algorithmes, la seconde, pour les images à bande bornée circulairement se satisfait de moins d'échantillons.

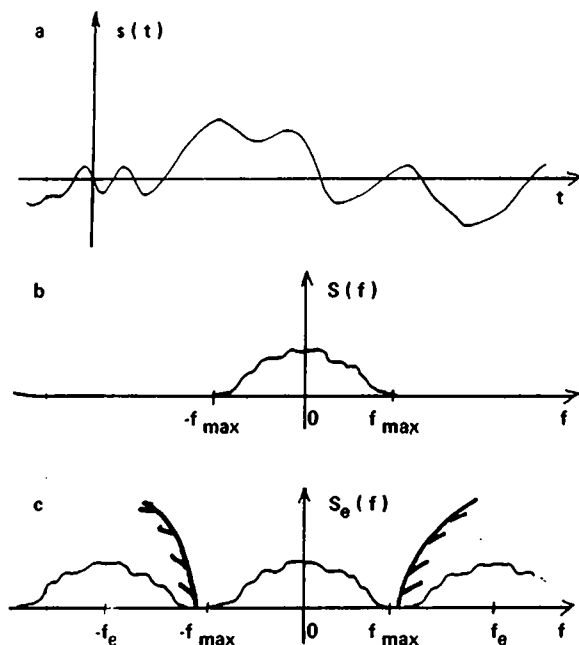


Fig.2.4 Un signal dont le spectre est a bande limitée (b) peut se représenter par des valeurs discrètes. La courbe c montre le spectre du signal $s(t)$ après échantillonnage à la fréquence f_e . La zone hachurée décrit les exigences d'un filtre capable de reconstituer $s(t)$.

La transformée de Fourier a la propriété remarquable de garder la rotation: une image tournée de 30 degrés conduit à une transformée elle aussi tournée du même angle. Comme nos images n'ont pas d'orientation précise, on peut considérer leur contenu spectral comme borne circulairement. Cependant, pour nos applications, l'échantillonnage sur une grille rectangulaire est retenu en raison de la simplicité du traitement qui en résulte:

$$B_a(f_x, f_y) = F\{b_a(x, y)\} \quad (2.1)$$

$$\text{Si } B_a(f_x, f_y) = 0, \text{ pour } \begin{cases} |f_x| > f_{x\max} \\ |f_y| > f_{y\max} \end{cases} \quad (2.2)$$

on peut numériser:

$$b(n_x, n_y) = b_a(n_x \Delta_x, n_y \Delta_y) \quad (2.3)$$

$$\text{avec } \Delta_x < 1/2f_{x\max} \quad \text{et} \quad \Delta_y < 1/2f_{y\max} \quad (2.4)$$

L'intensité lumineuse de chaque échantillon est aussi numérisée. La définition varie entre 2 niveaux et 256 suivant l'application. Une loi de quantification linéaire est en général adoptée.

2.4 Filtrage d'une image

Il est généralement utile de traiter spécifiquement les différentes composantes d'une image. Pour cela on utilise des filtres non-linéaires ou, le plus souvent, des filtres linéaires /22/.

Le filtrage d'image se fait presque toujours dans le domaine spatial. Pourtant, les filtres linéaires sont plus faciles à discuter dans le domaine fréquentiel.

Les fenêtres de filtrage varient en grandeur suivant l'effet recherché. Grandes, elles diminuent fortement l'influence du bruit à large bande présent dans l'image; petites, elles préservent le maximum de définition. Leur usage est tellement répandu qu'elles commencent à s'intégrer dans les "smart sensors" /23/.

2.4.1 Filtres linéaires

Le filtrage d'une image s'exprime par un produit de convolution:

$$b_{af}(x, y) \approx \iint h(\tau, n) b_a(x - \tau, y - n) d\tau dn \quad (2.5)$$

et sous forme numérique:

$$b_f(n_x, n_y) \approx \sum_i \sum_j h(i, j) b_a(n_x - i, n_y - j) \quad (2.6)$$

où $h(x, y)$ est la réponse impulsionnelle (point-spread function) du filtre.

Les filtres linéaires utilisés en traitement d'image forment principalement trois groupes.

Un premier type de filtres regroupe les opérateurs qui lissent l'image, réduisant le bruit granulaire dû à la quantification des niveaux de gris, le bruit introduit par les capteurs opto-électroniques et le bruit dû aux inhomogénéités de surface. Ces filtres ont une caractéristique passe-bas, et gardent la plus grande partie du spectre. Ils ont typiquement une taille comprise entre 2×2 et 5×5 éléments d'image. Bien qu'ils soient faciles à réaliser optiquement, (une légère défocalisation suffit), ils sont généralement réalisés électroniquement pour traiter également les bruits injectés en amont du capteur. Ces filtres, passe-bas, restreignent la bande passante du système d'acquisition. Or déjà avant la conversion analogique numérique (A/N), on ne permet pas une bande passante inutilement large (la limite est fixée par les détails de l'image qui sont intéressants). Les opérateurs passe-bas sont toutefois utiles après numérisation parce que le spectre de l'image ne s'arrête pas brutalement à une fréquence de coupure f_c , mais se prolonge, atténuée, bien au-delà. La fréquence d'échantillonnage est donc choisie en pratique au-delà de $2 \cdot f_c$, ce qui leur laisse un peu de marge. Le chapitre 4 traite de deux critères pour le choix des fréquences de coupure et d'échantillonnage en numérisation d'images.

Un deuxième type de filtre rassemble les opérateurs passe-haut qui permettent de corriger les variations d'éclairage sur la scène. Ils sont facilement réalisés en soustrayant l'image traitée passe-bas de l'image originale. Dans ces applications, le filtre passe-bas englobe au moins 10×10 éléments d'images dans sa réponse impulsionnelle.

Le troisième type de filtre, qui est très répandu, sert à accentuer les bords ou même, à ne garder qu'eux. L'opération de base est la dérivation de l'image. Une simple dérivation donne le gradient. Une double, le

laplacien. On a donc des filtres passe-haut.

En plus, un certain taux de filtrage passe-bas est souvent inclus, ce qui conduit à une grande variété d'opérateurs passe-bande (Sobel, Mero-Vassy, Prewitt...) /18/. Ces opérateurs intègrent le filtrage passe-bas typique du premier groupe à leur traitement essentiellement passe-haut. Le filtre 6x6 de Mero-Vassy /24/ est la convolution d'un filtre passe-bas triangulaire 5x5 par un gradient (Fig.2.5).

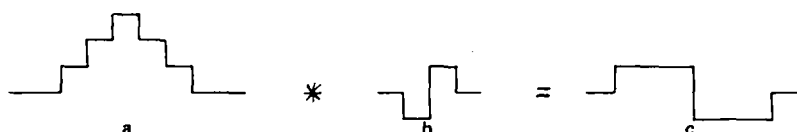


Fig.2.5 Illustration selon un axe de la combinaison d'un filtre passe-bas triangulaire (a) et d'un passe-haut (b), qui conduit au filtre de Mero et Vassy (c).

2.4.2 Filtres non-linéaires

Les filtres non-linéaires utilisés en traitement d'image sont très simples à réaliser mais se prêtent mal à l'analyse théorique. Il en existe une grande variété, mais nous n'en présenterons que cinq sortes.

- Tables de conversion. Il est très courant en traitement d'images que l'on change la caractéristique de quantification des gris. On remplace alors la valeur des gris en chaque point par une certaine fonction. Des conversions exponentielles, logarithmiques ou résultant de mesures de l'image sont courantes. Ceci peut par exemple améliorer la qualité subjective d'une image par augmentation du contraste. Une conversion logarithmique rend le contraste indépendant de l'illumination...
- Seuil. L'opérateur le "seuillage" est essentiellement un comparateur. Le signal d'entrée est comparé à une référence. La sortie du filtre montre alors le signe de

leur différence. La référence est souvent fixe. Elle varie aussi parfois en fonction (linéaire ou non) de l'image.

- Filtre médian. Le filtre médian donne à sa sortie l'échantillon d'entrée qui, dans un domaine donné, a autant de voisins plus grands que lui, que de voisins plus petits. Dans les images binaires, le filtre médian est équivalent à un opérateur "majorité". Le mérite de ce filtre est d'éliminer totalement du bruit ou des détails gênants qui se manifestent de manière isolée, tout en gardant, ailleurs, des transitions très raides.

Le filtre médian (non-linéaire) a une parenté assez grande avec le filtre passe-bas (linéaire). Alors que le filtre passe-bas opère essentiellement une estimation de moyenne le long d'un signal, le filtre médian opère une estimation de médiane. Le chapitre 5 traite plus en détail ces deux mesures statistiques.

Huang & al./25/ donnent un algorithme récursif pour le filtrage médian rapide des images. Justusson /26/ fait une analyse statistique de l'effet du filtrage médian.

- Opérateurs d'expansion et d'érosion. Un autre type d'opérateurs non-linéaires courants en traitement d'images binaires concerne les variations de taille des objets: L'opérateur d'expansion laisse diffuser dans une fenêtre les éventuels points appartenant à des objets. La fenêtre est convoluée avec l'image. Ceci peut être intéressant pour obtenir des surfaces uniformes car les petits trous sont éliminés. Les opérateurs d'érosion, au contraire, permettent d'éliminer les petits objets isolés (parasites) et, partiellement, de séparer des objets qui se touchent. Ce sont alors les éventuels points du fond qui s'imposent, dans la fenêtre. De plus, lorsqu'on leur adjoint un test de connexité, ils permettent d'extraire le squelette des lignes et des objets /27,28,29/.

Ces operateurs ont ete partiellement generalises pour le traitement des images a niveaux de gris. Une fonction de "membership" est associee a chaque point, indiquant combien ils sont blancs ou noirs. Les operations logiques sont alors remplacees par des tests de valeurs minimales et maximales selon les regles habituelles pour les ensembles flous /30/.

- Masques optimaux. Il existe de nombreuses techniques qui essaient une variete de fonctions generalement lineaires sur une image et, ensuite, selectionne celle qui satisfait au mieux un critere. Parmi les operateurs simples et bien connus qui appartiennent a cette categorie, notons en particulier l'operateur de Kirsch /18/ et celui de Frei et Chen /31/. Le premier est un operateur assez sensible utilisant une fenetre de 3x3 points. Quant au second, il presente l'avantage d'etre sensible tout en epaississant peu les bords.

Les masques optimaux sont parfois definis en nombre inutilement eleve /32,33/. Leur decomposition en composantes orthogonales permet la meme sensibilite, au prix de calculs moindres. Par exemple le filtre de Sobel, applique selon deux directions perpendiculaires permet d'estimer un bord selon des directions de $\pi/8$ radians. Le filtre de Hueckel /18/ assure la detection de bords et de lignes a l'aide d'une decomposition en serie de Fourier dans une base polaire.

2.4.3 Fenetres

Très souvent l'image est subdivisée en régions ou des paramètres sont estimés. Par exemple la référence adaptative d'un operateur a seuil, les mesures statistiques de texture, ou encore les histogrammes locaux de gradient sont effectuees dans de telles fenetres. Pour des raisons de symetrie, la zone de dependance de ces caracteristiques se distribue circulairement autour du point de mesure. En principe, la fenetre de calcul devrait etre centree en

chaque point.

La taille des fenêtres peut être vue comme imposée par un filtre passe-bas à réponse unité sur le domaine considéré et nulle ailleurs. De manière plus stricte, on peut examiner la réponse impulsionnelle du "filtre" qui mène au paramètre $p(x,y)$. Si on a une impulsion de Dirac au centre d'une image, il est clair que toutes les fenêtres qui incluent le centre de l'image vont être affectées. On retrouve donc pour $p(x,y)$ la forme de la fenêtre de mesure. Pour une dimension L_x de la fenêtre le long de x , on obtient alors une transformée en $\sin f_x/f_x$ le long de f_x , qui a son premier lobe compris dans le domaine $\{-1/L_x, 1/L_x\}$. Le raisonnement est similaire selon y .

On peut donc appliquer le théorème de l'échantillonnage et conclure que si les fenêtres sont réparties sur l'image avec un pas $L_x/2$ en x et en y , les paramètres en tout point peuvent être connus, par interpolation.

Il faut noter toutefois que ceci est une borne minimale et que, suivant la forme de la réponse impulsionnelle à l'intérieur de la région, la bande passante des paramètres peut être plus élevée, surtout si l'estimation des paramètres inclut des non-linéarités. Les fenêtres devraient donc au moins se superposer à 50% si chaque point de l'image doit être traité.

2.5 Dépendance, causalité et récursivité

Ce paragraphe discute deux propriétés importantes des signaux unidirectionnels et bidirectionnels, la dépendance entre chaque point et son voisinage et la connaissance d'un signal au cours du temps.

Pour les signaux où le temps est la seule dimension, on observe une dépendance (en particulier une autocorrélation) symétrique vers le passé et vers le futur. La causalité rend compte du fait que le futur est inconnu, et qu'à chaque instant toute estimation d'une valeur ou d'un phénomène futur ne peut résulter que des

connaissances acquises dans le passé (Le passé n'a d'intérêt que si une dépendance existe entre des phénomènes espacés dans le temps).

La récursivité utilise ces propriétés. Elle exprime la dépendance par rapport au passé sous forme différentielle. Par exemple l'équation

$$dy/dt=x(t) \quad (2.7)$$

ou, pour les systèmes échantillonnés:

$$y(k)-y(k-1)=x(k) \quad (2.8)$$

décrit un filtre qui cumule toutes les valeurs passées de x .

Dans les images, la dépendance entre points d'une image n'a pas de direction privilégiée. On lui associe généralement une symétrie circulaire. Au contraire des signaux unidimensionnels, les images s'acquièrent simultanément en chaque élément de leur plan. On ne peut donc, sans autres considérations, y définir causalité et récursivité. Ecrire une équation différentielle en x et en y , c'est arbitrairement sous-entendre une dissymétrie selon certaines directions de l'image.

Les algorithmes parallèles de traitement d'images, qui opèrent simultanément sur tous les points, sont bien adaptés à la structure des images. Cependant, ils coûtent chers et sont peu souples. C'est pourquoi, on retient en général des solutions séquentielles. Des lors, une pseudo-causalité apparaît et les algorithmes récursifs resurgissent. Par exemple, /34/ présente un filtre de Kalman pour la détection des contours, en balayage ligne par ligne.

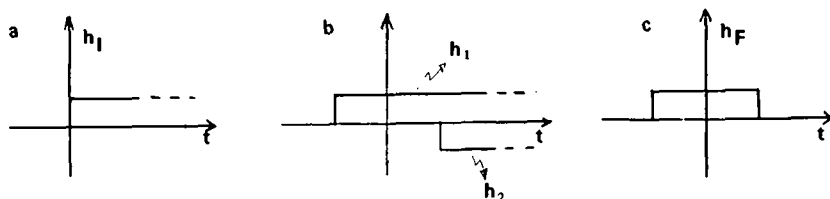


Fig.2.6 Un filtre récursif (a), bien que de réponse impulsionnelle infinie, nécessite peu de calculs. Un couple de filtres récursifs (b) peut remplacer un filtre à réponse impulsionnelle finie (c).

Un couple de filtres récursifs peut également remplacer un filtre à réponse impulsionnelle finie, entraînant des calculs moindres (Fig.2.6). Pour nos applications en robotique, un opérateur de filtrage passe-bas symétrique est synthétisé par deux filtres intégrateurs récursifs (eq. 2.8). En chaque point, une fenêtre de N éléments entraîne N additions par une technique non-récursive, alors qu'il n'y a que deux opérations par la technique récursive (une addition et une soustraction). Dans l'intégration de l'eq. 2.7, la même constante d'intégration apparaît pour les deux filtres utilisés. Elle peut donc se simplifier.

Huang & al./25/ présente aussi une technique récursive pour le filtrage median d'images.

2.6 Forme et Texture

Une image peut se modéliser comme une juxtaposition de zones relativement homogènes séparées par des contours. On associe des caractéristiques de texture aux zones et de forme aux contours. Forme et texture peuvent se définir par analogie: ces deux termes généralisent les notions courantes de silhouette et de couleur respectivement. L'information de forme nous permet de reconnaître tel animal comme étant un chat. L'information de texture nous permet de le différencier d'un autre chat, noir, tigré ou tacheté brun gris.

Dans l'espace, l'information de forme est essentiellement définie perpendiculairement à une surface, alors que la texture y apparaît tangentiellement.

Sur une surface, la différence n'est pas toujours objectivement définissable: Le chat de notre exemple peut se caractériser par une texture particulière, une tache blanche sur l'oreille, mais c'est la forme de cette tache qui, dans un autre contexte va être déterminante. La différence entre forme et texture n'est pas indépendante de l'échelle d'observation.

Notre étude s'est restreinte à des objets de textures très simples: tons de gris uniformes. L'information de forme est ici à la fois essentielle et directement accessible.

Le problème reste donc ouvert d'intégrer à notre approche une étape de détection de texture, conjointe à la détection de forme, pour traiter les images en toute généralité /35/.

3. DETECTION DE CONTOURS

Une image contient de multiples zones relativement homogènes, qui doivent être traitées individuellement. Le problème de la segmentation d'image conduit naturellement à l'emploi des contours.

Les contours sont contenus implicitement dans les images, comme des différences locales de textures. Ils forment un ensemble de points disjoints. Un opérateur de filtrage essentiellement passe-haut (cf. chapitre 2) met en évidence ces points.

D'abord, les points-contours doivent être groupés le long d'un paramètre, pour devenir des courbes. Ces courbes peuvent ensuite être amincies.

3.1 Segmentation d'image

Les images sont souvent complexes et doivent donc être segmentées en régions plus simples. Par exemple le fond de l'image est sans intérêt, ou encore, on veut traiter séquentiellement chacun des objets vus /36/. Trois méthodes sont courantes pour effectuer la segmentation d'une image: L'analyse de connexité, le morcellement par agglomération et division et la poursuite de contour. Parfois, on quitte le domaine spatial et on segmente l'image en classes dans un espace engendré par des caractéristiques locales (chapitre 6). Au retour dans le plan de l'image, une brève application de l'algorithme d'agglomération et de division de régions est encore nécessaire car les points ambigus dans l'espace des caractéristiques peuvent en général être classés grâce à des relations de voisinage dans le plan de l'image.

3.1.1 Analyse de connexité

L'analyse de connexité est la méthode la plus adaptée à la segmentation d'images binaires. Par l'information de présence (ou d'absence) locale du fond, les différentes régions de l'image peuvent être isolées. De petits détecteurs de connexité qui travaillent sur 4 points de l'image simultanément étiquettent séquentiellement chaque

point de l'image (Fig.3.1). A la fin de l'examen de l'image, chaque région, objet ou trou est marquée par des étiquettes qui lui sont propres /37,38,39/. Un arbre peut être construit simultanément, pour indiquer la hiérarchie des régions, c'est à dire leur ordre d'appartenance mutuelle. Cette méthode est particulièrement bien adaptée à une réalisation par circuits spécialisés, car elle travaille ligne par ligne. Elle ne peut toutefois pas être utilisée pour segmenter des images à plus de deux niveaux de gris car la notion de connexité qui lui est essentielle y est moins bien définie.

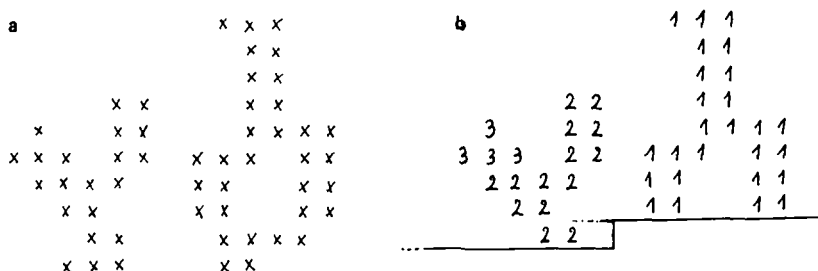


Fig.3.1 Analyse de connexité. En un balayage, chaque point de l'image est attribué à une région connexe. Simultanément, à chaque région sont associées différentes mesures.

3.1.2 Morcellement par agglomération et division

Le morcellement par agglomération et division est une méthode courante pour la segmentation d'images avec textures.

L'image est d'abord divisée en régions de manière arbitraire. Ensuite, les régions voisines dont les caractéristiques sont semblables détruisent leur frontière commune, alors que les régions non uniformes se subdivisent. Le processus est itératif.

La méthode est coûteuse en calcul. Par nature, elle produit des régions uniformes, fermées, ce qui conduit parfois à de fausses frontières.

3.1.3 Poursuite de contours

Appliquée à des images binaires, la poursuite de contours conduit aux mêmes résultats que l'analyse de connexité. Elle y a toutefois un désavantage; l'ordre des points qui subissent un test de connexité varie avec la région de la scène qui est traitée. L'adressage aléatoire de l'image exige donc en pratique une mémoire annexe.

Appliquée à des images à niveaux de gris, la poursuite de contour est par contre très intéressante.

Dans le cas général d'une image à niveaux de gris multiples, on ne peut jamais garantir que tous les bords d'une région soient clairement démarqués. Un objet brillant (par exemple une lame de rasoir, Fig.3.2) posé sur un fond gris uniforme présente simultanément, à cause de l'éclairage et des réflexions, des zones plus claires que le fond, et d'autres plus foncées. Il n'est pas possible, dans ces conditions, de délimiter les régions sans une connaissance a priori de la scène, et les deux méthodes d'analyse de connexité et de morcellement adaptatif dans ces conditions échouent. Elles définissent des régions uniformes, même au prix de contours arbitraires. La poursuite des contours au contraire se fige en "non-décision" lorsque le contraste est inexistant (On peut toujours, si on exige des régions fermées, relier les deux extrémités des contours ouverts par une droite, ou par une courbe arbitraire).

Comme Attneave /40/ l'a montré, l'information de forme est essentiellement contenue dans les contours et donc, en traitant un contour, même partiellement, on retire une information locale et dans ce sens on segmente l'image.

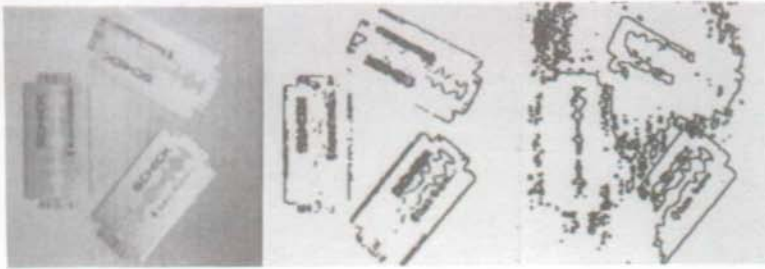


Fig.3.2 L'objet n'a pas une surface uniforme. La poursuite de contour se fige en "non-décision" lorsque le contraste est inexistant. Les méthodes d'analyse de connexité et de morcellement adaptatif génèrent de fausses frontières.

3.2 Poursuite et squelettisation des contours

Les bords sont détectés essentiellement comme une différence en ton de gris entre des régions voisines d'une image. La taille des régions peut varier. On troque de manière optimale, pour une application donnée, de la définition spatiale et de la simplicité de calcul contre une réduction du bruit et des effets d'amincissement.

Comme il est montré au chapitre 2, des opérateurs généraux de squelettisation existent; ils effacent itérativement tous les éléments de contours qui sont superflus pour la connexité. Ces opérateurs traitent l'image indépendamment dans chaque région et se prêtent donc bien à un calcul parallèle par des circuits spécialisés.

Si les calculs se font séquentiellement à cause du processeur disponible, ou si l'utilisation d'information contextuelle est nécessaire, les techniques de poursuite de contours sont préférables. Par exemple, les traitements de filtrage du bruit et d'amincissement des lignes peuvent se limiter à de petites régions de l'image, quasi-simultanément à l'analyse sémantique du bord poursuivi. Ainsi la plus grande partie de l'image peut ne pas être traitée du tout. En traitement séquentiel, il est aussi possible

de sauter selectivement par dessus des ruptures de contours, dans la direction poursuivie auparavant, plutôt qu'uniformément dans n'importe quelle direction. Le lissage de courbes et le codage de contours sont d'autres applications possibles.

3.2.1 Poursuite de bord

Un bon algorithme de poursuite des bords (ou des contours) doit satisfaire 2 contraintes:

- le pointeur qui poursuit les bords doit avancer par pas de taille minimale. Un grand pas augmente le risque de quitter le bord poursuivi, pour sauter sur un bord voisin, et meme si on ne fait pas de telles erreurs grossieres, les imprécisions augmentent avec la longueur du pas /41/.
- le pointeur doit avancer vers le plus proche voisin dont la position forme l'angle le plus petit (ou le plus grand) par rapport aux positions présente et passée du pointeur. Cette contrainte garantit que le pointeur ne pénètre pas a l'interieur de l'objet mais au contraire poursuit toujours le bord.

Suivant l'angle minimum (ou maximum) permis par la seconde contrainte, le pointeur se deplace dans le sens trigonométrique (ou en sens inverse) le long du bord des objets.

16	9	15		
17	6	2	5	14
10	3	<u>A</u>	1	8
18	7	<u>B</u>	4	13
19	11	12		

Fig.3.3 Chaque point est susceptible de devenir la prochaine étape du pointeur dans sa course le long des contours, avec une priorité qui dépend de sa position relative aux étapes précédentes. A et B marquent les arrêts précédents. Si par exemple le voisin "1" appartient au contour, on s'y déplace immédiatement, sans consulter les 18 autres voisins.

La Fig.3.3 montre 19 voisins de deux éléments de bords consécutifs A et B, rangés par ordre décroissant de priorité. Le voisin '12' n'est considéré que si '4' n'appartient pas au contour. Le voisin '19' a une condition similaire sur '7'. Ceci diminue le risque, pour les pointeurs, de se perdre à l'intérieur du contour poursuivi.

Le pointeur a la possibilité, si nécessaire, de contrôler 19 voisins, et par là, de sauter par dessus des trous dans n'importe quelle direction. Pourtant, comme la probabilité de présence des trous est faible, l'avance du pointeur se fait en moyenne avec deux tests par pas seulement.

Ceci est plus rapide, plus précis et moins limité que le "code de chaîne" à huit éléments de Freeman /42/.

3.2.2 Squelettisation de lignes

Deux poursuivants ont été définis. Ils diffèrent par la deuxième contrainte qui force chaque pointeur à se déplacer avec un angle résultant minimal, ou respectivement, maximal.

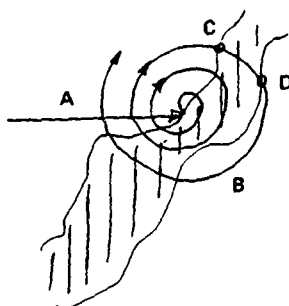


Fig.3.4 Recherche d'une zone d'activité (A). Recherche de points d'entrée et de points de sortie du contours (B). C et D servent de points de départ pour la poursuite duale du contour.

Quand ces deux pointeurs sont placés sur les deux côtés d'un contour épais (Fig.3.4), et qu'ils se déplacent sous une troisième contrainte de distance minimale entre eux, l'axe de la courbe (aussi appelé "ligne mince" ou "squelette") peut s'estimer facilement comme le lieu des moyennes de leurs positions.

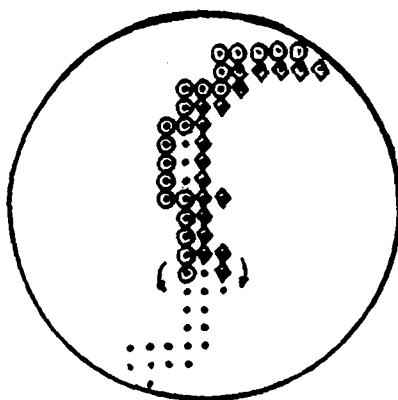


Fig.3.5 Exemple de poursuite d'un contour épais par deux pointeurs. Ils avancent sous une contrainte de distance minimale entre eux. Leur moyenne donne le "squelette" du contour.

En plus d'une estimation facile des lignes minces, cette technique permet une détection rapide des croisements de contours, des courbes fermées et un traitement selectif des trous qui peuvent maintenant être comblés dans la direction du squelette. La Fig.3.5 montre les pointeurs complémentaires, qui travaillent le long d'un contour épais.

Un point ne peut pas absolument se justifier dans l'estimation du squelette. Le contour "vrai" passe-t-il vraiment par le milieu (moyenne) du contour épais? Ceci peut être vrai lorsque les transitions sont progressives. Le blanc diffuse dans les régions noires (Fig.3.6). Par contre les zones d'ombres portées sont très dissymétriques. Le "vrai" bord se confond alors avec le côté de la zone épaisse correspondant à l'objet le plus haut placé. Celui-ci, suivant son orientation relative, peut aussi bien être "plus blanc" que "plus noir", par rapport à la pièce partiellement cachée. L'analyse de cette situation est assez ardue, et de toute façon ne peut pas se faire localement sur l'image. C'est seulement à une échelle plus élevée que le recouvrement pourrait éventuellement se détecter, conduisant à une squelettisation itérative des contours.

Un test sur l'amplitude du gradient se fait parfois, dans une direction perpendiculaire au squelette. Ceci peut se combiner avec l'algorithme de poursuite duale. Est-ce que ceci conduirait à un squelette plus juste? Selon le modèle de la diffusion, les deux critères sont équivalents.

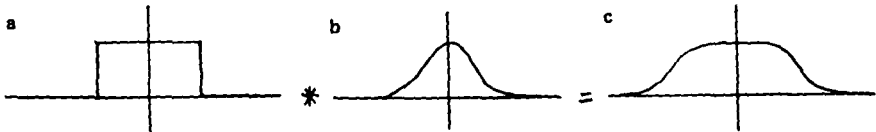


Fig.3.6 Des bords parfaits (a) sont convolués avec une fonction gaussienne de diffusion (b) pour devenir les bords observés (c).

3.2.3 Raffinements possibles de l'algorithme de squelettisation

Le démarrage de la poursuite des contours peut se faire de différentes manières. Ce point n'a pas été approfondi. L'image est parcourue grossièrement jusqu'à ce qu'on rencontre une zone d'activité. On se déplace alors en spirale jusqu'à ce que des points d'entrée et de sortie du contour soient localisés (Fig.3.4), puis le suivi de contour commence. La solution la plus complète consiste à essayer différentes stratégies sur les n points suivants et de retenir celle qui se révèle a posteriori optimale (l'intégrale de la distance entre pointeurs sur les n mouvements est minimale). Ceci ralentit énormément l'avance des pointeurs. La solution la plus simple et la plus rapide consiste à corriger une erreur déjà produite! Lorsque le mouvement d'un pointeur augmente la distance entre les pointeurs, c'est l'autre pointeur qui devient actif, pour essayer de la réduire. Comme le test se fait à chaque pas, l'estimation de distance est affectée par le bruit de quantification, ce qui réduit les performances, surtout la précision, de cette approche lorsque les contours changent de direction.



Fig.3.7 Les deux pointeurs ont un parcours commun avant l'occurrence d'un trou. Chacun peut alors tenter l'interpolation.

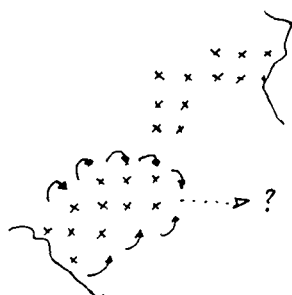


Fig.3.8 Les pointeurs se déplacent en sens opposé à l'approche d'un trou. Il faut alors extrapoler dans la direction du squelette.

Selon que les contours sont localement minces ou épais, deux types d'extrapolation peuvent apparaître, en présence de trous. Le premier cas (Fig.3.7) se distingue par une direction et un sens commun aux deux pointeurs. Le trou peut alors être comblé par chacun des pointeurs indépendamment, pour autant qu'il ne soit pas trop grand. Le second cas (Fig.3.8) se distingue par une direction commune mais un sens opposé des pointeurs. Il convient dans ce cas d'explorer une direction qui leur soit perpendiculaire, ou mieux, une direction définie par les points précédents du squelette.

Ces différents raffinements font l'objet d'une étude plus illustrée dans /43/ (Fig.3.9).

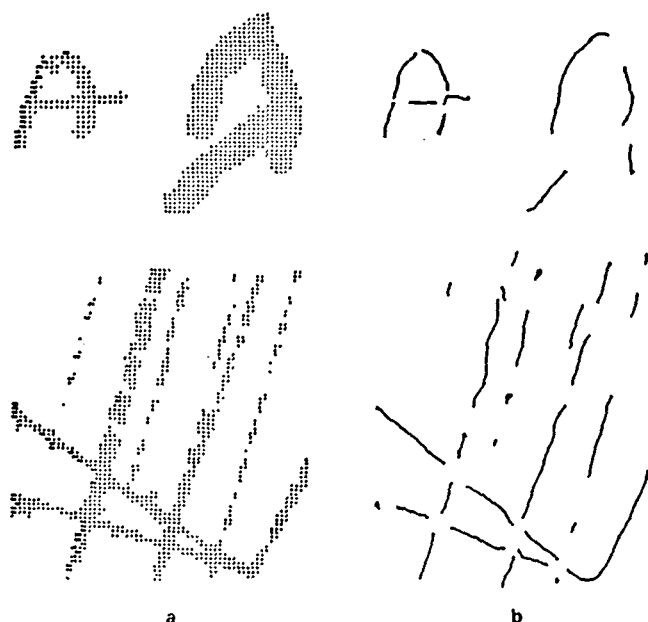


Fig.3.9 Utilisation de l'algorithme de poursuite et de squelettisation pour le traitement de caractères manuscrits et d'images de chambres à bulles. A partir de l'image (a) les contours sont poursuivis, squelettisés, puis filtrés. L'image originale est due à l'obligeance du Prof. C.Y.Suen, de Concordia University, Montréal.

4. ECHANTILLONNAGE ET LISSAGE DES COURBES

Les images contiennent la plus grande partie de leur information dans des bords. Pour cette raison, la plupart des techniques d'analyse d'images commencent par les extraire. Ils deviennent alors des contours plus ou moins connexes et épais, et ultérieurement, des courbes. Comme les courbes n'ont pas d'épaisseur, elles se décrivent bien par des fonctions paramétrées orthogonales. Dans le plan, $x(s)$ et $y(s)$ s'utilisent; et pour la profondeur, dans les images stéréo par exemple, on ajoute $z(s)$.

Quand on numérise une image, du bruit corrompt les courbes. Sous différentes hypothèses, plusieurs approches permettent de diminuer l'effet de ce bruit.

Dans ce chapitre, deux problèmes très liés sont présentés. D'abord on analyse le contenu des images numérisées, ensuite différents lissages de courbes sont présentés. En particulier, le taux approprié de lissage pour une image est déduit de son contenu significatif.

La première section traite du pas d'échantillonnage nécessaire pour une image donnée. En général le théorème de l'échantillonnage résout ce problème. Une approche différente est toutefois possible si l'information se concentre dans les contours. Le pas d'échantillonnage est alors vu comme un intervalle de quantification. Courbes et bruit de quantification sont analysés dans le domaine de Fourier et le choix d'un intervalle de quantification est illustré par un exemple.

La seconde section présente différents algorithmes pour lisser les courbes bruyantes, basés sur l'interpolation, sur l'approximation par splines ou par arcs de cercles, sur l'usage de tables de conversion et de filtres passe-bas. Leurs propriétés respectives sont analysées dans le domaine de Fourier. On déduit une relation entre le contenu significatif d'une image et le filtre passe-bas approprié pour le lissage de ses contours.

Bien que les exemples de contours lissés donnés dans ce chapitre se rapportent à des courbes fermées, toutes les techniques décrites ci-après peuvent s'appliquer à des contours ouverts.

4.1 Intervalle d'échantillonnage minimum pour une image donnée.

Les contours peuvent se représenter de manière équivalente par des différences en ton de gris dans l'image, ou par des fonctions paramétrées. Quand l'information se trouve dans la forme et non dans la texture, l'échantillonnage d'une image correspond à la quantification des courbes paramétrées. Pas d'échantillonnage et intervalle de quantification ont les mêmes unités physiques et conduisent au même ensemble de points-contours.

4.1.1 Courbes paramétrées

Considérons les fonctions x et y d'une courbe paramétrée C :

$$C: (x(t), y(t)) \quad (4.1)$$

En principe, le paramètre t garantit seulement la connexité et décrit la séquence des éléments de courbes. Dans des contextes particuliers, le paramètre peut revêtir une information plus riche. Par exemple pour l'acquisition de données par tablette de numérisation, le paramètre est le temps, et le filtrage de courbes se fait souvent le long du temps dans ces applications particulières, parce que le bruit y apparaît essentiellement le long de l'axe du temps (par ex. /44/).

Dans les images numérisées, la longueur d'arc s paraît être un paramètre adéquat parce que les traitements s'opèrent essentiellement dans le domaine spatial. Même si une séquence est artificiellement définie durant l'analyse, elle dérive de propriétés spatiales, telles que des distances entre voisins ou des angles de direction. Il faut toutefois noter que la longueur d'arc utilisée pour paramétrer une courbe peut en fait dériver d'une autre courbe (cf. $x_q(s)$ dans la Fig.4.2). De même, les distances entre éléments de courbes peuvent être différentes de l'intervalle correspondant sur le paramètre. Par exemple, les distances précises sont presque toujours calculées le long de courbes qui sont paramétrées le long d'une estimation grossière de

la longueur d'arc.

4.1.2 Pas d'échantillonnage et intervalle de quantification.

Le chapitre 2 a rappelé le théorème de l'échantillonnage appliqué aux images. Il y a été dit qu'une grille d'échantillonnage rectangulaire (cartésienne) est retenue pour notre étude.

$$b(n_x, n_y) = b_a(n_x \Delta_x, n_y \Delta_y) \quad (4.2)$$

où Δ_x et Δ_y désigne les pas d'échantillonnage en x et en y respectivement. Le problème, au moment de numériser une image est donc de choisir Δ_x et Δ_y de manière à pouvoir ensuite traiter et recréer l'image.

Dans une image numérisée, un contour C apparaît comme un ensemble d'éléments disjoints, où le ton de gris (l'intensité lumineuse) varie:

$$C: \{(n_i \Delta_x, n_j \Delta_y)\} \quad (4.3)$$

pour i, j tels qu'un critère sur la variation de $b(n_x, n_y)$ est vérifié.

Lorsqu'un contour est paramétré, et que ses fonctions $x(s)$ et $y(s)$ sont quantifiées, elles conduisent aussi à un ensemble de points disjoints. Cependant la différence est que le paramètre garantit encore la connexité qui est essentielle aux courbes.

Nous nous intéressons aux courbes parce qu'il est nécessaire d'établir des relations de voisinage le long de l'abscisse curviligne pour y étouffer la sensibilité au bruit de numérisation /45/. Une alternative au problème de choisir un pas d'échantillonnage consiste alors à choisir un intervalle de quantification pour des courbes paramétrées. Les rapports signal-sur-bruit (SNR), qui prennent en compte les propriétés statistiques du bruit de quantification et

des fonctions paramétrées peuvent servir ici, comme pour les signaux unidimensionnels. /46/ fait référence a ce sujet.

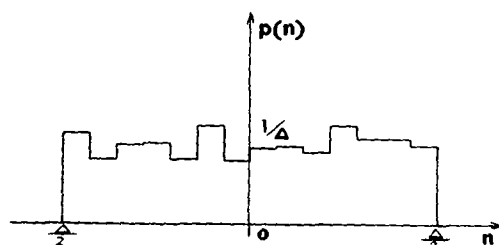


Fig.4.1 Le bruit de quantification, mesuré ici comme la différence entre les courbes a et b de la Fig.4.4, a une densité de probabilité uniforme

Dans une image, les contours peuvent apparaître n'importe où. Un quantificateur uniforme est donc approprié. Le bruit de quantification présente une densité de probabilité uniforme dans l'intervalle de quantification (Fig.4.1). Il est purement granulaire car les contours qui sortent du champ de vue ne sont pas interprétés. Le bruit est généralement aléatoire; il a une moyenne nulle et sa puissance est proportionnelle au carré de l'intervalle de quantification ($\Delta^2/12$). Puisque le quantificateur peut, dans le cas le plus défavorable, introduire une erreur de $\Delta/2$ en chaque point, la limite supérieure de la puissance du bruit vaut:

$$P_{nx} = \Delta^2 x / 4 \text{ et } P_{ny} = \Delta^2 y / 4 \quad (4.4)$$

Pour une courbe de puissance P_x, P_y selon chaque axe respectivement, le SNR r vaut:

$$r_x = P_x / P_{nx} \text{ et } r_y = P_y / P_{ny} \quad (4.5)$$

Le long de l'axe x par exemple, il résulte que si un SNR r est requis pour une courbe donnée de puissance P_x , l'intervalle de quantification suivant devrait être choisi:

$$\Delta_x = 2 \sqrt{P_x / r} \quad (4.6)$$

Il faut encore remarquer que le quantificateur est appliqué ici à des signaux continus. Le paramètre n'est pas échantillonné et appartient à un continuum (cf. Fig.4.2). La sortie du quantificateur est une fonction particulière continue par morceaux; une fonction à accroissements bornés /47/. A cause de sa nature quantifiée, une courbe paramétrée (continue par rapport à s) apparaît comme un ensemble de points disjoints dans les images numérisées.

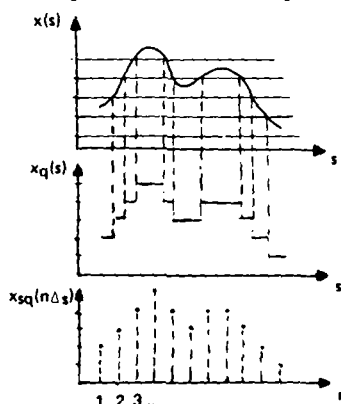


Fig.4.2 $x(s)$ peut être quantifié sans échantillonnage. Il en résulte $x_q(s)$ (cas des courbes dans les images numérisées). En général pour les signaux unidimensionnels, la quantification est précédée d'un échantillonnage ($x_{sq}(n\Delta_s)$).

Une alternative à l'utilisation du théorème de l'échantillonnage consiste donc à déduire l'intervalle de quantification des courbes d'un critère de rapport signal-sur-bruit choisi a priori.

4.1.3 Contenu spectral des courbes et du bruit de quantification

Avant de comparer les différentes stratégies de lissage des contours, il convient de décrire le spectre des courbes et du bruit. Les fonctions $x(s)$ et $y(s)$ d'une courbe paramétrée s'expriment le long d'une abscisse curviligne s .

Leur transformée de Fourier aura donc comme unités un nombre de cycles par mètre plutôt que par seconde comme dans le domaine de Fourier habituel (la fréquence spatiale s'exprime aussi parfois, surtout pour les images, en nombre de cycles par millimètre).

Il se trouve une contrainte que toutes les courbes satisfont, dans le plan:

$$ds = \sqrt{dx^2 + dy^2} \quad (4.7)$$

où dx , dy et ds sont des incréments infinitésimaux le long de x , y et s .

L'analyse spectrale de $x(s)$ et de $y(s)$ révèle une forte dépendance entre amplitude et fréquence. Tout changement en x et en y , positif ou négatif, accroît la longueur d'arc. À cause de cette propriété, une composante de fréquence élevée (taux de changement rapide) ne peut pas avoir une grande amplitude. D'autre part, une composante de basse fréquence ne peut être que grande. Ceci est illustré par la Fig.4.4 et sera justifié quantitativement, plus loin, pour les cercles. La limitation (4.7) ne tient pas cependant lorsqu'une courbe n'est pas paramétrée le long de sa propre longueur d'arc. Par exemple dans le cas théorique où le vrai contour est connu a priori, l'abscisse curviligne exacte peut s'utiliser comme paramètre commun pour la description du contour correspondant, à travers différents traitements. Pour la Fig.4.4 chaque courbe est paramétrée le long de la même abscisse curviligne, qui est dérivée de l'estimation la plus fine.



Fig.4.3 Exemple de pièce industrielle

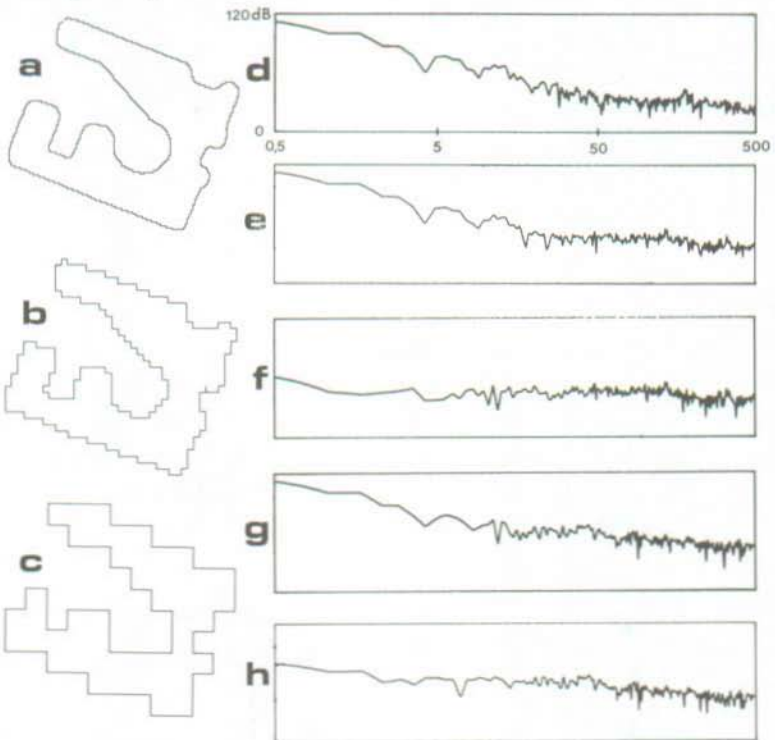


Fig.4.4 Spectre de courbes paramétrées. Les courbes d,e,g correspondent aux contours a,b,c tirés de la Fig.4.3. Les courbes f et g illustrent le bruit de quantification, défini comme la différence entre b et a pour la première, et entre c et a pour la seconde. Seule la fonction selon l'axe x est utilisée.

La courbure est une propriété importante, étroitement liée à l'énergie de flexion. Elle est définie en chaque point le long des contours comme l'inverse du rayon du cercle osculateur (le cercle qui assure le meilleur contact avec la courbe). Le cercle est donc un cas particulier intéressant. Il est paramétré par des sinusoides en x et en y . Son amplitude est bornée par $1/f$ dans le domaine spectral: Un cercle de rayon R est de fréquence spatiale $1/R$. Il est intéressant de noter que la plus haute fréquence pour $x(s)$ et $y(s)$ apparaît là où le cercle osculateur est le plus petit, ou, de manière équivalente, là où la courbure est maximale.

Le contenu spectral du bruit de quantification dépend du choix du paramètre. Si le paramètre est choisi comme étant la longueur d'arc après quantification, la même dépendance apparaît pour le bruit que pour la courbe. Il est plus adéquat d'analyser le bruit de quantification le long du contour non-quantifié, parce que le bruit de quantification corrompt indépendamment chacune des fonctions $x(s)$ et $y(s)$. Il peut alors être modélisé très simplement comme un bruit additionnel $n_x(s)$ et $n_y(s)$ (Fig.4.5). Bien que le contour non-quantifié soit rarement disponible comme paramètre en pratique, il est utile ici pour des raisons théoriques.

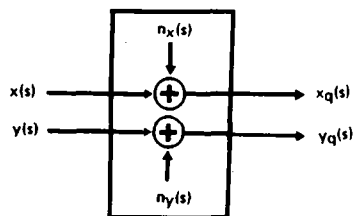


Fig.4.5 Modèle de quantificateur. $x_q(s)$ est la somme de la fonction $x(s)$ décrivant la courbe originale et d'un bruit aléatoire $n_q(s)$ à distribution uniforme. Il en va de même selon y .

Sous l'hypothèse que les contours sont indépendants de la grille d'échantillonnage le bruit est blanc (ceci n'est pas

toujours vrai, surtout pour les objets manufactures). Sa fonction d'autocorrélation présente un pic à l'origine. Comme les courbes ont toujours une longueur limitée, le bruit n'est jamais absolument aléatoire. Son spectre n'est donc pas à largeur de bande infinie, et en particulier, sa puissance n'est pas nulle dans la bande significative des contours. Si le bruit n'est pas aléatoire, il se concentre dans certaines régions du spectre. Sa borne supérieure est plus élevée, mais elle est aussi constante.

Il résulte des considérations précédentes que le rapport signal-sur-bruit des courbes est relativement petit pour les composantes de fréquence relativement élevées (grande courbure)(Fig.4.4). Ceci conduit au choix d'un intervalle de quantification souvent plus petit que le pas exigé par le théorème de l'échantillonnage.

Bennet /48/ en particulier a publié une étude très fouillée du spectre du bruit de quantification.

4.1.4 Exemple commenté

Un exemple pratique va montrer comment le pas d'échantillonnage et l'intervalle de quantification se déduisent soit du théorème de Shannon, soit d'une contrainte exprimée en rapport signal-sur-bruit.

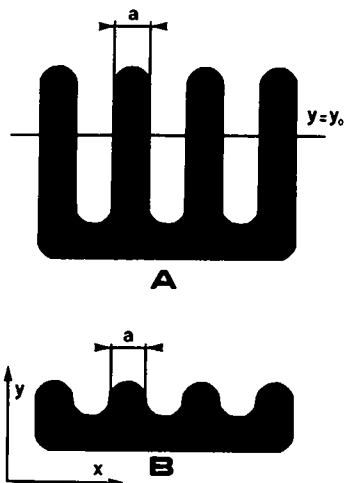


Fig.4.6 Exemple d'image à numériser (voir texte)

Examinons un objet dans un champ de vue comme l'objet A de la Fig.4.6. L'image $b_a(x,y)$ ne prend que deux valeurs, blanc et noir. La fréquence la plus élevée apparaît le long d'une ligne parallèle à l'axe x ($y=y_0$), coupant les pointes de l'objet. La fonction monodimensionnelle qui en résulte $b_a(x,y_0)$ est une onde carrée le long de x . Conformément au taux de recouvrement des spectres qui est toléré, la $(2n+1)^{\text{ième}}$ harmonique peut être considérée comme étant la composante de fréquence maximale. Choisissons par exemple $n=0$ pour la suite. La période est alors égale à $2a$ (cf. Fig.4.6), et le pas d'échantillonnage qui en résulte Δx ne devrait pas excéder a le long de x pour un tel objet.

L'objet peut avoir n'importe quelle position par rapport au système de coordonnées. La même définition est donc requise le long des deux axes:

$$p = \Delta x = \Delta y \quad (4.8)$$

De plus, comme nous avons une grille cartésienne, un supplément de $\sqrt{2}$ en définition le long de x et de y doit être accordé de manière à pouvoir traiter les situations

possibles ou l'objet fait un angle multiple de 45 degrés par rapport aux axes x et y . Un pas d'échantillonnage de taille $a/\sqrt{2}$ évitera le recouvrement.

Examinons le SNR du contour. Pour l'objet A, les pointes seront clairement détectées parce qu'elles sont principalement composées de long segments droits qui sont des contributions du contour à fréquence nulle selon x et y . Il est clair cependant qu'avec la valeur choisie pour p , peu d'information peut être donnée quant à leurs extrémités; sont-elles carrées, triangulaires, rondes? Considérons l'objet B. Pour éviter des erreurs de recouvrement, le même pas d'échantillonnage que précédemment serait choisi, par le même raisonnement. Cependant il faut remarquer que dans le cas le plus défavorable, le SNR en haute fréquence est maintenant proche de 6 dB après quantification. L'amplitude maximale du bruit est d'environ $p/2$, alors que la composante de fréquence maximale selon x et selon y a une amplitude de p .

Si on veut par exemple 20 dB de SNR pour les pointes des objets A et B (y compris les extrémités), en utilisant les équations (4.5) et (4.6), le pas d'échantillonnage -l'intervalle de quantification- devrait être choisi 5 fois plus petit:

$$p = a / (5\sqrt{2}) \quad (4.9)$$

Il faut encore remarquer que le recouvrement possible dû à l'échantillonnage affecte $b(n_x, n_y)$, alors que le bruit de quantification affecte $x(s)$ et $y(s)$ directement.

4.2 Lissage de courbes bruyantes

Pour pouvoir traiter numériquement les courbes, il est encore nécessaire d'échantillonner les fonctions paramétrées. Qu'une courbe doive être échantillonnée à intervalles réguliers ou non, le premier problème à résoudre est l'estimation de longueur d'arc. La plupart des caractéristiques extraites des courbes dépendent d'une

manière ou d'une autre de la longueur d'arc. Ceci est évident pour le périmètre et pour la courbure, mais c'est parfois plus subtil comme pour les moments d'inertie où des incréments d'arcs locaux pèseraient les coordonnées du contour.

D'habitude, la longueur d'arc n'est connue que par les fonctions quantifiées. Pour le plus petit changement de ces fonctions après quantification, c'est-à-dire lorsqu'elles varient d'un niveau élémentaire p , l'incrément de longueur d'arc non-quantifié se trouve dans le domaine $(0, \sqrt{2} p]$. Il a donc une incertitude de $1/\sqrt{2} p$. il n'est possible d'améliorer les estimations -de diminuer cette incertitude- qu'en examinant le voisinage. C'est un phénomène remarquable, discuté par Steinhaus /49/ et par Mandelbrot /50/ que la longueur d'arc augmente généralement exponentiellement lorsque l'échelle d'observation se fait plus fine: Plus est grande la finesse avec laquelle on examine un lieu réel, plus longue est la longueur d'arc et plus denses sont les extrema de la courbure.

La numérisation d'image induit un bruit de quantification sur les courbes. Les fonctions de la courbe ont alors des discontinuités à chaque changement de niveau (Fig.4.2). Le théorème d'échantillonnage appliqué aux courbes requiert en particulier, que les fonctions soient bornées dans le domaine de Fourier. De plus il résulte de l'expérience et des considérations du chapitre 3 que les composantes de basse fréquence sont prépondérantes dans le contenu significatif des courbes, surtout après quantification. Ainsi quelque sorte de lissage est nécessaire ("lissage" désigne au sens large n'importe quel type de filtrage passe-bas).

Des algorithmes courants de lissage de courbes sont présentés ci-dessous.

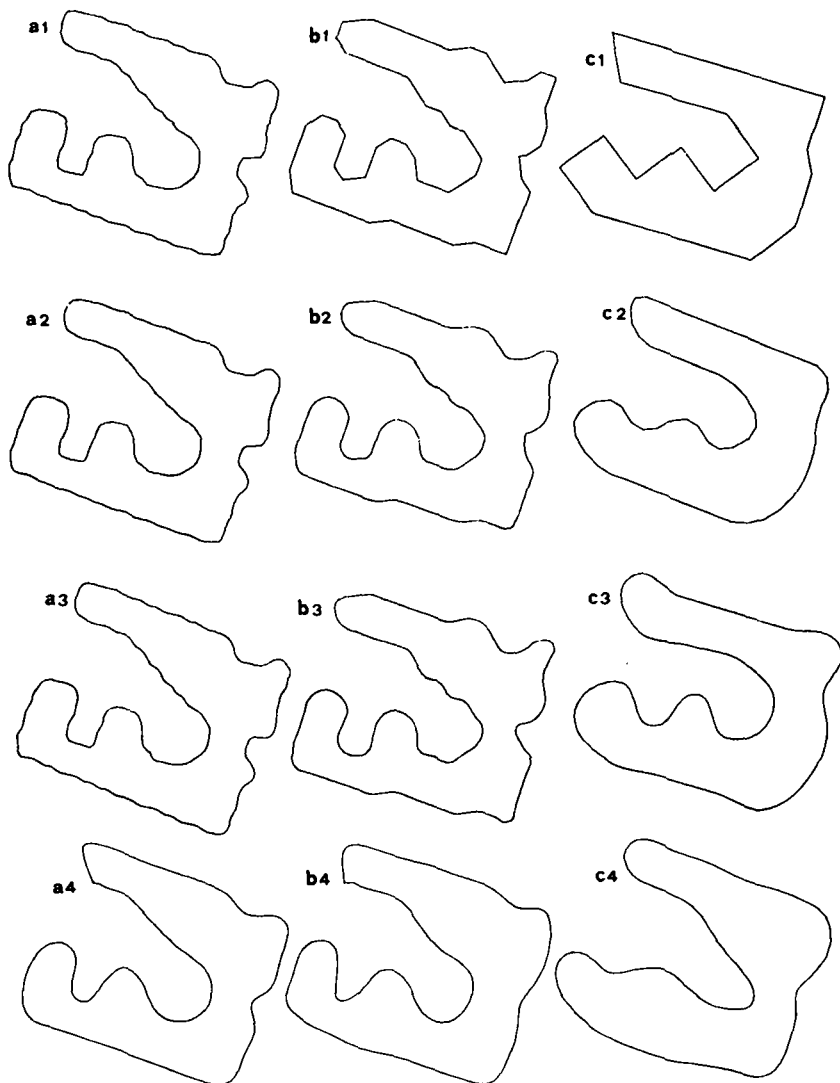


Fig.4.7 Les courbes de la Fig.4.4 sont rééchantillonnées à 1 point sur 5 (a) et 1 point sur 3 (pour b et c). Les courbes sont alors lissées par différentes méthodes:

1. Interpolation linéaire
2. Interpolation quadratique
3. Interpolation d'ordre 3 (/53/)
4. Approximation par fonctions splines. 40 noeuds sont employés, sans régression entre eux.

4.2.1 Interpolation entre quelques points des courbes

Une méthode simple et souvent suffisante de lisser les courbes quantifiées consiste à sous-échantillonner les courbes, négligeant des éléments de la courbe par intervalles, et ensuite à reconstruire par interpolation entre les points restants une approximation des courbes (Fig.4.7.1/2). L'interpolation peut être linéaire, quadratique, cubique ou même d'ordre plus élevé.

Le lissage résulte de l'interpolation, dont la fonction de transfert de type passe-bas est un $(\sin f/f)^{n+1}$ qui a son premier zéro pour $f=1/p$. La valeur n est l'ordre d'interpolation (1 pour linéaire, 2 pour quadratique...), et p le pas d'échantillonnage.

Comme discuté précédemment, les composantes de basses fréquences ont toujours une assez forte puissance; et pour cette raison, le recouvrement du spectre dû au sous-échantillonnage n'est, bien souvent, pas trop gênant. Young /51/ a même prouvé, pour les courbes, que $1/n^5$ est une borne supérieure pour la puissance relative des harmoniques. L'interpolation qui s'opère après l'échantillonnage retient la bande de base de la courbe.

Deux méthodes paraissent ici particulièrement intéressantes. L'une, publiée par Pavlidis /52/ confond les étapes d'échantillonnage et d'interpolation. De nouveaux échantillons sont choisis itérativement de manière que l'erreur globale d'une approximation linéaire -spline du premier ordre- soit minimale. Dans une première phase, un algorithme de division-agglomération place grossièrement des "noeuds" (nouveaux échantillons) le long des contours. Ensuite, l'algorithme de Newton permet l'optimisation de leur position. La deuxième méthode est une extension due à Midgley /53/ d'un algorithme de suppression de discontinuités publié par Rutkowski /54/. Des calculs relativement simples interpolent les données à l'aide d'un polynôme du troisième degré dans chaque intervalle

(Fig.4.7.3). Deux points de part et d'autre de l'intervalle définissent les coefficients du polynôme utilisé.

4.2.2 Approximations par splines

Quelquefois, très peu de noeuds sont choisis le long des courbes, à cause du taux de lissage ou de réduction de données désirés, et pour limiter la complexité des calculs (par exemple, dans /55/ environ un point sur 10 sert de noeud). La courbe est ensuite approximée entre les noeuds par des fonctions simples, habituellement polynômiales (splines)(Fig.4.7.4).

Le choix des noeuds (ou les splines s'arrêtent) est délicat; la description y est sensible. Pour réduire cette difficulté, à la fin de chaque segment ce sont les dérivées qui servent de contraintes à l'approximation, au lieu des valeurs absolues. En conséquence, l'approximation peut ne plus passer par les noeuds mais les transitions entre segments sont maintenant très douces. Les meilleurs résultats sont obtenus lorsque des régressions se font à l'intérieur des intervalles entre les noeuds (voir /55/ et les références qui s'y trouvent), c'est-à-dire lorsque toute l'information disponible est utilisée. Une approximation est différente d'une interpolation en cela que la courbe lisse qui en résulte peut ne pas contenir les points originaux du contour.

Le lissage obtenu par l'approximation de courbes par des splines est essentiellement le même que celui du paragraphe précédent. Les courbes sont échantillonnées grossièrement, puis une interpolation s'opère (généralement avec un polynôme de degré inférieur à 6), traitant par le fait même les courbes par un filtre passe-bas. Cependant, comme peu d'échantillons sont retenus, l'expérience montre que le recouvrement ne peut plus être ignoré et en pratique des contraintes spatiales - par exemple la plus petite erreur au sens des moindres carrés- sont introduites afin de se rapprocher du contenu basse-fréquence de la courbe

originale, plutôt que de simplement garder la bande de base avec ses distorsions.

Comme le pas d'échantillonnage varie et que les échantillons sont tirés des dérivées plutôt que des coordonnées originales, le théorème de l'échantillonnage doit être généralisé pour nous dire la largeur de bande exacte de telles approximations. Jerri /56/ a publié un article de synthèse très complet des extensions apportées au théorème de Shannon. Il en ressort que, grosso modo, la largeur de bande $f_{\max}$ est le nombre de mesures ponctuelles (valeurs absolues ou dérivées) rapporté à la durée d'observation.

Les splines permettent une description mathématique, continue des contours, et des calculs locaux en tout point. L'effet de lissage peut être extrême. A cause de sa complexité, la méthode n'est recommandée que pour le lissage de courbes quantifiées très grossièrement. Il est un peu artificiel de négliger la plus grande partie des données dans un premier temps, pour ensuite recréer toute la courbe. Par contre, l'approximation par splines est très bien adaptée aux applications où les données manquent pour des raisons extérieures (par exemple mesures manuelles, quantification grossière, statistiques...). Pour utiliser des polynômes de degré supérieur à 2, une erreur globale mesurée le long des courbes doit être minimisée. La méthode -itérative- est assez complexe. Bookstein /55/ a publié un article définitif sur l'approximation de courbes par des coniques -splines avec polynômes de degré 2. Cette publication donne une technique linéaire pour faire passer des coniques par des points, avec invariance aux rotations, aux translations et aux changements d'échelle.

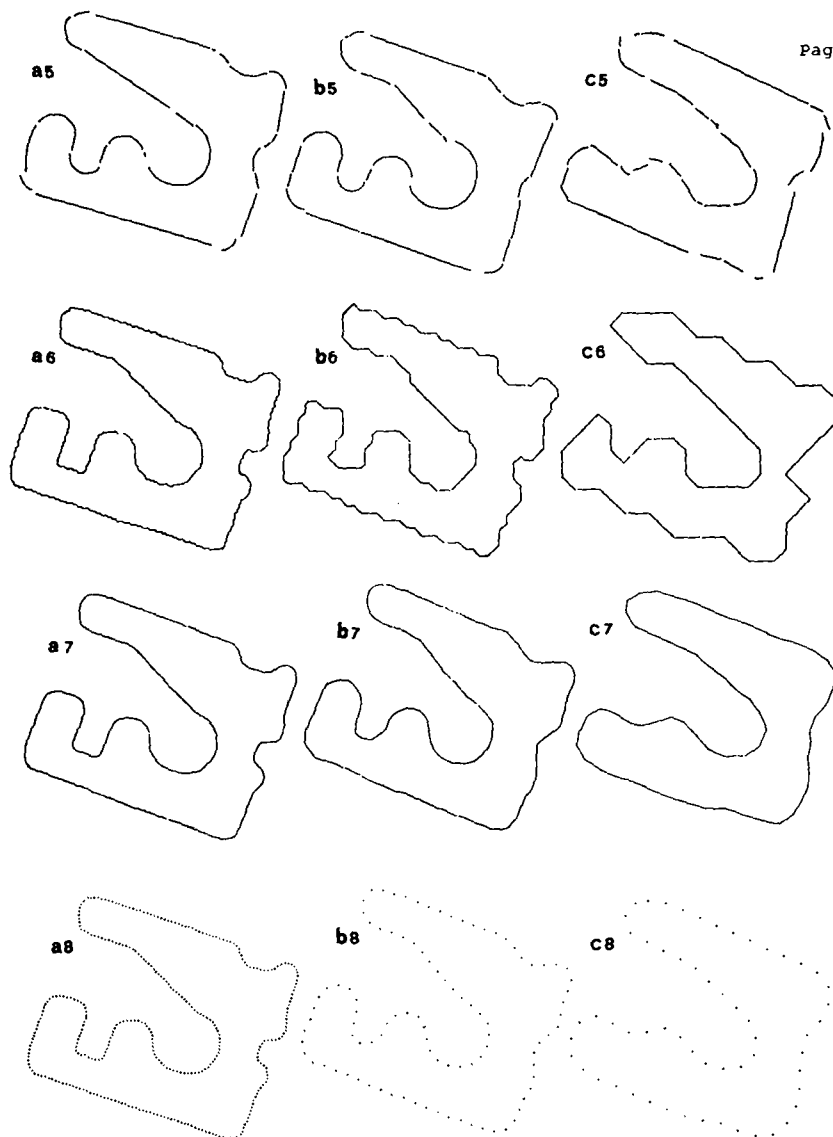


Fig.4.8 Lissage des courbes de la Fig.4.4.

5. Approximation par segments de droite et de cercles, en travaillant dans le plan courbure-longueur d'arc. La courbure est estimée d'une courbe préalablement lissée comme en 7.
6. Lissage de contour par tabulation /57/. Un filtrage sur deux unités de longueur est implicite.
7. Lissage curviligne itératif /58/. On choisit de perdre une partie du signal (coins arrondis) pour demeurer insensible au bruit.
8. Après filtrage comme en 7, les courbes sont échantillonnées

4.2.3 Approximation de courbes par segments de cercles

L'approximation de courbes par des segments de cercles est un cas particulier de l'approximation par des coniques mentionnée au paragraphe précédent. Pourtant une attention particulière doit être accordée ici aux algorithmes qui travaillent dans le plan courbure-abscisse curviligne ou pente-longueur d'arc plutôt que dans le plan x-y de l'image. D'une part, pentes et courbures sont fortement affectées par le bruit de quantification des contours, et ils ont particulièrement montré la nécessité d'un lissage des courbes. Mais d'autre part, elles permettent une description immédiate des cercles. Les cercles ont une pente qui varie linéairement avec la longueur d'arc; leur courbure est constante.

Approximer des courbes par des cercles correspond à l'approximation de fonctions mono-dimensionnelles -dans notre cas $x(s)$ et $y(s)$ - par des segments de sinusoides.

Barrow et Popplestone /6/ ont approximé des courbes par des arcs de cercles en ajustant des segments de droite dans le plan tangente-longueur d'arc. Perkins /7/ a étendu ceci à la description structurale de pièces mécaniques. La méthode est intéressante pour représenter des objets simples faits artificiellement, et des résultats expérimentaux ont prouvé sa validité. Il faut toutefois remarquer que les angles sont extrêmement sensibles au bruit et qu'un lissage préalable des contours, des angles, ou des courbures est toujours requis. Les illustrations de la Fig.4.8.5 résultent de l'estimation de la courbure le long de courbes lissées et de l'approximation de cette courbure par des segments de fonctions constantes.

4.2.4 Amélioration des courbes à l'aide de tables

Dans beaucoup d'applications, les courbes sont considérées élément par élément. Chaque élément de contour considère dans son voisinage définit une configuration. Diverses

configurations, qu'on sait très bruyantes, sont remplacées par des configurations plus lisses mais topologiquement équivalentes (Fig.4.8.6).

Cette méthode marche très bien et très vite (par exemple /57/,/8/). Elle est particulièrement appropriée lorsque les courbes sont décrites par le code de chaîne de Freeman. Cependant comme le nombre de configurations possibles croît très vite lorsque le voisinage étudié s'élargit, on se limite en pratique à trois éléments.

La méthode apparaît souvent comme un cas particulier de filtrage passe-bas (traité plus loin). Par tabulation les données sont remplacées une à une par la sortie correspondante d'un filtre. A cause du nombre de configurations possibles (habituellement petit), des caractéristiques locales telles que la pente ou l'incrément différentiel de longueur d'arc peuvent aussi être tabulées plutôt que de se recalculer sur les données lissées, à chaque occurrence.

Par exemple, Kammenos /57/ convolue une réponse impulsionnelle carrée de largeur 2 p avec les courbes, et tabule les incréments de longueur d'arc. Bowie /8/ convolue une fenêtre de largeur 3 p avec les courbes et arrondit le résultat.

4.2.5 Filtrage des courbes

Les méthodes présentées ci-dessus supposent que le contenu significatif des courbes est prédominant en basse fréquence. Cette hypothèse est validée par leurs résultats expérimentaux et par l'équation (4.7). Ainsi, le traitement le plus immédiat consiste en fait à filtrer passe-bas les courbes, c'est-à-dire leurs fonctions paramétrées, après numérisation d'image mais avant tout échantillonnage de la courbe.

La fréquence de coupure f_c du filtre se trouve par des considérations complémentaires à celles qui conduisent au choix de p au paragraphe 4.1. Si la fréquence maximale des courbes est connue, on peut la choisir comme fréquence de coupure. Lorsqu'on exige un SNR r , le cas le plus défavorable doit être considéré. Comme discuté en 4.2.3, le bruit a une borne supérieure constante sur tout le spectre; il peut se concentrer dans n'importe quelles régions du spectre lorsqu'il n'est pas indépendant des courbes. D'autre part, en haute fréquence, la puissance des courbes est relativement petite. Dans le cas le plus défavorable, la courbe est un cercle à la fréquence de coupure (la fréquence la plus haute considérée). Il résulte alors de l'équation (4.6) que la fréquence de coupure doit valoir ceci:

$$f_c = 1/(2\pi p\sqrt{2r}) \quad (4.10)$$

Si la courbe a toutes ses composantes spectrales au-dessus de f_c , r ne peut être atteint, et la courbe doit disparaître. Si toutes les composantes se trouvent au-dessous de f_c , le bruit s'atténue et ainsi, la courbe est améliorée (Fig.4.9).

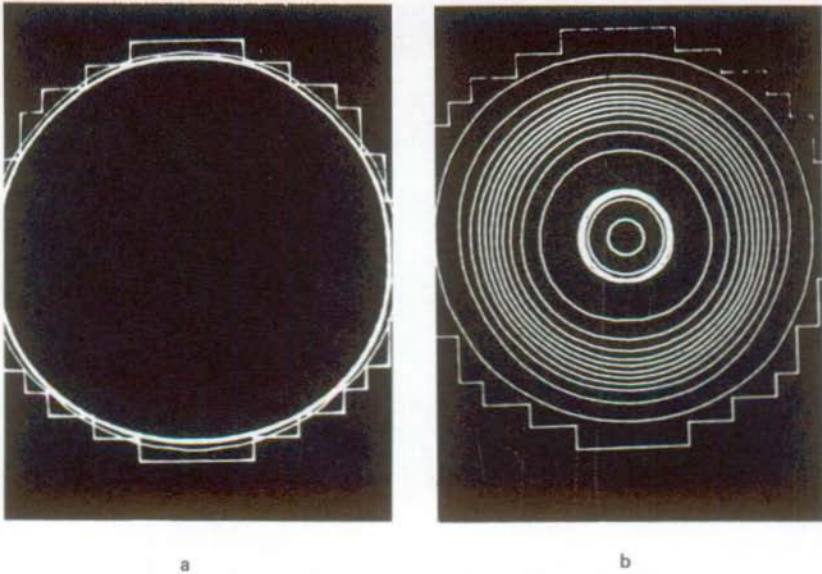


Fig.4.9 Un cercle de rayon R converge vers une limite où il est légèrement atténué, avec un filtre de durée R également (a). Avec une durée $1.8R$, une limite stable ne peut plus être atteinte. En pratique, les courbes sont toujours grandes par rapport à la réponse impulsionnelle du filtre et il y a donc une solution stable.

En pratique, les courbes ont généralement des composantes qui sont tout à la fois au-dessus et au dessous de f_c . Le filtrage ne retient alors que les composantes susceptibles de satisfaire le SNR requis. La courbe lissée qui en résulte est alors différente de la courbe originale. Cependant, elle accuse maintenant une sensibilité réduite au bruit de quantification. En traitement d'image, les caractéristiques extraites de cette courbe lissée sont aussi insensibles (dans les limites du SNR choisi) à la position et à l'orientation relative de la grille d'échantillonnage ($/58/$). A cause des composantes de hautes fréquences qui ont été coupées, les caractéristiques extraites ont un biais. Dans ce sens seulement, la précision est garantie. La Fig.4.10 présente des courbes douces résultant du filtrage d'un carré, numérisé selon deux orientations différentes.

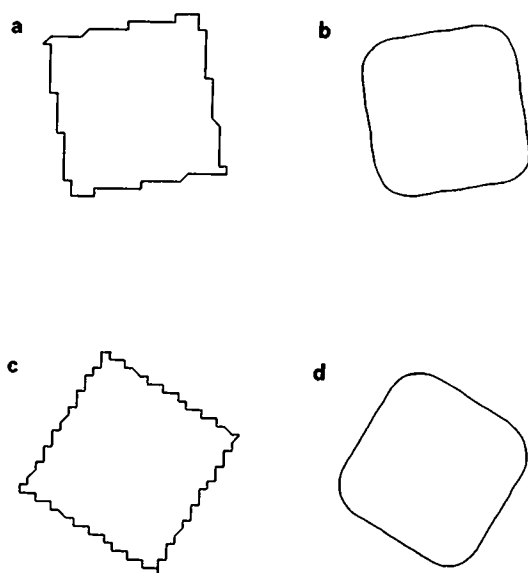


Fig.4.10 Un carré tourne de 10° (a) et de -30° (c) devant une grille de quantification. Un filtre curviligne itératif restitue le carré original, moins sensible au bruit de quantification, mais privé des hautes fréquences (b,d).

Lorsque la fréquence de coupure est choisie, le filtre peut être conçu. Le compromis entre la simplicité et l'efficacité varie avec l'application. Par exemple, le filtre moyenneur est très simple; il est adéquat pour le filtrage continu par ordinateur (par exemple /58/). L'utilisation de transformées de Fourier fournit un autre exemple de réalisation courante de filtrage sur ordinateur. La transformée du signal est évaluée, multipliée par une fonction de transfert passe-bas, et sa transformée inverse est calculée (par exemple /59/). Il faut faire attention dans ce dernier cas, au comportement du filtre entre les échantillons de sa fonction de transfert.

Le filtre conçu a des caractéristiques fixes. Sa fonction de transfert ne change pas, quelle que soit la valeur du paramètre. Par analogie avec les filtres courants, qui

opèrent le long du temps, on dira que le filtre conçu est stationnaire. Pourtant, lorsqu'il est utilisé le long de courbes bruitées, il est équivalent à un filtre non-stationnaire opérant le long de la longueur d'arc originale. Par exemple, si une courbe s'allonge par addition de bruit, la réponse impulsionnelle apparente du filtre le long du contour original semble proportionnellement plus courte. Comme l'abscisse curviligne augmente statistiquement lorsqu'un bruit indépendant à moyenne nulle s'ajoute à la courbe (annexe 3), la fréquence de coupure du filtre non-stationnaire est en général plus élevée que celle du filtre stationnaire. Cependant dans des cas particuliers, surtout si le bruit n'est pas indépendant du contour, elle peut atteindre sa limite inférieure:

$$f_c' = f_c / \sqrt{2} \quad (4.11)$$

parce que pour chaque incrément de l'abscisse curviligne l'estimation de longueur se cantonne dans le domaine $(0, \sqrt{2} p]$. Ceci définit la fréquence de coupure telle que la courbe ne soit pas affectée par le filtre. /60/ donne des valeurs statistiques aux incréments de contours quantifiés.

Même si la fréquence de coupure n'est pas placée aussi bas que possible, le bruit est en partie éliminé, la longueur d'arc est mieux connue et un filtrage plus sélectif devient possible. La solution est alors itérative (cf. /58/ pour une étude plus détaillée d'un filtre curviligne itératif).

En pratique chaque élément de contour est considéré, caractérisé par une estimation de l'incrément local de l'abscisse curviligne. Le filtrage est continu par morceau et dans une phase ultérieure du traitement, lorsque la vraie longueur d'arc est suffisamment approchée, la courbe est échantillonnée (Fig.4.8.8).

Il résulte du paragraphe 4.2.2 que le théorème de Nyquist appliqué à l'échantillonnage des courbes exige que le pas

d'échantillonnage le long des contours soit au moins égal au rayon du plus petit cercle osculateur le long du contour lissé (Fig.4.8.7).

4.3 Lissage de courbes unidimensionnelles

Le filtre curviligne itératif décrit dans ce chapitre est particulièrement bien adapté aux courbes paramétrées, de dimension égale ou supérieure à 2.

Il est aussi possible d'appliquer l'algorithme à des courbes unidimensionnelles, en paramétrisant celles-ci. On obtient alors un filtre non-linéaire qui adoucit la courbe tout en gardant les pics marqués. En fonction de la fréquence de coupure choisie, un pic d'amplitude donnée peut subsister ou disparaître (Fig.4.11). Soit une fonction f , de dérivée f' , lissée par un filtre curviligne de largeur de bande B .

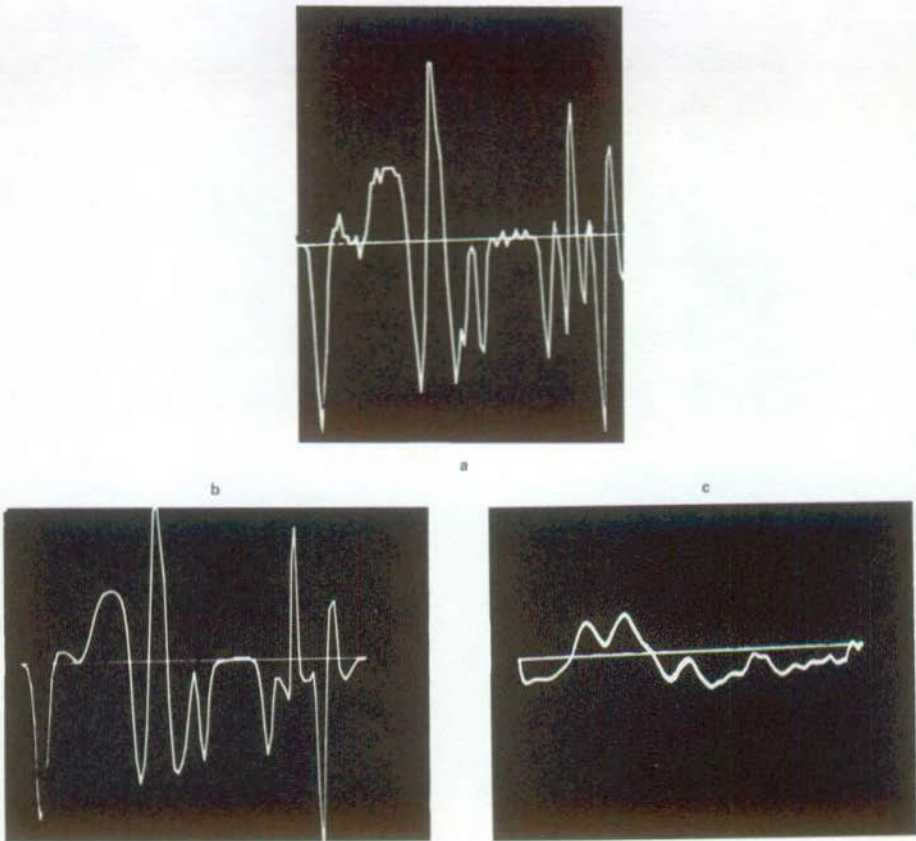


Fig.4.11 Par lissage curviligne itératif, les pics d'une fonction unidimensionnelle (a) peuvent être mis en évidence, alors que les zones de faibles variation sont lissées (b). En c tout est lissé, car le filtrage est plus sévère, et aucun pic n'est alors assez marqué.

Mathématiquement, la non-linéarité résulte de la contrainte (4.7) introduite lorsqu'on paramétrise la courbe. En première approximation, la largeur de bande (instantanée...) du filtre, B_{ai} , augmente ainsi:

$$B_{ai} = B \sqrt{1+f'^2} \quad (4.12)$$

Intuitivement, on peut voir que si la courbe est parallèle à l'axe de la variable libre, la "durée" apparente de la réponse impulsionnelle du filtre est grande; si au contraire la pente est

forte, la durée apparente est très courte.

5. EXTRACTION DES CARACTERISTIQUES

L'extraction des caractéristiques est une étape fondamentale entre la capture des données de l'environnement et la classification.

Son but est de concentrer au maximum l'information diffuse dans la scène pour ne garder que ce qui en fait l'essentiel.

Mis-a-part la texture (qui ne sera pas étudiée ici), toutes les informations d'une image résident dans les contours. Ceci souligne l'importance d'un prétraitement qui en optimise la qualité (Chapitre 4).

Le chapitre 5 traite des nombreuses caractéristiques couramment extraites des images. Par combinaison de caractéristiques simples, coordonnées, longueurs, pentes, d'autres paramètres se déduisent: Surface inscrite (si le contour est fermé), courbures, moments d'inertie, coefficients de forme, mesures statistiques... Des combinaisons plus complexes sont possibles aussi. Elles conduisent par exemple aux codages polaires, aux transformations de Fourier, aux représentations structurelles.

5.1 Caractéristiques simples.

Les contours sont connus par leurs coordonnées. En utilisant des relations de voisinage, pour autant que les points soient ordonnés, il est possible de diminuer le bruit local. Ceci se fait au détriment des très hautes fréquences du contour.

La première caractéristique associée aux coordonnées est un incrément de longueur d'arc. Entre chaque couple de points P_i, P_j on peut le décrire ainsi:

$$\Delta_{s1} = \sqrt{\Delta_{xi}^2 + \Delta_{yi}^2} \quad \text{où} \quad \begin{cases} \Delta_{xi} = x_{i+1} - x_i \\ \Delta_{yi} = y_{i+1} - y_i \end{cases} \quad (5.1)$$

Pour être symétrique, la longueur associée à chaque point peut se définir:

$$\Delta_{si} = (\sqrt{\Delta_{xi}^2 + \Delta_{yi}^2} + \sqrt{\Delta_{xi-1}^2 + \Delta_{yi-1}^2})/2 \quad (5.2)$$

La longueur d'un contour s'obtient simplement par intégration curviligne des incréments décrits:

$$L = \oint_S ds \approx \sum_i \Delta_{si} \quad (5.3)$$

La pente en tout point d'un contour se définit de la manière suivante:

$$\operatorname{tg} \theta = dy/dx \approx \Delta_y / \Delta_x \quad (5.4)$$

ce qui donne l'angle θ_i que fait la tangente au point i :

$$\theta_i = \operatorname{Arc} \operatorname{tg} \Delta_{yi} / \Delta_{xi} \quad (5.5)$$

Longueurs et pentes sont sensibles au bruit de quantification $n_x(s)$ et $n_y(s)$ (Ch. 4). Sous l'hypothèse d'un bruit indépendant du contour, l'estimation de longueur présente un biais positif. D'après Groen et Verbeek /60/, ce biais vaut 1/0.77 pour un contour décrit par le code de chaîne de Freeman. Lorsqu'on a au contraire une dépendance marquée entre contour et quantificateur, par exemple lorsqu'un segment de droite est numérisé sur une grille cartésienne, le biais (compensable) est moins significatif que l'écart-type. Celui-ci peut atteindre 10% (Fig.5.1).

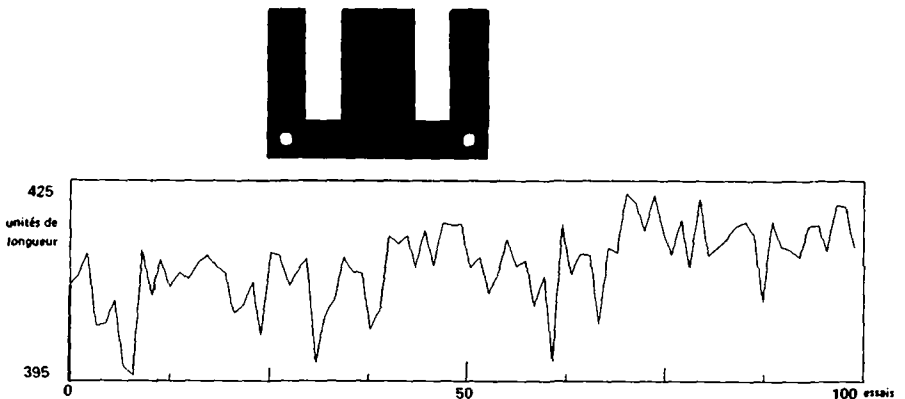


Fig.5.1 La longueur du périmètre de la pièce ci-dessus, estimée pour 100 orientations aléatoires, montre de grandes variations

malgré un léger lissage du contour.

La mesure de pente est encore plus sensible au bruit de quantification. Lorsque $y(s)$ est très petit, un faible bruit additionnel fait varier la pente entre $+\infty$ et $-\infty$. Sur une grille cartésienne, la pente ne peut en fait prendre que 4 ou 8 valeurs suivant la définition de connexité adoptée.

5.2 Mesures composites

Beaucoup d'autres paramètres se déduisent des coordonnées, des longueurs et des pentes. Il devient cependant toujours plus difficile d'analyser comment le bruit de quantification injecté dans $x(s)$ et $y(s)$ se propage.

- La courbure K en chaque point d'un contour s'exprime comme la dérivée locale de θ par rapport à s :

$$K(s) = d\theta/ds \approx \Delta\theta/\Delta s \quad (5.6)$$

Le signe de la courbure varie avec le sens du parcours le long du contour. Pour éviter toute ambiguïté, on en prend donc parfois la valeur absolue (par ex. /45/).

- Si le contour est fermé, une intégrale curviligne détermine la surface inscrite:

$$S = \oint_S y dx \approx \sum_i y_i \Delta x_i \quad (5.7)$$

- Les moments d'inertie forment une classe très intéressante de caractéristiques:

$$m(x^i, y^j) = \frac{1}{L} \oint_S x^i y^j ds \approx \frac{1}{L} \sum_k \Delta x_k^i \Delta y_k^j \Delta s_k \quad (5.8)$$

Par exemple le couple $(m(x^1, y^0), m(x^0, y^1))$:

$$(5.9)$$

est souvent utilisé comme repère de position. D'autres moments définissent les axes principaux d'inertie et ils sont employés pour la reconnaissance et pour l'estimation d'orientation d'objets à trois dimensions /61/. Les poids Δ_{Si} qui apparaissent dans les formules (5.8) et (5.9) ne peuvent être omis que lorsque la courbe est échantillonnée à intervalles réguliers.

- Des coefficients de forme, qui sont obtenus par une combinaison sans unité de caractéristiques servent de paramètres pour classifier les objets (par exemple: $L^2/4\pi S$, $L \cdot \sigma_K$, où σ_K est l'écart-type de la courbure). Le nombre d'Euler est également employé dans les images binaires pour caractériser des structures: Il représente le nombre d'objets disjoints, diminué du nombre de trous présents dans la scène.

5.3 Mesures statistiques

En statistique, la moyenne (espérance mathématique), la variance et l'écart-type sont des valeurs classiques. Pour une variable aléatoire x , de loi statistique (densité de probabilité) $p(x)$, elles se définissent ainsi:

$$m_x = E[x] = \int_{-\infty}^{\infty} x p(x) dx \quad (5.10)$$

$$\sigma_x^2 = E[(x - m_x)^2] = \int_{-\infty}^{\infty} (x - m_x)^2 p(x) dx \quad (5.11)$$

$$\sigma_x = \sqrt{\sigma_x^2} \quad (5.12)$$

Moins connus, les quartiles donnent des informations similaires sur une variable aléatoire: à la moyenne correspond la médiane q_2 et, comme l'écart-type, les quartiles q_1, q_3 peuvent caractériser la dispersion d'une variable aléatoire.

Les quartiles q_n sont les valeurs d'une variable aléatoire pour lesquelles la fonction de répartition $F(x)$ atteint $n/4$:

$$F(q_n) = n/4 \quad (5.13)$$

$$\text{ou} \quad F(x) = \int_{-\infty}^x p(\eta) \, d\eta \quad (5.14)$$

Comme $F(x)$ est une fonction monotone, l'équation (5.13) peut toujours s'expliciter. On a alors:

$$q_n \approx M[x] = F^{-1}(n/4) \quad (5.15)$$

Le quartile q_2 est appelé la médiane.

En statistique, il est connu que la médiane $M[x]$ conduit à un estimateur plus robuste et plus efficace que la moyenne $E[x]$.

$M[x]$ est plus efficace que $E[x]$ dans le sens qu'il est défini pour une plus grande classe de statistiques: Alors que toutes les densités de probabilité pour lesquelles une moyenne existe ont aussi une médiane, l'inverse n'est pas vrai (ex: loi de Cauchy).

La robustesse caractérise un estimateur faiblement sensible au caractère aléatoire des mesures, faites en nombre fini. En général $M[x]$ est aussi plus robuste que $E[x]$. Par exemple si x est régi par une loi gaussienne, m et q_2 tendent vers la même valeur asymptotique. Cependant lorsqu'un événement très improbable se produit, il fausse l'estimation m par son poids. Sur la médiane au contraire, il a une influence négligeable.

La robustesse est une qualité d'autant plus décisive que l'estimation se fait sur un nombre limité d'échantillons, ou sur une courte durée.

En pratique, le mode d'évaluation limite l'utilisation de q_2 . Alors que m ne requiert que des additions et une division, q_2 exige souvent plus de transferts de données car les mesures doivent être ordonnées pour permettre l'estimation des quartiles. Lorsque les mesures se font en nombre limité, ou qu'on est prêt à payer plus de calculs pour une précision plus grande, la médiane s'impose.

La moyenne et l'écart-type sont très utilisés en traitement d'image, pour la localisation d'objets et l'identification de scènes et de textures. Ils sont aussi au coeur des techniques stochastiques (cf. par exemple, la détection de courbes chez /62/).

La médiane conduit en particulier à un filtre non-linéaire souvent préférable au filtre passe-bas, dont le comportement asymptotique est le même si le processus obéit à une loi statistique symétrique, ou connue. Aucune application n'a encore été faite (à ma connaissance) où les quartiles q_1 et q_3 remplacent l'écart-type dans les mesures de textures; ils doivent s'y révéler très utiles. Le nombre d'échantillons traités y est faible et par leur robustesse, les quartiles sont bien adaptés à ce genre de problème.

5.4 Combinaisons complexes

Il est parfois utile de projeter l'image dans un autre espace, afin de mieux séparer certains aspects de la scène /63/.

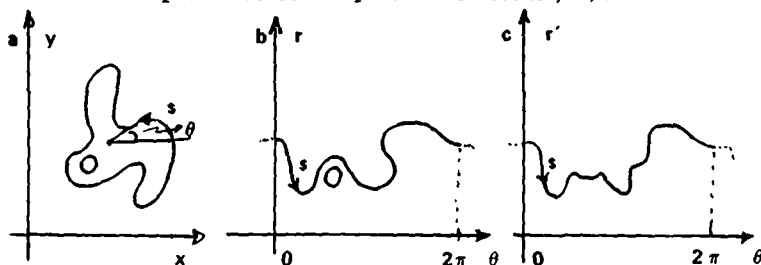


Fig.5.2 Les contours dans le plan de l'image (a) restent des contours dans le plan polaire (b). Une combinaison linéaire des valeurs multiples de r peut conduire à une description univoque /57/.

5.4.1 Transformation polaire

La transformation polaire applique l'image d'un référentiel cartésien $x-y$ à un référentiel polaire $r-\theta$ /57,64,65/ dont l'origine correspond à un point particulier de la scène (centroïde)(Fig.5.2).

Cette transformation est invariante à la position et facilite donc une classification ou une estimation d'orientation ultérieures.

Bien que la forme change, un contour dans $x-y$ reste un contour, par cette application dans $r-\theta$. En général, on ne peut pas facilement le représenter sous forme explicite:

$$C: r = g(\theta) \quad (5.16)$$

car sa définition n'est pas univoque.

Kammenos /57/ regroupe alors les multiples valeurs parfois associées au rayon, par une combinaison linéaire qui est non injective:

$$r' = \sum_{i=1}^{M(\theta)} \alpha_i g_i(\theta) \quad (5.17)$$

où $M(\theta)$ est le nombre de valeurs différentes que g peut prendre en θ ; et les α_i représentent des coefficients de pondération.

La représentation qui en résulte est beaucoup plus facile à traiter.

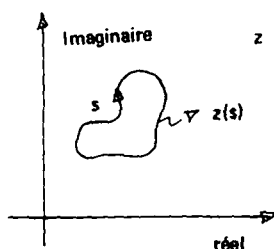


Fig.5.3 Un contour peut se représenter par une fonction complexe

5.4.2 Transformation de Fourier

Les chapitres précédents ont déjà introduit la représentation paramétrique des courbes. Parfois, un contour est représenté par une variable dans le plan

complexe (Fig.5.3):

$$z(s)=x(s)+i \cdot y(s) \quad (5.18)$$

$$Z(f)=F[z(s)] \quad (5.19)$$

Le formalisme est différent, mais le traitement est essentiellement le même que celui du chapitre 4. Au lieu d'opérer une transformée sur chacun des $x(s)$ et $y(s)$, réels, on calcule une transformée sur $z(s)$ complexe.

On se restreint ici aux courbes fermées, ce qui conduit à un spectre de raies (spectre discret). Les coefficients d'harmoniques contiennent alors toute l'information de la courbe /59/.

Un lissage et une réduction de redondance se combinent facilement avec cette approche, par sélection d'harmoniques appropriées. De plus, certaines harmoniques contiennent implicitement des informations de position et d'orientation du contour /66/.

5.4.3 Caractéristiques structurelles

Il est courant aussi que l'on retiennent des caractéristiques de forme d'une courbe, telles que le plus grand rayon inscrit (annexe 2), le plus grand rectangle inscrit, le plus petit rectangle circonscrit, la position relative de trous ou de repères (/58,67,68/).

Le plus grand rayon en coordonnées polaires ou le nombre des lobes d'une cellule en biologie sont deux parmi de nombreux autres exemples de caractéristiques structurelles. Ces caractéristiques servent soit à identifier des scènes, des objets, soit par exemple à en déterminer l'orientation.

Il se trouve deux catégories de caractéristiques structurelles. La première contient des caractéristiques synthétiques qui retiennent le minimum d'information d'un

objet (par exemple: le plus grand rectangle circonscrit). La seconde, beaucoup plus intéressante, exprime des particularités de la scène sous une forme peu sensible aux distorsions (par exemple le nombre de lobes d'une feuille). Le chapitre 6 discute en particulier de la différence entre "bruit" et "distorsion".

5.5 Caractéristiques locales ou globales

Au moment de choisir des caractéristiques, le problème apparaît toujours de retenir des informations globales, qui caractérisent l'ensemble d'un objet ou au contraire des informations locales, qui n'en décrivent qu'une partie.

Les caractéristiques globales sont très synthétiques et permettent de réduire à un minimum la description d'un objet. Elles sont extrêmement utiles pour son identification. Le périmètre, les moments d'inertie, la surface, les coefficients de Fourier appartiennent, entre autres, à cette catégorie. Les caractéristiques globales ont toutefois trois défauts:

- Les petits détails d'une image se perdent dans l'effet de moyennage lié à l'extraction de caractéristiques globales.
- La sensibilité au bruit est parfois mauvaise. Toutes les perturbations de l'image sont prises en compte. Des perturbations ponctuelles peuvent donc fausser l'estimation et même les petites erreurs peuvent, dans les cas défavorables, se cumuler.
- Il est enfin évident que les caractéristiques globales ne peuvent pas s'estimer si un objet n'est pas traité dans son ensemble. Or il arrive souvent qu'un objet ne puisse pas être complètement perçu, à cause d'un manque de contraste ou parce qu'il est partiellement caché.

5.6 Propriété d'invariance

Une image peut se représenter de nombreuses manières. Chacune correspond à un modèle. Une image numérisée est par exemple une fonction de deux variables exprimée par $N \times N$ points, mais c'est aussi un point dans un espace à N^2 dimensions ou encore N^2 points à une seule dimension (balayage TV).

En robotique industrielle une image est intuitivement décrite comme un ensemble d'objets, auxquels on associe un type, une taille, une position (x, y, θ), un état (pile, face, $n^{\text{ième}}$ état stable)... L'image peut alors s'exprimer par un ensemble de points dans l'espace engendré par ces caractéristiques. Certaines dimensions sont discrètes par nature (état, type), d'autres peuvent être continues. Ce modèle est extrêmement bien adapté aux connaissances a priori qu'on a de la scène, et la quantité d'information de l'image est alors extrêmement faible (à moins qu'on ait une multitude d'objets pathologiquement petits).

Pour une image binaire de 1000×1000 points, on a 10^6 bit sous forme brute. En la représentant comme un ensemble de points dans un espace particulier basé sur le type (par ex.: 10 bit), la position en x (10 bit), en y (10 bit), l'orientation (10 bit), l'état stable (10 bit), on n'a que 50 bit par objet de la scène. Le modèle est intéressant s'il y a moins de 20000 objets simultanément sur la scène!

Un invariant peut se définir mathématiquement comme un hyperplan dans l'espace des caractéristiques. C'est un sous-espace qui a une ou plusieurs dimensions de moins que l'espace complet. Une diminution de dimensionnalité entraîne un très grand gain lorsqu'on utilise une méthode brutale comme la corrélation pour comparer des fonctions car la quantité des calculs impliqués suit la dimension selon une loi exponentielle.

Appelant $N_1, N_2, \dots, N_p$ le nombre des niveaux selon les p dimensions d'un espace numérisé, on passe de Q_1 opérations utilisant tout l'espace:

$$Q_1 = (N_1 \cdot N_2 \cdot \dots \cdot N_p)^2 \quad (5.20)$$

a p comparaisons par l'utilisation d'invariants.

Un invariant est par définition orthogonal a une partie de l'espace, et a la limite, p invariants peuvent former une base orthogonale de l'espace considéré. Selon un modèle quelconque, si l'un des parametres n'est pas independant des autres, il peut être négligé. Si au contraire il est indépendant, il peut très simplement conduire a un invariant ... par soustraction de ses composantes dépendantes (cf. la procédure d'orthogonalisation de Gram-Schmitt).

Par exemple dans une image $b(x,y)$, les coordonnées x_i des points d'un objet sont dépendantes de la translation de l'objet selon x . Pourtant après soustraction de la coordonnée x_0 d'un point arbitrairement choisi sur l'objet, les coordonnées $(x_i - x_0)$ sont trivialement invariantes a la translation selon x .

6. IDENTIFICATION

L'identification de classes connues a priori ("classification with teacher") se fait par trois types d'approches, qui sont adaptées aux caractéristiques utilisées. Les méthodes corrélatives traitent des caractéristiques très locales, présentes en grand nombre dans la scène. Les méthodes de classification dans l'espace des caractéristiques sont particulièrement indiquées pour l'interprétation des paramètres globaux. Enfin, les méthodes structurales sont hybrides. Traitant en général des mesures régionales, elles incluent une étape d'identification des lettres de leur alphabet par classification ou par corrélation, même si elles gardent parfois comme attributs les valeurs brutes des caractéristiques tirées de la scène /69,70,71/.

6.1 Méthodes corrélatives

Les méthodes corrélatives sont très répandues, sous des formes diverses.

6.1.1 Fonction de corrélation

La fonction de corrélation, mono ou bidimensionnelle, est classique en traitement du signal:

$$R(t) = \int_{-\infty}^{\infty} x(\tau-t) y(\tau) d\tau \quad (6.1)$$

et, sous forme numérique:

$$R(n) = \sum_i x(i-n) y(i) \quad (6.2)$$

A deux dimensions, la fonction de corrélation s'exprime ainsi:

$$R(t,u) = \iint h(\tau-t, \eta-u) y(\tau, \eta) d\tau d\eta \quad (6.3)$$

$$R(n,m) = \sum_{ij} h(i-n, j-m) y(i, j) \quad (6.4)$$

Si $x(t)$ est un signal connu, $y(t)$ peut être détecté par corrélation. Un bruit additionnel à moyenne nulle, indépendant de $y(t)$ est théoriquement sans influence sur cette détection. La fonction de corrélation a une parenté très étroite avec le filtre adapté [72].

La fonction de corrélation paie par plusieurs défauts sa remarquable insensibilité au bruit:

- Les multiplications sous l'intégrale de corrélation ne permettent pas d'évaluer la similitude entre deux signaux dont l'un est nul. Or il se trouve des situations où les parties nulles d'une fonction contiennent autant d'information que les autres. Un segment de droite par exemple a une courbure nulle et ne doit pas être confondu avec un arc de cercle.
- Les multiplications conduisent à une complexité relativement grande, lorsque la corrélation doit s'évaluer par des circuits spécialisés.
- Par nature, la corrélation exige un grand nombre de calculs. Dans son expression discrète, elle exige N^2 multiplications pour l'évaluation de la corrélation de signaux mono-dimensionnels à N échantillons. L'utilisation de la transformée de Fourier rapide permet de se ramener à une dépendance en $N \log N$. Cependant ceci est encore souvent trop.

Pour les images, la corrélation entraîne même N^4 multiplications. Il faut remarquer à ce propos, que, même en ayant une mémoire de taille fixe, on peut échanger de la définition spatiale contre de la définition en tons de gris, avant la corrélation ([13]).

- On se trouve parfois en présence de distorsions sur le signal perçu, qui ne permettent pas la comparaison globale par corrélation. Le modèle classique d'un canal perturbé, exprimant la sortie comme la somme d'une entrée et d'un

bruit additif n'est plus adapté (cf. 6.3)

6.1.2 Substituts de la fonction de corrélation

Pour pallier à divers défauts de la corrélation "classique", plusieurs simplifications sont parfois envisagées:

- Corrélation de signes. On ne retient, pour la corrélation de signes, que le signe du signal connu et celui du signal reçu. L'opération de multiplication est alors triviale. Elle se limite à un changement de signe conditionnel. On a essentiellement une décision binaire: égal/différent.

Avant de calculer le signe, il est parfois nécessaire de centrer le signal (variable aléatoire, ou fonction déterministe). Ceci se fait par soustraction de la moyenne.

- Intégrale des différences en valeur absolue (IDVA). Au lieu de calculer un produit, sous l'intégrale de corrélation (6.2), on évalue la valeur absolue d'une différence. Pour des valeurs discrètes, à une dimension il résulte:

$$D(n) = \sum_i |x(i-n) - y(i)| \quad (6.5)$$

et, pour une image:

$$D(n, m) = \sum_{i,j} |x(i-n, j-m) - y(i, j)| \quad (6.6)$$

Bien que cette mesure de similitude soit moins facile à traiter analytiquement que la corrélation, elle a d'autres avantages, souvent déterminants. Cette fonction permet d'estimer la ressemblance entre des signaux dont l'un est nul. Cette propriété peut être fondamentale pour un signal à deux dimensions aussi bien qu'à une. Dans nos applications, une courbure nulle est associée à des segments de droites. À deux dimensions, la valeur nulle désigne souvent le fond d'une image.

Sur ordinateur ou lors de la réalisation de cette fonction par des circuits spécialisés, le remplacement des multiplicateurs par des circuits arithmétiques plus simples n'entraîne pas de gains substantiels en temps de calcul, car la limitation principale vient généralement du transfert des données.

6.2 Classification

Dans ce paragraphe, pour éviter des répétitions trop nombreuses, on utilisera indifféremment, "mesure" et "caractéristique", faisant l'hypothèse que chaque caractéristique est estimée par une seule mesure.

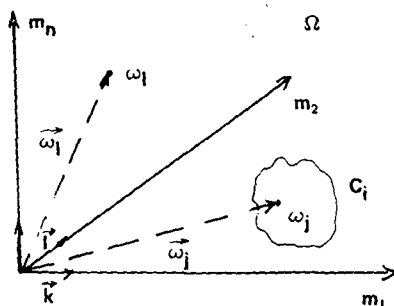


Fig.6.1 Dans l'espace des caractéristiques Ω , l'événement ω_j appartient à la classe C_i .

Le problème fondamental de la reconnaissance des formes c'est d'une part la définition de N classes (C_i) d'objets, de scènes ou, plus généralement d'événements, dans l'espace (Ω) des caractéristiques (m_k), c'est d'autre part la classification, ou l'attribution d'un événement à l'une des classes définies (Fig.6.1).

Le premier aspect est essentiel en analyse des données, où la complexité et la multitude des données ne permettent qu'une solution automatique. Au contraire, pour nos applications en robotique, il se réduit à peu de chose; un opérateur définit une nouvelle classe en "montrant" à la machine un objet nouveau (cf. chapitre 7).

La deuxième facette nous concerne plus. Sur la base de caractéristiques mesurées sur la scène (cf. chapitre 5), l'événement perçu doit s'associer à une classe prédéfinie.

6.2.1 Espace des caractéristiques

L'espace des caractéristiques est engendré par les mesures (périmètre, surface inscrite ...) faites sur la scène. La dimension de cet espace est au plus égale au nombre de mesures faites. Celles-ci sont rarement orthogonales et même pas toujours indépendantes. Alors que le choix premier des caractéristiques utilisées est laissé à l'intuition de l'ingénieur, la sélection peut ultérieurement se faire avec l'aide d'une machine. Après orthogonalisation de l'espace, certaines caractéristiques se révèlent superflues.

6.2.2 Différence entre événements

Chaque événement correspond à une configuration particulière des mesures, qui peut se représenter par un point dans l'espace des caractéristiques.

La différence entre événements peut alors se définir comme une distance. En supposant un espace où les mesures sont orthogonales, on utilise couramment deux distances d :

- La distance euclidienne d_E :

$$d_E = \| \vec{\omega}_i - \vec{\omega}_j \| \quad (6.7)$$

- La "city block distance" d_C :

$$d_C = \langle \vec{\omega}_i - \vec{\omega}_j, \vec{k} \rangle + \langle \vec{\omega}_i - \vec{\omega}_j, \vec{l} \rangle + \dots \quad (6.8)$$

où $\langle, \rangle$ dénote le produit scalaire, $\| \cdot \|$ la norme, $\vec{\omega}_i, \vec{\omega}_j$ sont les vecteurs de l'origine à ω_i et ω_j et $\vec{k}, \vec{l}, \dots$ les vecteurs unitaires de la base. Cette distance est très facile à calculer, puisque c'est simplement la somme des différences selon chaque coordonnée.

En pratique, chaque caractéristique est souvent pondérée pour tenir compte de sa dispersion ou de son importance.

6.2.3 Définition de classes (apprentissage)

La définition automatique de classes sur la base d'événements mesurés est assez rare en traitement d'images. Par exemple, on l'utilise parfois pour segmenter de manière "optimale" une image en N classes (régions) de texture uniforme /73/.

L'algorithme le plus utilisé est celui des "nuées dynamiques". On fixe au départ, arbitrairement, N centres de classes G_i dans l'espace des caractéristiques (Fig.6.2). Chaque événement mesuré est ensuite attribué au G_i le plus proche. Finalement chaque G_i est recalculé comme la moyenne des points de sa classe. Ce cycle est répété un certain nombre de fois. Les solutions stables sont multiples, dépendant des conditions initiales, mais l'algorithme converge en général assez bien vers une solution "raisonnable".

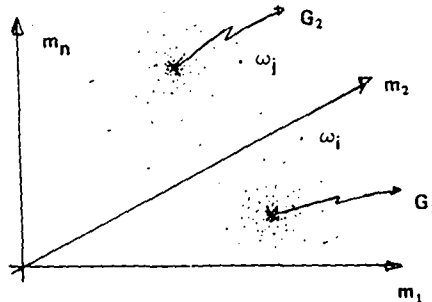


Fig.6.2 G_1 et G_2 sont les centres de deux classes.

Lorsque la définition des classes est assistée par un opérateur, elle est grandement simplifiée. Par exemple une simple mesure de la référence définit la classe. Il est souvent préférable de répéter la mesure un certain nombre de fois, pour permettre l'estimation non seulement d'un point dans l'espace, mais également une densité spatiale de probabilité. L'espace se divise alors en volumes, chacun

associe a une classe. Il faut remarquer que "classification", en anglais, designe la definition de classes, alors qu'en français cela signifie l'identification d'un evenement.

6.2.4 Identification ou classification

Un evenement ω_o est identifié comme appartenant a la classe C_j si un certain critere T est satisfait. Ce critere est de complexité variable, suivant le type de classes rencontré. Par exemple pour une distribution uniforme autour des G_j nous avons:

$$T_u: \omega_o \in C_j \text{ si } d(\omega_o, G_j) \leq d(\omega_o, G_k) \quad (6.9)$$

pour tout k.

Pour l'identification d'un evenement parmi des classes C_j , de formes quelconques, contenant les elements ω_1 , la regle du plus proche voisin donne:

$$T_{NN}: \omega_o \in C_j \text{ si il existe } \omega' \in C_j \text{ tel que } d(\omega_o, \omega') \leq d(\omega_o, \omega_1) \quad (6.10)$$

pour tout 1.

6.3 Méthodes structurales

Les objets se caracterisent parfois moins par leurs propriétés locales que par la structure générale de leurs parties. Les arbres et les grammaires sont alors très utiles pour leur description.

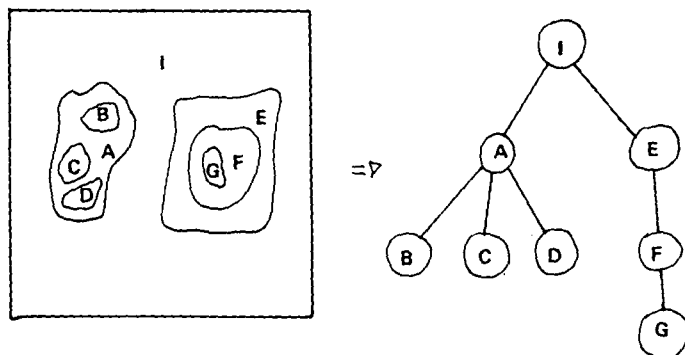


Fig.6.3 Représentation structurale d'une image (gauche) à l'aide d'un arbre (droite)

6.3.1 Arbres

La représentation de structures par des arbres est plaisante car elle est très facile à interpréter. C'est la méthode la plus naturelle.

En utilisant simplement des relations d'appartenance, ou d'inclusion, une structure peut d'une part se définir, d'autre part se reconnaître (Fig.6.3).

Description et reconnaissance peuvent se faire selon un schéma deductif (du général au particulier) ou inductif (du particulier au général).

Les limitations des méthodes en arbre viennent principalement du modèle qui leur est associé. Dans l'exemple ci-dessus, les relations topologiques qui font l'essentiel du modèle ne permettent pas la comparaison pondérée et quantitative des structures. La comparaison de deux arbres conduit alors essentiellement à une décision binaire: identité/différence. Par contre, dans un autre contexte, on peut définir des distances plus fines entre arbres (par exemple /83/).

6.3.2 Grammaires

Les grammaires G sont définies comme des quadruples:

$$G: \{ S, T, NT, P \} \quad (6.11)$$

S , symbole de départ, correspond à la scène analysée. T est l'ensemble des symboles terminaux, c'est-à-dire des primitives ou caractéristiques, qui forment les lettres de G . NT contient des combinaisons de primitives que les règles de production P engendrent.

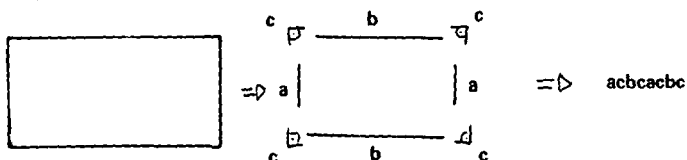


Fig.6.4 Représentation structurale d'un rectangle à l'aide d'une grammaire.

Dans l'exemple de la Fig.6.4, la lettre "c" remplace un angle droit, la lettre "a" correspond à la primitive "petit segment de droite", alors que "b" désigne "un grand segment de droite".

Une grammaire qui admet le symbole non-terminal $acbcacbc$ permet la description d'une large classe de rectangles. Si on veut reconnaître un carré, et qui plus est, un carré dont l'arête est un segment de droite de longueur "moyenne", on peut enrichir la grammaire avec la notion d'attribut.

Une grammaire avec attributs permet un premier tri très rapide de structures, par analyse grammaticale, tout en garantissant ensuite, une identité très précise sur la base des attributs.

Revoyons l'exemple précédent. G' , différent de G par une variable "d", associée à un attribut l (longueur), au lieu de a et b , permet la représentation précise des carrés.

Les primitives d'une grammaire peuvent se classer schématiquement en deux groupes, correspondant à des caractéristiques respectivement qualitatives et quantitatives. Lorsqu'on analyse un symbole non-terminal pour voir s'il obéit aux règles d'une certaine grammaire, on peut comptabiliser le nombre d'incompatibilités mais non leur degré. Ceci est surtout adapté au traitement des primitives qualitatives.

Si par exemple des primitives qualitatives décrivent différentes pièces d'assemblage (vis, boulons...) seule la décision identité/différence a un sens. Si par contre on analyse des trous de diverses dimensions, un certain degré dans la différence se justifie.

Deux manières différentes se sont principalement répandues pour concilier variables quantitatives et grammaires. L'une consiste à numériser les variables, traitant ensuite qualitativement les états discrets. L'autre associe un attribut à un descriptif assez général. Si par exemple une structure nécessite l'usage d'angles pour sa description, on peut soit utiliser n valeurs discrètes, soit le descriptif général "angle" associé à l'attribut adéquat.

Les variables floues, avec leur fonction d'appartenance, forment un sous-ensemble intéressant de primitives pour grammaires avec attributs.

6.4 Relations entre techniques de comparaison

Corrélation, classification, et analyse structurale sont des méthodes complémentaires en reconnaissance des formes. Elles sont liées à des échelles d'observation différentes et permettent la comparaison et l'identification hiérarchisée des scènes. par exemple sur deux fonctions à p échantillons, l'estimation d'IDVA, présentée plus haut comme une technique corrélatrice, peut aussi théoriquement se modéliser comme une "city block distance" d_c dans un espace à p dimensions. En pratique les techniques de classification travaillent sur des espaces de dimension beaucoup

plus petite que p . De plus, la distance d_c est un scalaire alors que l'IDVA est une fonction (vecteur).

Au niveau élémentaire, la corrélation traite des caractéristiques locales. Ces caractéristiques sont les premières, et parfois les seules significatives, qui décrivent une image. La corrélation garantit la détection optimale au sens des moindres carrés d'un signal sans distorsion additionné de bruit.

A un niveau plus élevé, quelques caractéristiques globales, très synthétiques, permettent souvent, quand elles peuvent s'extraire, l'identification de l'image. Ces caractéristiques engendrent un espace où la classification est généralement simple et rapide.

Les techniques structurales utilisent le plus souvent la corrélation ou la classification pour reconnaître les primitives sur lesquelles elles travaillent. Elles remplacent avantageusement la corrélation, à une échelle d'observation plus élevée, pour trois raisons principales:

- Alors que les distorsions sont négligeables dans les petites régions, elles deviennent parfois prohibitives pour la corrélation à l'échelle de l'image.
- Même si les distorsions sont nulles, la corrélation ne peut souvent pas être utilisée à une grande échelle à cause de la croissance rapide des calculs qu'elle requiert.
- La scène à reconnaître est parfois essentiellement formée de parties bien définies, qui doivent être reconnues comme telles, et dont l'agencement relatif est décrit de manière simple et adéquate par les méthodes structurales, à l'échelle de l'image entière (par ex. /14/).

7. APPLICATION EN ROBOTIQUE

Les chapitres 2 à 6 discutent les concepts fondamentaux et différentes techniques du traitement d'images. Le chapitre 7 présente un mélange particulier de ces techniques, pour l'identification et la localisation de pièces plates. Les résultats correspondants sont indiqués.

Le premier paragraphe (7.1) expose les problèmes industriels que nous avons voulu résoudre, puis une vue synthétique des solutions apportées. La suite du chapitre (7.2 à 7.8) revient en détail sur chaque maillon de la chaîne de traitement proposée.

7.1 Problèmes... et solutions

Notre étude s'est attaquée à la reconnaissance et à la localisation automatiques de pièces mécaniques. Le problème s'est cantonné à des scènes essentiellement bidimensionnelles, sans texture. Si des pièces tridimensionnelles se présentent, elles doivent offrir un nombre fini de profils. De plus, le profil ne doit pas trop varier lorsque les pièces se superposent.

Tenant compte de contraintes industriels, nous avons développé deux solutions, l'une pour le cas où les pièces ne se touchent pas, et la seconde pour celui où des objets plats peuvent se superposer à quelques niveaux (Fig. 7.1, 7.2, 7.3).

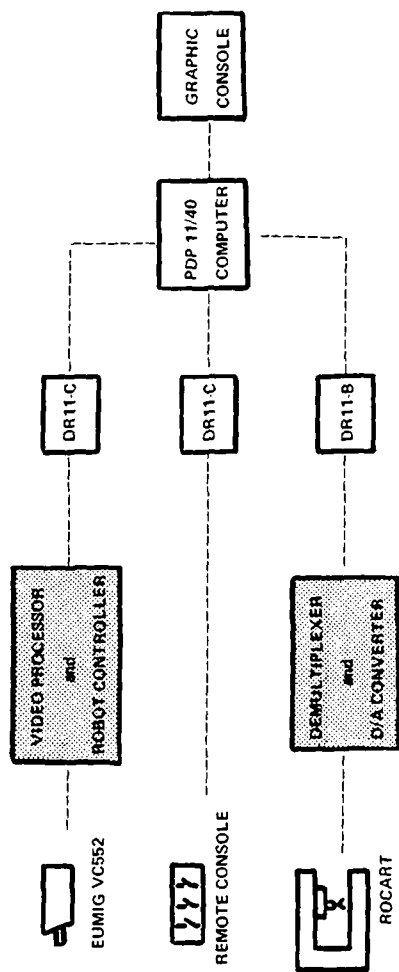


Fig.7.1 Schema-bloc du matériel utilisé. On voit, tramés, les circuits spécialement développés au LTS

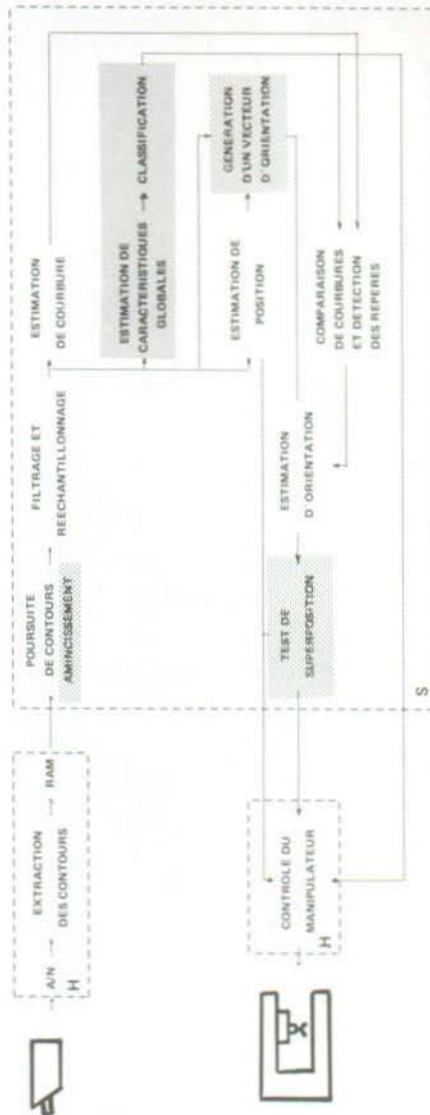
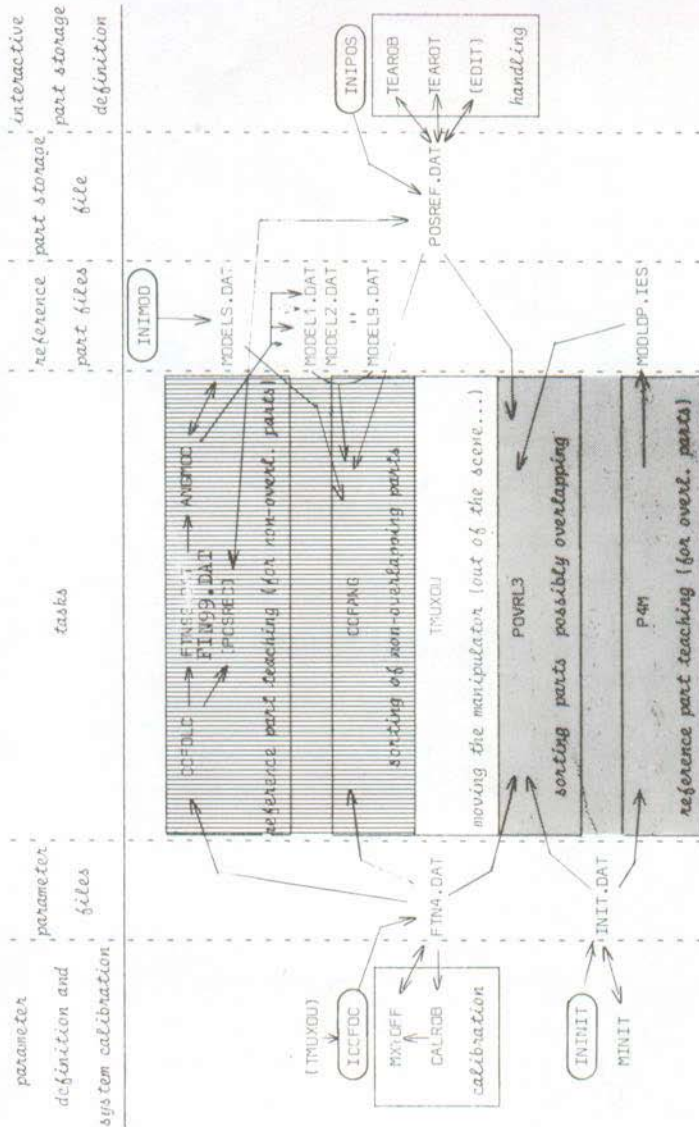


Fig.7.2 Schéma-bloc des fonctions impliquées pour la reconnaissance et la localisation de pièces industrielles plates. Finement trames, les blocs spécifiquement utilisés pour le traitement de pièces entièrement visibles. La trame grossière indique les extensions nécessaires lorsque les pièces peuvent se toucher ou se superposer



This chart summarizes the commands and files used for automatical "flat" part sorting. "□" denotes an initialisation command. The arrows show the information flow.

Fig.7.3 Vue d'ensemble des commandes et des fichiers nécessaires au traitement de pièces industrielles plates. La zone hachurée correspond aux commandes spécifiques pour le traitement de pièces entièrement visibles. La trame dénonce les commandes pour le traitement de pièces éventuellement enchevêtrées

7.1.1 Contraintes industrielles

Pour être utile en atelier, un outil robotisé doit satisfaire de nombreuses contraintes. Voici les principales:

- Rapidité. La cadence d'une chaîne de production varie avec la taille des objets manipulés. Pour les pièces estampées par exemple, la cadence d'une pièce par seconde est typique.
- Faible coût. Il est clair que pour être rentable, le robot doit produire à meilleur marché que toute autre solution. En concurrence avec le robot, on trouve d'une part le travailleur humain et d'autre part la machine spécialisée.

Le travailleur manuel change de type de pièce sans investissement particulier. Par contre, son coût augmente proportionnellement (avec une forte pente) au nombre de pièces manipulées.

Pour un type de pièce donné, la machine spécialisée a un coût marginal quasi-nul. Par contre un gros investissement initial est requis lorsqu'on change de production.

Il reste donc au robot à rentabiliser son coût initial élevé en traitant un grand nombre de pièces, prises de petits lots différents (Fig.7.4).

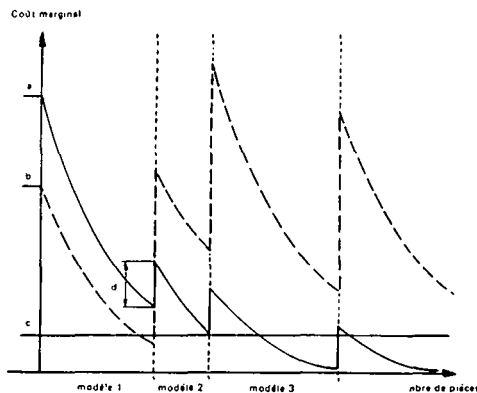


Fig.7.4 Représentation schématique du coût marginal de production en fonction du temps. L'investissement pour un robot (a) est généralement beaucoup plus élevé que pour une machine spécialisée (b). Le coût marginal d'un poste non-automatisé est constant (c). On note en particulier que les frais entraînés par un changement de production sont relativement modestes, pour un poste robotisé (d).

- Adaptabilité. Un robot est un outil peu spécialisé, qui doit permettre la production par petits lots. Il est donc essentiel que l'interface opérateur-robot soit simple, afin que les nouvelles tâches soient définies facilement.
- Fiabilité. Alors que le taux de fausses décisions est sans grande importance dans certaines applications (tri postal, mesures statistiques...), il est vital en atelier que les fausses décisions soient évitées. Elles bloquent la chaîne de production et parfois même endommagent les machines-outils.

7.1.2 Pièces qui ne se touchent pas

Comment reconnaître et positionner des pièces multiples dans le champ de vue, sous l'hypothèse qu'elles ne se touchent

pas?

Si les pièces à manipuler sont séparées les unes des autres, il est généralement possible de bien les voir. Un fond judicieusement choisi (ce choix est facile à faire) contraste bien avec les objets, et la scène peut se représenter par une image binaire. L'opérateur de Roberts (cf. 7.3.3) révèle la présence des bords, en temps réel, à partir du signal vidéo numérisé. L'image des contours est alors analysée, et, à partir d'une zone d'activité rencontrée aléatoirement, une poursuite des bords segmente l'image. Du bord poursuivi, puis lissé, différentes caractéristiques s'extraient (longueur, surface, courbure, centroïde), qui permettent la reconnaissance et l'estimation de position des objets.

Si un objet de la scène est reconnu, on compare sa fonction de courbure à celle de la référence, et on découvre ainsi les repères qui définissent l'orientation de la pièce. Le cycle se termine par la prise de l'objet et par son positionnement dans une place préalablement indiquée (Fig.7.5 et /58,74/). Avec le matériel de la Fig.7.1, la cadence de travail possible est d'une pièce par cinq secondes.

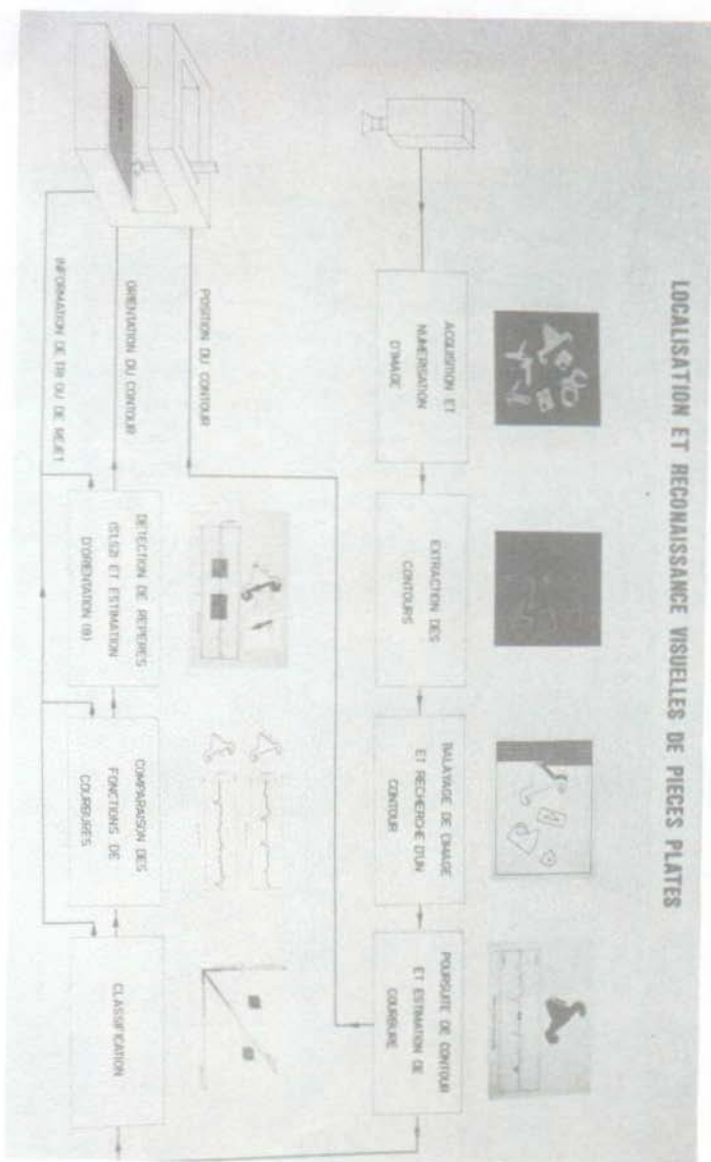


Fig.7.5 Schéma-bloc du traitement des pièces qui ne se touchent pas

7.1.3 Pièces plates avec recouvrement partiel

Comment reconnaître et positionner une pièce partiellement cachée, ou au-dessus d'autres pièces du même type?

Le bord des pièces, dans une image, se traduit par une différence en intensité lumineuse. Si les pièces sont de même type et se superposent, on ne peut garantir que cette différence soit perceptible. Plus la définition en gris perçus augmente, plus se réduit la probabilité de non-détection des bords.

Notre montage code l'intensité lumineuse à 4 bit. L'opérateur de Roberts en extrait les contours, qui peuvent tour à tour être épais, s'entrecroiser et disparaître.

Lorsqu'une zone d'activité se détecte, un algorithme de poursuite et de squelettisation isole les contours par segments. Dès qu'un segment est suffisamment grand, il est lissé. Ensuite, on identifie le type de l'objet correspondant, par comparaison des courbures. Un vecteur d'orientation est automatiquement défini sur le segment de la scène. Reporté sur la référence, il permet l'estimation d'orientation relative.

Enfin, la référence est brièvement corrélée avec la scène, confirmant que la classification et l'estimation d'orientation sont sans erreur (Fig.7.6 et /75/). Le temps de calcul varie avec la complexité et le contraste de la scène. Avec le matériel de la Fig.7.1, la cadence atteinte est d'une pièce par 25 secondes si chaque segment de contour traité permet la reconnaissance d'une pièce. Dans la mesure où la scène présente des segments de contour courts à cause de superpositions nombreuses, ou si le contraste insuffisant ne révèle pas certains croisements des contours, un temps additionnel variable est requis.

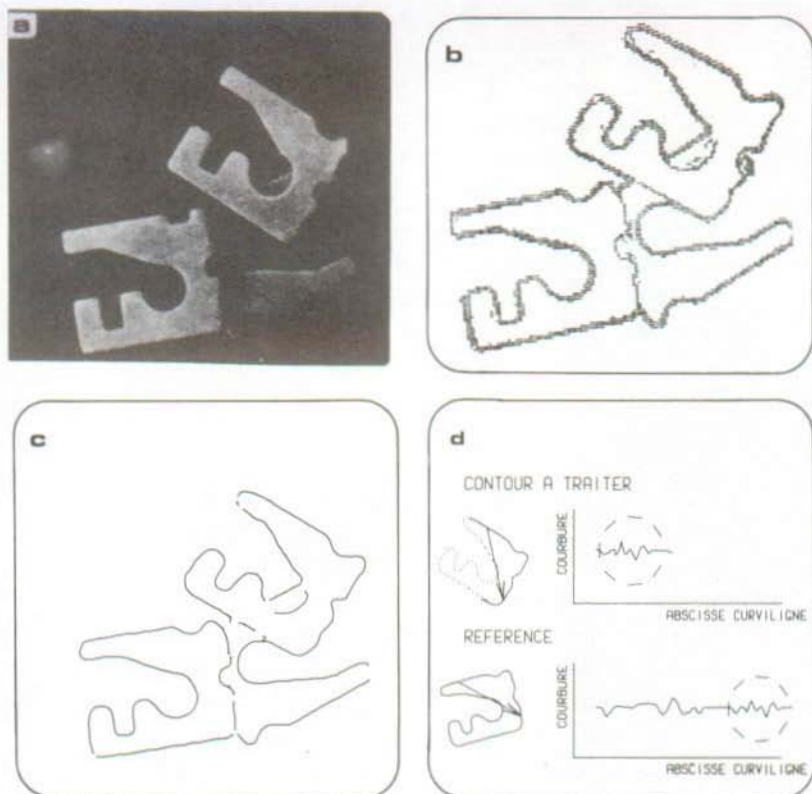


Fig.7.6 Schéma-bloc du traitement de pièces qui peuvent se chevaucher. Après numérisation de l'image (a), les contours sont extraits (b). Un à un, les segments sont poursuivis, squelettisés, et filtrés (c). (d) Sur chaque segment, un vecteur d'orientation est défini, et, après identification par comparaison de courbures, se voit reporté sur la référence correspondante. La position relative étant connue, la référence est enfin déplacée sur la scène pour vérifier la présence des bords non poursuivis (pointillés).

7.2 Montage expérimental

La robotique fournit un terrain d'essai très intéressant pour les techniques nouvelles développées en traitement d'images: Les performances peuvent immédiatement se visualiser, à une échelle et

a un rythme très humains.

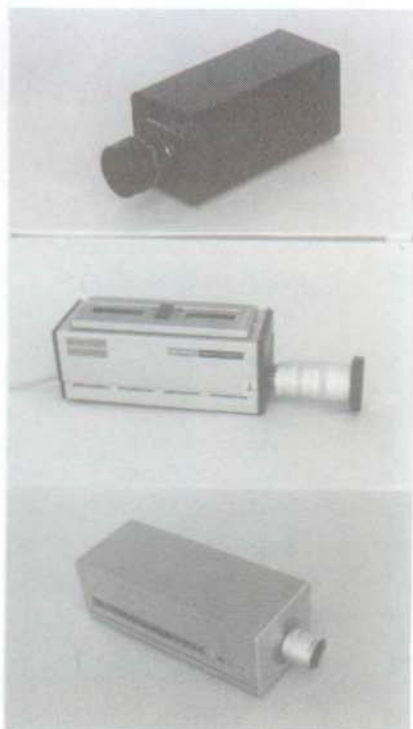
Un robot se compose schématiquement de trois parties: des capteurs d'information, des unités arithmétiques et logiques, avec enfin des actionneurs.

7.2.1 Capteurs

Les essais ont porté sur l'utilisation de caméras à tube vidicon. Chez Brown Boveri Corp. (R.Karg) et au Laboratoire de traitement des signaux (J.M.Zürcher), des tests ont démontré la mauvaise qualité des cibles à transferts de charges livrées en Suisse. D'autres caméras "solid state" (par ex. la 2200 de General Electric) sont aussi très difficiles à obtenir.

Alors que la majorité des tests utilisent une caméra C1 (Eumig VC552) ou C2 (Sony AVC-325 OCE), de qualité "amateur", une caméra de haute qualité C3 (Hamamatsu C1000) sert de référence.

Le tableau de la Fig.7.7 montre leurs caractéristiques respectives en rapport signal sur bruit et en distorsion géométrique, pour une optique donnée. Dynamique de gris perçus et diffusion (blooming) sont surtout liés au choix de la cible à vidicon et sont semblables sur les trois caméras. La Fig.7.8 présente les mires de tests utilisées pour ces mesures.



	SNR	Résolution nbre de lignes au centre
Eumig	40 dB	450
Sony	44 dB	550
Hamamatsu	>40 dB	>800

Fig.7.7 Caractéristiques principales des caméras utilisées pour nos expériences

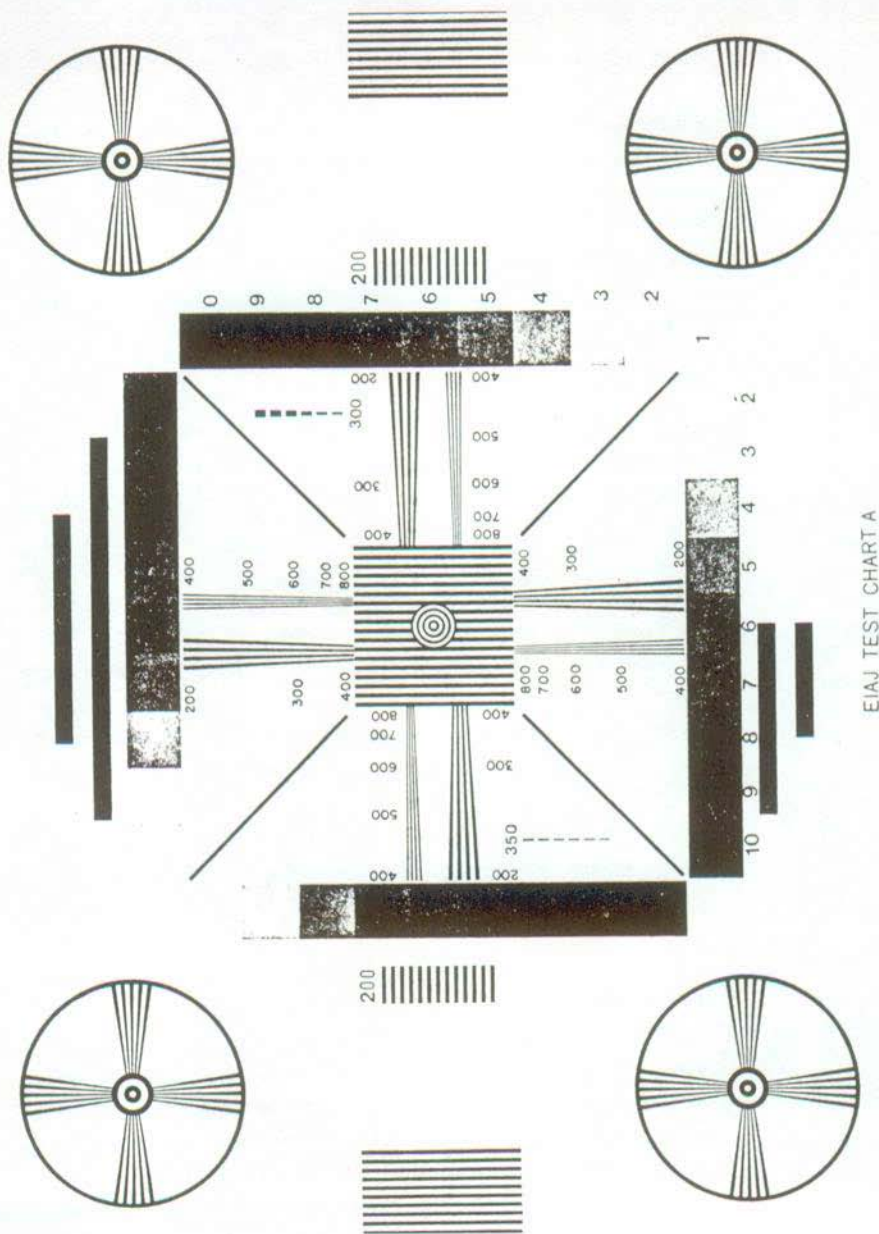


Fig.7.8 Mires de test

7.2.2 Unités de traitement

Souplesse et généralité vont à l'encontre de la vitesse et de l'économie de moyens. Dans la recherche de nouveaux algorithmes, les premières qualités ont la priorité.

- Un miniordinateur PDP 11/40 permet d'abord l'étude de faisabilité et ensuite l'évaluation des performances de nouveaux algorithmes. Il peut même simuler, à vitesse ralentie, un processeur spécialisé.

Dans notre configuration, le miniordinateur dispose principalement de 24 kilomots de 16 bit en mémoire centrale, d'une unité câblée de calculs en virgule flottante, et de disques RK05..

La quasi-totalité des programmes sont écrits en Fortran, le reste l'étant en langage assembleur (Macro-11). Le système d'exploitation (RT11-FB) est résident en mémoire et certains programmes utilisent la technique des "overlays". La plus grande partie du temps est consacrée aux transferts des données et des programmes.

- Une unité câblée de prétraitement d'image, développée au LTS, assure en temps réel l'extraction des contours, à partir du signal vidéo d'une caméra de télévision (cf. /76/).

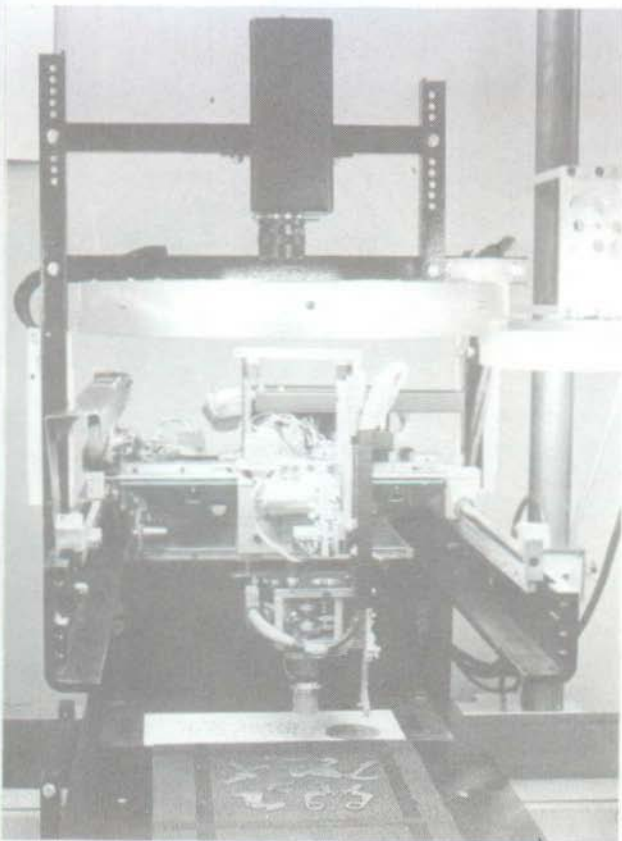
Après conversion analogique/numérique à 4 bit, des circuits TTL, en technologie intégrée à petite et moyenne échelle, réalisent un seuillage optionnel pour passer aux images binaires lorsque le contraste de la scène est très bon, un calcul de gradient travaillant sur deux lignes simultanément, et un deuxième seuillage programmable manuellement pour décider des éléments de contours significatifs. Enfin, l'image binaire des contours est stockée sur une mémoire MOS, où le miniordinateur vient la lire.

Une image prétraitee est disponible a chaque balayage de la cible vidicon, c'est-a-dire a la cadence standard de cinquante fois par seconde. L'image memorisee se compose de 256 lignes et de 256 colonnes. Elle occupe, centree, les deux tiers de l'image de television.

7.2.3 Déplacement des objets

Deux équipements complémentaires permettent le déplacement des objets. Un manipulateur cartésien, programmable, se distingue par ses caractéristiques dynamiques, et permet d'illustrer les résultats dus aux unités de traitement d'image. L'autre, une table x-y avec rotation, garantit des mesures précises de performances pour les nouveaux algorithmes. Les déplacements connus a priori par l'opérateur peuvent se comparer aux estimations résultant de l'analyse d'image.

- Manipulateur cartésien (Fig.7.6). Le manipulateur est construit de manière modulaire. Moteur a courant continu, réducteur, potentiomètre dans la boucle de contrôle et électronique de commande forment un ensemble identique pour chacun des axes x, y, z et pour la rotation θ d'une "pince" magnétique. Le tableau de la Fig.7.6 donne les caractéristiques principales.



	dynamique	précision
x	25 cm	± 1 mm
y	17 cm	± 1 mm
z	16 cm	± 1 mm
θ	340°	$\pm 3^\circ$

Fig.7.9 Manipulateur cartésien à 4 degrés de liberté. En plus de la translation selon trois axes, une rotation est possible sur la "pince" magnétique.

L'emploi de potentiomètres exclut une grande précision des mouvements. Les moteurs électriques, assez petits, limitent les accélérations possibles et conduisent donc à des cadences de travail assez lentes. Les cartes de commande enfin, supportent une électronique de contrôle assez primitive. Il en résulte des dérives thermiques et des oscillations passagères qui ne peuvent être négligées.

Avant d'annoncer que ce manipulateur a été développé en un temps extrêmement bref par l'institut de microtechnique de l'EPFL, il convient de dire qu'il satisfait pleinement son cahier des charges. Facile à commander, fiable, ne

nécessitant aucun entretien, il remplit parfaitement sa tâche de manipulateur de démonstration en laboratoire.

Les problèmes de trajectoire optimale et de comportement dynamique n'ont pas été abordés. Pour nos expériences, la programmation du manipulateur est très simple, d'autant qu'il est cartésien.

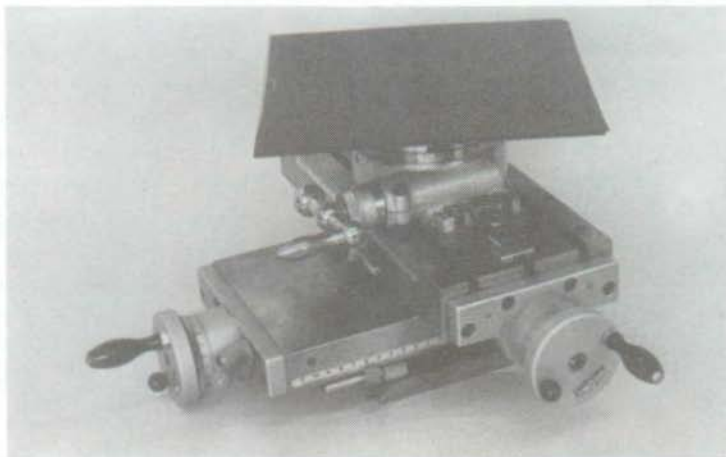


Fig.7.10 Table utilisée pour les mesures de précision

- Table x-y (Fig.7.10). La table de haute précision, à manivelles, s'utilise pour les mesures quantitatives de performances. Eu égard à la résolution des capteurs, les erreurs dues à cette table peuvent être considérées comme nulles.

7.3 Acquisition d'images

Une caméra de télévision observe des objets, sur un plan de travail. En temps réel, un processeur spécialisé numérise l'image perçue et en calcule les contours. Le résultat est stocké en mémoire.

7.3.1 Quantification de l'intensité lumineuse

L'intensité lumineuse I perçue d'un élément de surface sur la scène, dépend de trois grandeurs: La lumière incidente $\vec{E}$, l'orientation de la surface dans l'espace $\vec{n}$, et la réflectance propre r de cette surface. En chaque point on a une relation de la forme:

$$I = f(\vec{E}) \cdot g(\vec{n}) \cdot h(r) \quad (7.1)$$

La lumière incidente est en soi sans intérêt. L'orientation de la surface dans l'espace conduit par contre à une information précieuse. En pratique c'est souvent une légère différence d'orientation des surfaces qui permet de détecter les contours lorsque des objets de même réflectance se superposent (/18/ donne un bon résumé de l'influence des orientations entre capteurs, surfaces, et sources lumineuses). La réflectance enfin est une caractéristique essentielle des surfaces.

On prend souvent le logarithme de l'intensité perçue, afin de séparer par simple soustraction les trois composantes de l'équation 7.1. Cependant le modèle de l'équation 7.1 est quelque peu trop simple pour rendre compte de l'image après le capteur. Le capteur introduit toujours des distorsions. Notre montage expérimental ne permet pas l'estimation du logarithme.

La dynamique de l'intensité perçue est fondamentalement limitée, dans nos applications, par le tube à vidicon. Au-dessus d'une valeur d'intensité lumineuse $I_{\max}$, la définition spatiale de la cible diminue fortement. On parle alors de "blooming". On peut quelque peu augmenter la dynamique des gris perçus en utilisant des filtres optiques ou plus simplement un diaphragme à ouverture variable. Ceci n'est toutefois qu'un palliatif, car une augmentation de dynamique ne peut se faire qu'au détriment de la définition.

La définition des gris perçus est limitée par le bruit du capteur. Ce bruit est peu dépendant de l'intensité perçue. A cause de la relation 7.1, les petites différences en

réflectance et en orientation peuvent être mieux captées en présence d'une forte illumination. On a alors une définition des gris meilleure, mais une partie de la dynamique doit être sacrifiée. Le rapport entre bruit du capteur et dynamique du signal nous conduirait à l'utilisation de cinq à six bit. Nous n'avons utilisé que quatre bit dans nos expériences, et ceci constitue le goulot d'étranglement du flot d'information.

Dans un contexte industriel, lorsque les objets ne se superposent pas, le contraste entre objets et surface de travail est assez bon. L'intensité lumineuse peut alors être codée par une fonction binaire.

7.3.2 Echantillonnage de l'image

Une caméra de télévision à tube acquiert les images. L'intensité lumineuse est traduite par une charge électrique sur une cible analogique. La diffusion des charges entre parties élémentaires voisines de la cible limite la définition spatiale.

La définition n'est pas uniforme sur toute la surface réceptrice. Elle est en général suffisante pour représenter des images ayant jusqu'à 300 cycles en direction verticale et 400 cycles en horizontale, pour autant que la lumière ne soit pas trop intense.

L'échantillonnage est assuré dans le sens vertical par le balayage ligne par ligne standard en télévision. Dans le sens horizontal, c'est par échantillonnage temporel du signal vidéo que la discrétisation s'opère.

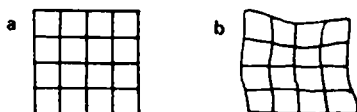


Fig.7.11 Distorsions géométriques dues à la caméra

Le balayage n'est pas linéaire dans le temps. Les intervalles entre échantillons voisins sur les lignes et sur les colonnes ne sont pas réguliers (Fig.7.8). De plus ces distorsions fluctuent en fonction de la température et du vieillissement. Le rapport en amplitude entre distorsion et champ de vue est de 2-3% pour C1 et C2, et de 1% pour C3.

Dans une certaine mesure, les distorsions géométriques peuvent être corrigées par une interpolation et un rééchantillonnage qui tiennent compte de la part systématique des distorsions. Cette amélioration se répercuterait directement sur l'estimation des caractéristiques, car c'est actuellement la source déterminante des perturbations.

7.3.3 Extraction des contours

Comme les surfaces de pièces estampées ont une texture très simple, toute l'information se concentre dans les contours. Les contours sont extraits en temps réel du signal vidéo numérisé, à l'aide de l'opérateur de Roberts (Fig.7.9).

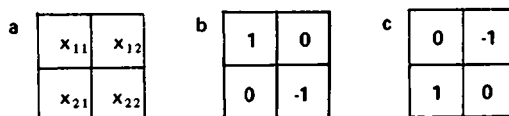


Fig.7.9 Opérateur général de filtrage (a). Opérateur de Roberts (b et c)

En utilisant les notations de la Fig.7.9, on peut représenter ainsi l'amplitude A_c et l'orientation θ_c du gradient associé à chaque élément de contour.

$$A_c = \sqrt{(x_{22} - x_{11})^2 + (x_{12} - x_{21})^2} \quad (7.2)$$

$$\theta_c = \text{Arc tg } (x_{12} - x_{21}) / (x_{22} - x_{11}) \quad (7.3)$$

Pour les images binaires, trois amplitudes du gradient sont possibles. Un gradient nul décrit des zones uniformes. Les autres valeurs désignent des différences de gris et

permettent d'obtenir des contours connectés au sens fort (une face commune entre chaque voisin).

En vue d'une réalisation par circuits spécialisés, l'opérateur de Roberts peut se simplifier. L'orientation θ_c est négligée et l'estimation d'amplitude se réduit à:

$$\tilde{A}_c = (|x_{22} - x_{11}| + |x_{12} - x_{21}|) / 2 \quad (7.4)$$

Un registre à décalage garde le signal vidéo correspondant à deux lignes courantes sur l'image /76/. Des circuits logiques détectent donc les éléments de contour dans l'image avec un léger retard (70ns).

Pour les images à niveaux de gris multiples, le même opérateur est utilisé. Ici toutefois les irrégularités de surface et le bruit de quantification causent des variations du gradient qui ne sont pas informatives, en-dessous d'une certaine amplitude. L'amplitude de décision est donc fixée à deux intervalles de quantification, pour la détection de nos contours.

Il serait raisonnable, pour les pièces qui se superposent, de sacrifier une partie de l'excellente définition spatiale de l'opérateur de Roberts, en le faisant précéder d'un filtre médian de dimension 2x2 ou 3x3 éléments d'image. Ceci entraînerait une réduction du bruit perçu et supprimerait une partie des "fausses alarmes".

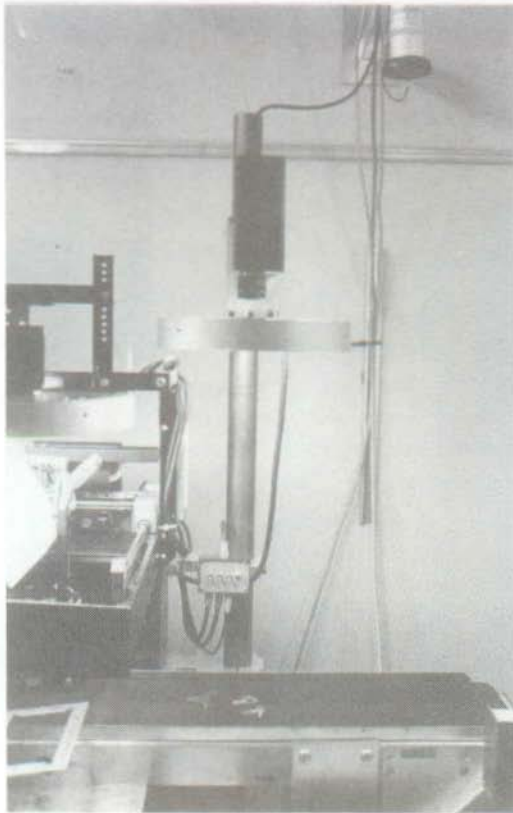


Fig.7.13 Eclairage dans l'axe du capteur

Pour les pièces qui se superposent, la différence de gris qui trahit la présence de bords a une composante aléatoire. Dans la mesure où l'illumination est dans l'axe du capteur (Fig.7.13), les différences sont purement dues à l'orientation et à la réflectance. Il arrive donc qu'un bord ne conduise à aucune variation de gris. Ou bien les méthodes d'analyse d'images s'accommodent de "trous" le long des contours (c'est l'option que nous avons choisie), ou alors un type d'éclairage différent doit être adopté /75/.

Le tube à vidicon a une bande passante limitée. Les fortes transitions de gris conduisent à des zones de transition plutôt qu'à des flancs raides. D'autre part, les ombres portées sont graduelles lorsque l'éclairage est diffus

(Fig.7.14). Les contours peuvent donc être épais, lorsque l'intensité lumineuse n'est pas représentée par un simple signal binaire.

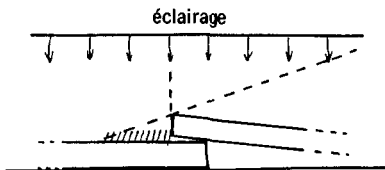


Fig.7.14 Eclairage diffus et ombres graduelles

7.4 Poursuite et squelettisation des contours

Un opérateur local d'estimation de gradient révèle des bords. Il faut ensuite segmenter l'image pour tenter d'isoler tour à tour chaque objet de la scène.

La segmentation de la scène est faite, pour nos applications en robotique, par poursuite des contours.

Le traitement des contours se heurte à des difficultés bien différentes suivant que la scène est traduite par une image binaire ou par une image à niveaux de gris multiples.

7.4.1 Images binaires

Dans les images binaires, l'opérateur de Roberts révèle presque toujours des contours parfaits: aucun trou, aucune bifurcation, épaisseur unité, structure bouclée. La poursuite des contours est alors triviale et se termine avec succès (Fig.7.15). La Fig.7.16 montre une situation rare, où la connexité de l'objet, et partant, celle du contour sont ambiguës.

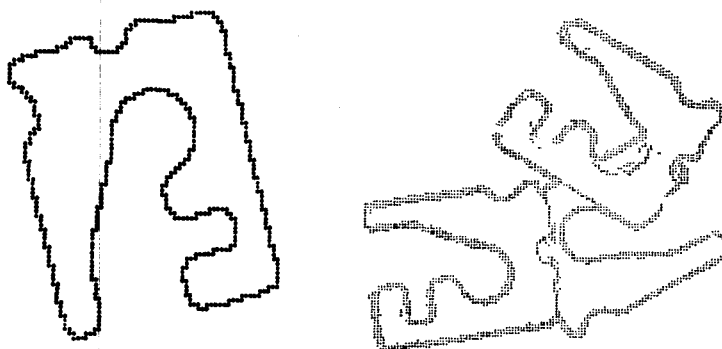


Fig.7.15 Application de l'opérateur de Roberts à des images binaire (gauche) et à tons de gris (droite).

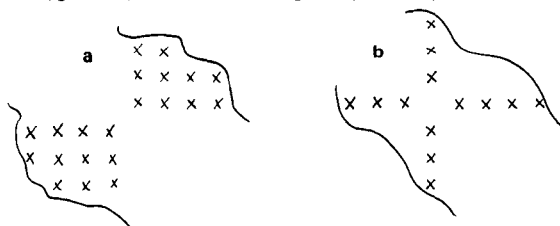


Fig.7.16 Situation où la connexité est ambiguë (a). Contours résultants (b).

7.4.2 Images à niveaux de gris multiples

Dans les images à nombreux niveaux de gris, les contours détectés sont épais. Ils ont aussi des trous. Ces deux défauts sont grandement réduits grâce à l'algorithme de suivi et de squelettisation des contours, décrit au paragraphe 3.2. Il a été utilisé dans sa version la plus simple. Chaque mouvement tente de réduire la distance entre les pointeurs. Même si celle-ci augmente, on ne revient pas en arrière et c'est l'autre pointeur qui à son tour essaie de se rapprocher.

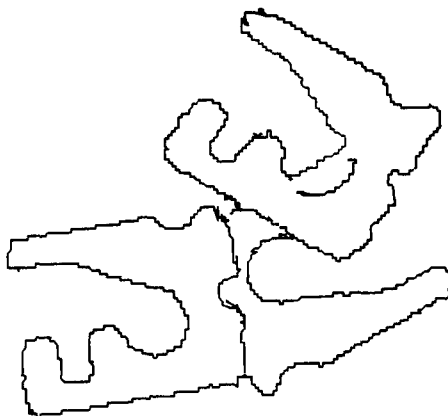


Fig.7.17 Squelettisation des contours de la Fig.7.15 par double poursuite. En pratique, un seul segment est traité, puis l'analyse continue

La Fig.7.17 montre comment l'algorithme de poursuite duale des contours agit sur nos images. Parmi les avantages de cette méthode séquentielle on peut noter:

- Simplicité. Une petite table contient l'adresse relative des voisins à tester.
- Vitesse. Deux tests en moyenne suffisent pour progresser d'un pas. Les cas les plus défavorable où plus d'une dizaine de tests sont nécessaires se rencontrent rarement. La probabilité que l'on rencontre un trou dans le voisinage d'un point est de l'ordre du pourcent.
- Robustesse. Même en présence de trous, jusqu'à deux intervalles de quantification, dans n'importe quelle direction, le suivi de contour ne s'interrompt pas.

Parmi les désavantages on doit noter:

- Placement des pointeurs. Le démarrage de la poursuite duale, sur un contour épais est parfois inadéquat (Fig.7.18). Plutôt que de rechercher un algorithme plus

robuste, il suffirait probablement d'"oublier" les tentatives qui avortent: Les pointeurs placés en A et en B (Fig.7.18) se rejoignent immédiatement lorsqu'ils se rapprochent et se distancent trop lorsqu'ils s'écartent.

- Sensibilité aux trous, dans les croisements. Les pointeurs définis au chapitre 3 accordent une grande priorité aux voisins immédiats. Dans un croisement, si l'un des contours est faiblement connecté aux autres, il peut être, à tort, ignoré (Fig.7.19). Ce dernier défaut est le plus gênant dans nos applications, car il conduit à des segments composés de parties qui ne correspondent pas toujours au même objet (Fig.7.20). On mentionnera plus loin quelles sont les conséquences de ce défaut.

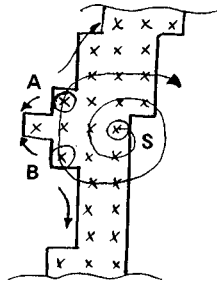


Fig.7.18 En cas de faux départ (A et B), les pointeurs se rejoignent ou divergent immédiatement. Une nouvelle tentative peut se faire, autour d'un autre point que S

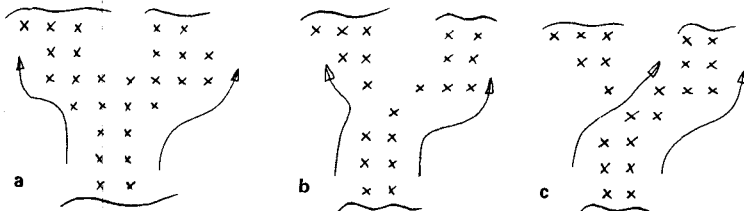


Fig.7.19 Un segment faiblement connecté est ignoré par les pointeurs (c).

Page inutilisée

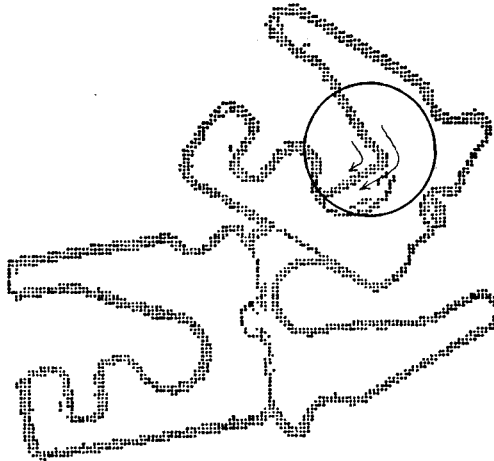


Fig.7.20 Une bifurcation n'a pas été détectée. On a donc un contour composite correspondant à deux objets différents

7.5 Lissage curviligne itératif

Les contours quantifiés doivent être lissés avant toute extraction de caractéristiques. Le bruit de quantification peut, à ce stade, être réduit par un simple filtrage passe-bas. Si au contraire, on le laisse se propager le long de la chaîne d'extraction des caractéristiques, il se transforme au travers de traitements non-linéaires et devient de plus en plus insaisissable.

Nous avons utilisé le filtrage curviligne itératif présenté au chapitre 4. Un filtre moyennneur de "durée" L est appliqué sur les contours pendant deux itérations. La durée est en général fixée à 9 intervalles de quantification. Pour des segments courts cependant, L diminue afin d'éviter les instabilités [45]. Ces conditions peuvent s'exprimer comme suit:

$$L = \begin{cases} 9 & , \text{ si } N \geq 27 \\ N/3 & , \text{ si } N < 27 \end{cases} \quad (7.5)$$

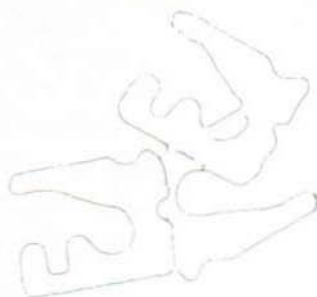


Fig.7.21 Lissage curviligne itératif des contours de la Fig.7.17

La Fig.7.21 montre les contours lissés dans une application pratique. La Fig.7.22 illustre l'effet de la taille de la fenêtre de filtrage et du traitement itératif. La convergence est très rapide, pour autant qu'on ne soit pas trop près des conditions limites de stabilité. Ceci est présenté en détail dans l'annexe 1.

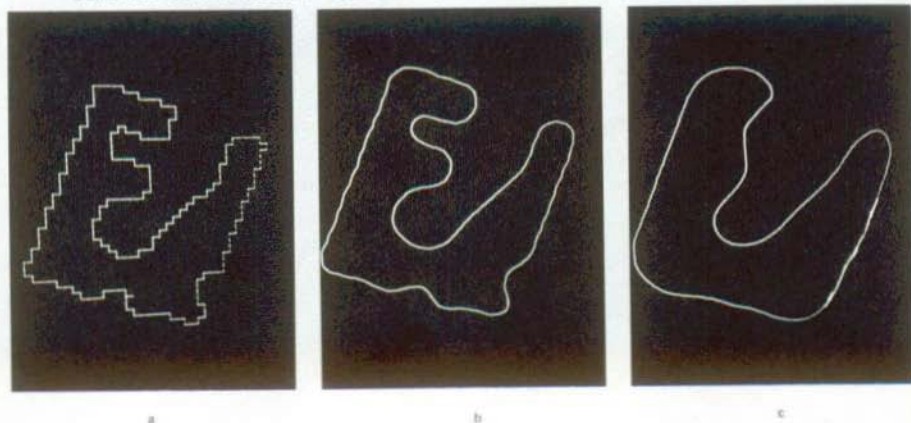


Fig.7.22 Un contour (a) est filtré "normalement" (b). En c le filtrage est exagéré, car ce qu'on perd en signal n'est pas gagné sur le bruit: Les distorsions du capteur sont alors plus significatives que le bruit de quantification

Le critère le plus adéquat pour déterminer les performances d'une technique de lissage, en reconnaissance des formes, c'est la dispersion des caractéristiques extraites lorsqu'une forme se déplace au hasard sur la grille de quantification.

La Fig.7.23 présente l'effet du filtrage curviligne itératif sur la caractéristique de longueur d'un périmètre. Il apparaît le résultat très intéressant que par rapport aux distorsions du capteur, le bruit de quantification après filtrage est négligeable. On a donc une remarquable insensibilité à la grille de quantification.

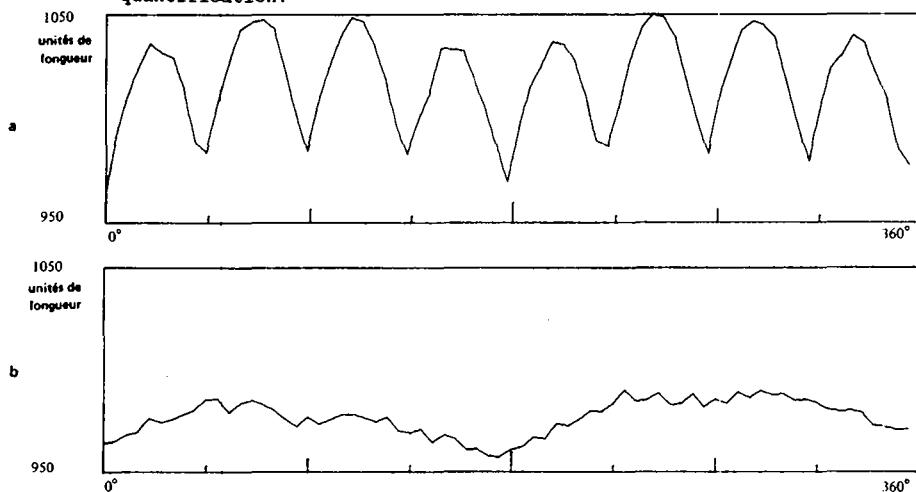


Fig.7.23 Estimation de longueur du périmètre. L'objet de la Fig.5.1 tourne de 360° . Son contour est moyenné sur 3 unités de longueur pour 2 itérations (a) ou sur 9 unités pour 2 itérations également (b). Il faut noter dans ce dernier cas les distorsions dues à la caméra

7.6 Mesures de caractéristiques

Autant que possible, nous avons extraits des caractéristiques globales pour identifier et localiser les objets. Des caractéristiques locales et structurelles ont permis l'estimation d'orientation et la reconnaissance de contours partiels.

7.6.1 Caractéristiques globales

L'estimation l de longueur (eq. 5.3) des contours lisses est affectée par les distorsions géométriques du capteur et par un blooming éventuel. L'erreur d'estimation n'excède pourtant pas 1%

(Fig.7.23). Pour comparaison, l'erreur d'estimation peut atteindre 20% sur les contours bruts.

L'estimation S de surface (eq. 5.7) conduit à des erreurs relatives (légèrement) inférieures à celles sur l , surtout si les objets sont compacts ($l^2/4\pi S \approx 1$). L'estimation S doit aussi se faire à partir du contour lissé.

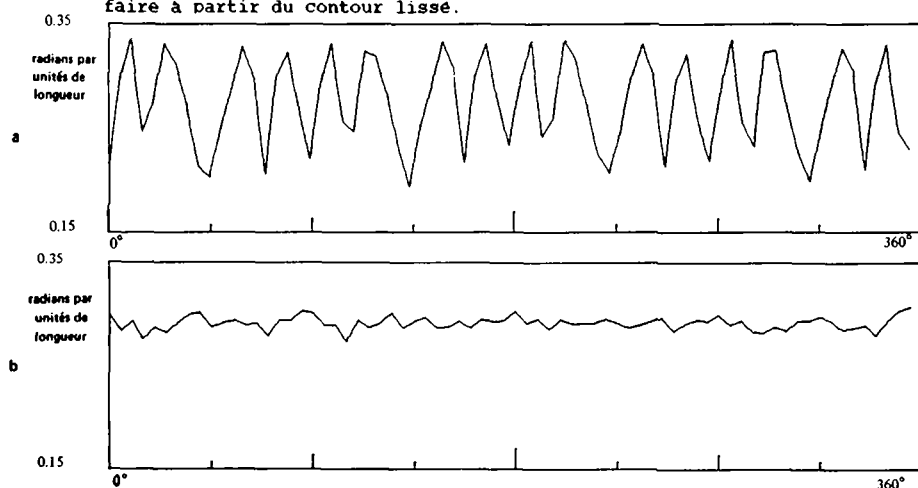


Fig.7.24 Estimation de l'écart-type de la courbure. L'objet de la Fig.5.1 tourne de 360° . Son contour est moyenné sur 3 unités de longueur (a) ou sur 9 unités (b).

L'imprécision sur l'écart-type de la courbure K est sensiblement plus grande que 1% (Fig.7.24). Ceci est dû aux nonlinéarités des fonctions trigonométriques et aux dérivations qui mènent à $K(s)$ (eq. 5.6). Rappelons que ce sont les composantes spectrales les moins favorables en rapport signal sur bruit que la dérivation accentue.

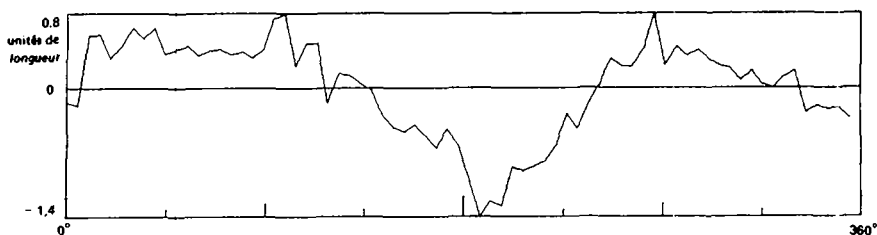


Fig. 7.25 Estimation de position. L'objet de la Fig. 5.1 tourne de 360° . Son contour est lissé sur 9 unités de longueur. On remarque les distorsions de la caméra

Le premier moment d'inertie (eq. 5.9) nous sert de repère de position. Des mesures systématiques en révèlent la même sensibilité aux distorsions du capteur que les caractéristiques précédentes: Un objet fait une rotation de 360 degrés autour d'un point quelconque et l'erreur d'estimation est définie comme l'écart entre la position estimée et le cercle sur lequel le moment d'inertie se déplace (Fig. 7.25).

7.6.2 Estimation de courbure

Lorsqu'un objet n'est pas entièrement visible, il ne peut être décrit que localement. Les coordonnées locales d'un contour dépendent de la position et de l'orientation d'un objet. La première dérivée n'est dépendante que de l'orientation. Et la deuxième dérivée possède les propriétés d'invariance à la translation et à la rotation (cf. le paragraphe 5.6 pour une discussion de l'intérêt de l'invariance).

A cause de la double dérivation qui y conduit, la courbure K (eq. 5.6) est particulièrement sensible au bruit (de quantification) du contour. Avec la connexité au sens fort (une face commune entre éléments de contour), K ne peut prendre que 4 valeurs, et ceci quelle que soit la résolution du capteur.

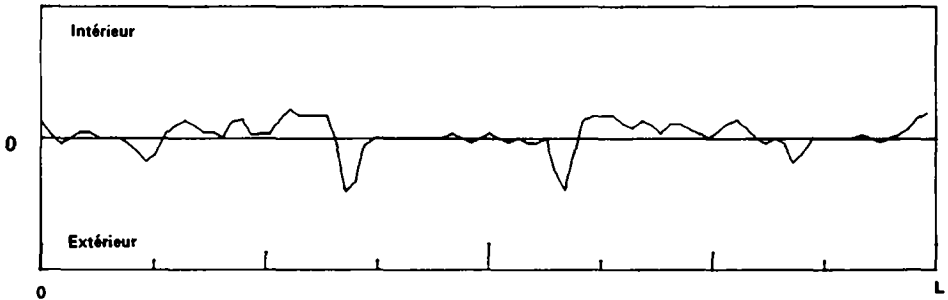
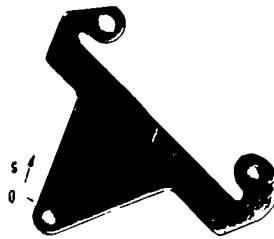


Fig.7.26 Fonction de courbure estimée à partir du contour lissé.

Un prétraitement des coordonnées qui réduise le bruit de quantification (cf. Chapitre 4) est ici très utile. Un lissage curviligne itératif "standard" des contours conduit à une estimation de courbure utilisable (Fig. 7.23). Nos valeurs typiques des paramètres (moyennage sur 9 points, 2 itérations, objets représentés par des zones de 64x64 points environ, dans des images de 256x256 points) conduisent à une "quantification" de la courbure à plus de 6 bit au lieu de 2-3 bit sur le contour brut. La courbure ne subit aucun traitement particulier mis-a-part le léger effet de filtrage passe-bas propre à l'estimation discrète de la dérivée.

Comme alternative à un lissage préalable, on a parfois essayé un moyennage local de la courbure pour en améliorer l'estimation /8,77,78,79/. Il apparaît toutefois deux désavantages. D'une part, le filtrage qui s'opère le long de la courbure ne peut pas facilement s'itérer: Lorsqu'on intègre la courbure adoucie pour reconstituer une courbe, l'erreur s'accumule (en général une courbe fermée traitée de la sorte ne se ferme plus). D'autre part, la quantification

quantification de la courbure est toujours beaucoup plus grossière que celle du contour.

7.6.3 Vecteur d'orientation

Pour nos applications, l'orientation d'une pièce est arbitrairement définie par deux repères P_1 , P_2 sur le contour. Ces repères sont caractérisés par une position (coordonnées x-y) et par un segment local de la fonction de courbure $K(s)$. Ils sont choisis (automatiquement) de manière à minimiser, pour l'estimation ultérieure d'orientation, la sensibilité au bruit.

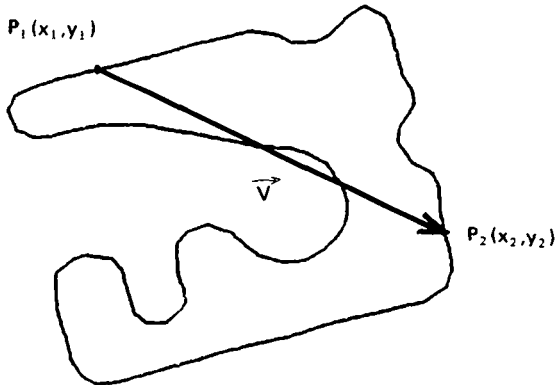


Fig.7.27 Origine (P_1) et extrémité (P_2) d'un "vecteur d'orientation" associé à un contour

Associations $P_1(x_1, y_1)$ à l'origine et $P_2(x_2, y_2)$ à l'extrémité d'un vecteur $\vec{v}$ (Fig.7.27). L'orientation de l'objet est alors définie, dans le plan de l'image par :

$$\theta = \begin{cases} \text{Arc tg } (y_2 - y_1) / (x_2 - x_1) & \text{si } x_2 - x_1 > 0 \\ \text{Arc tg } (y_2 - y_1) / (x_2 - x_1) + \pi & \text{si } x_2 - x_1 < 0 \end{cases} \quad (7.6)$$

Voyons maintenant l'estimation d'orientation θ . Supposons les repères P_1 et P_2 reconnus par leur fonction locale de courbure (paragraphe 7.7). Il se glisse toutefois une certaine erreur sur la position estimée. Cette erreur est

inférieure au demi-pas Δ_i d'échantillonnage de l'image, dans le sens normal au contour, et inférieure au demi-pas Δ_c d'échantillonnage du contour, dans le sens tangentiel (Fig. 7.28).

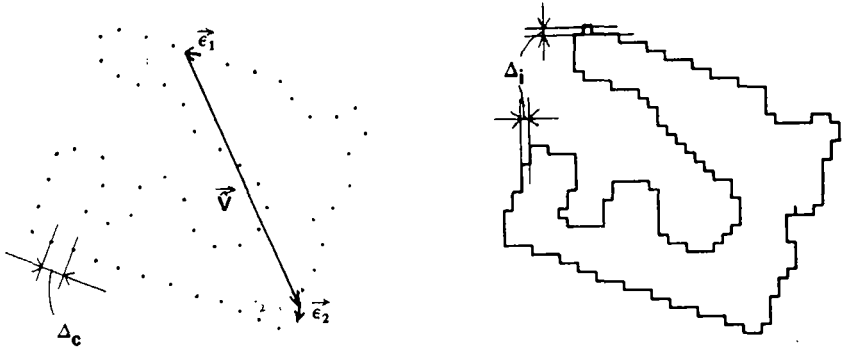


Fig. 7.28 Erreurs sur l'estimation du vecteur d'orientation, dues aux pas d'échantillonnage de l'image Δ_i et du contour Δ_c

L'erreur sur l'estimation de θ peut s'exprimer ainsi:

$$\epsilon_{\theta} = \text{Atg}(\|\vec{\epsilon}_1\| \sin\phi_1 + \|\vec{\epsilon}_2\| \sin\phi_2) / \|\vec{v}\| \quad (7.7)$$

Ce qui donne la borne supérieure:

$$\epsilon_{\theta} < \text{Atg}(\|\vec{\epsilon}_1\| + \|\vec{\epsilon}_2\|) / \|\vec{v}\| \quad (7.8)$$

De plus, $\vec{\epsilon}_1$ et $\vec{\epsilon}_2$ sont plutôt tangents au contour, parce que Δ_c est généralement plus grand que Δ_i après lissage (typiquement $\Delta_c = 4\Delta_i$). On en tire plusieurs conclusions sur le choix du vecteur d'orientation:

- $\vec{\epsilon}_1$ et $\vec{\epsilon}_2$ doivent être aussi petits que possible (!)
- Le module de $\vec{v}$ doit être maximal
- $\vec{v}$ doit être choisi de manière à ce qu'il soit plutôt collinéaire au contour dans le voisinage de P_1 et de P_2 . L'avantage accessoire de cette stratégie réside dans le fait que l'erreur en P_1 et en P_2 se répercute sur le

module de $\vec{v}$, ce qui permet un test ultérieur de qualité.

Un test complet d'optimalité du choix de $\vec{v}$ se justifie pour un objet entièrement visible, car ceci se fait une seule fois dans une phase d'apprentissage. Par contre lorsque seulement des segments de contours sont visibles, le compromis pour le choix de $\vec{v}$ entre rapidité, simplicité, coût et précision doit varier de cas en cas. Nous avons retenu le critère de la longueur de $\vec{v}$, en évitant toutefois de lui choisir comme extrémités les fins de contours à cause d'éventuels "effets de bord". Alors que ce critère conduirait à N^2 calculs de norme, où N est le nombre d'échantillons du segment, une simplification supplémentaire (Fig.7.29) réduit les calculs à $2 \cdot N$ comparaisons (les points de contacts entre le contour et les petits côtés du contour circonscrit définissent $\vec{v}$).

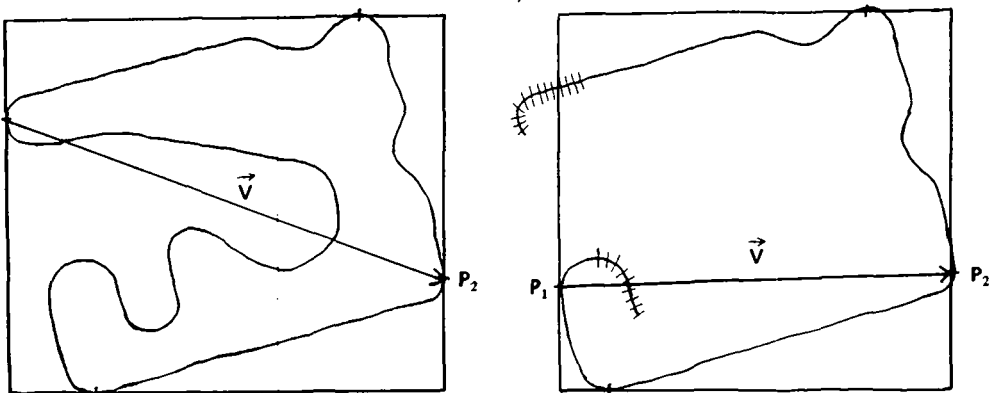


Fig.7.29 Les extrémités du vecteur d'orientation sont fixées par les petits côtés du rectangle circonscrit. Pour les contours ouverts, les fins de segments sont négligées

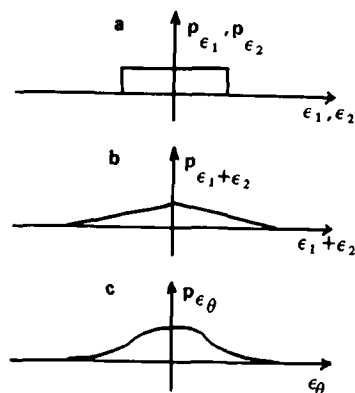


Fig.7.30 Modèle pour l'erreur en estimation d'orientation (voir texte)

La Fig.7.30 présente un modèle statistique de l'erreur ϵ_θ sur l'estimation d'orientation, en fonction des pas d'échantillonnage. Avec les densités de probabilités constantes pour ϵ_1 et ϵ_2 , on obtient pour la somme ϵ_s :

$$p_{\epsilon_s} = p_{\epsilon_1} * p_{\epsilon_2} \quad (7.9)$$

Comme l'erreur sur l'orientation est liée par l'équation:

$$\epsilon_\theta \leq \text{Atg} (\epsilon_s / v) \quad (7.10)$$

et que de plus la probabilité d'un événement est indépendante de sa représentation:

$$p_{\epsilon_s} d_{\epsilon_s} = p_{\epsilon_\theta} d_{\epsilon_\theta} \quad (7.11)$$

il résulte:

$$p_{\epsilon_\theta} = v / (1 + \epsilon_s^2) \quad (7.12)$$

La Fig.7.31 illustre des mesures systématiques faites pour une pièce entièrement visible.

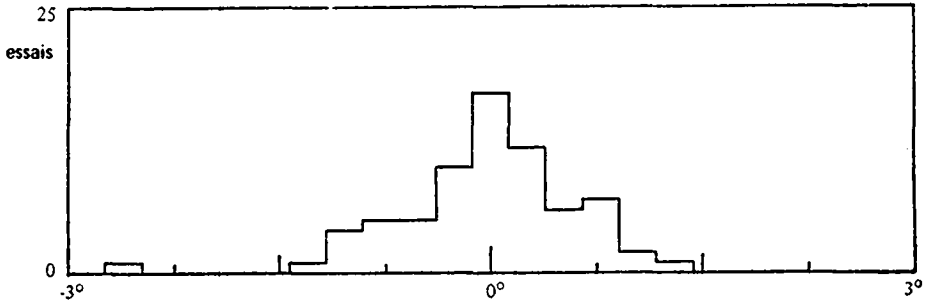


Fig.7.31 La pièce de la Fig.7.26 tourne de 360^0 , par pas de 5^0 .

On représente l'histogramme de l'erreur sur l'estimation d'orientation

7.7 Reconnaissance des pièces et des repères

Le chapitre 6 a présenté différentes méthodes d'identification de scènes visuelles. Dans un contexte de robotique industrielle, voyons comment elles se complètent.

7.7.1 Pièces complètement visibles

Pour autant qu'une pièce soit complètement visible, on peut en général en extraire des caractéristiques globales. La longueur du périmètre, la surface et l'écart-type de la courbure σ_K sont utilisés dans nos expériences. Ces caractéristiques engendrent un espace où les classes correspondant à nos pièces sont à symétrie approximativement sphérique ("ellipsoïdale"), et peuvent essentiellement se décrire par leur centre. La "city block distance" (cf. 6.2.2) est adéquate pour estimer la ressemblance entre scène et références. La différence en σ_K est toutefois pondérée pour tenir compte de sa plus grande dispersion (cf. 7.6). Le classificateur est très simple mais ne limite pas du tout les performances du système. Pour éviter de fausses détections, les classes doivent être éloignées (les pièces de références doivent être assez différentes) dans la mesure où les caractéristiques ont une grande dispersion.

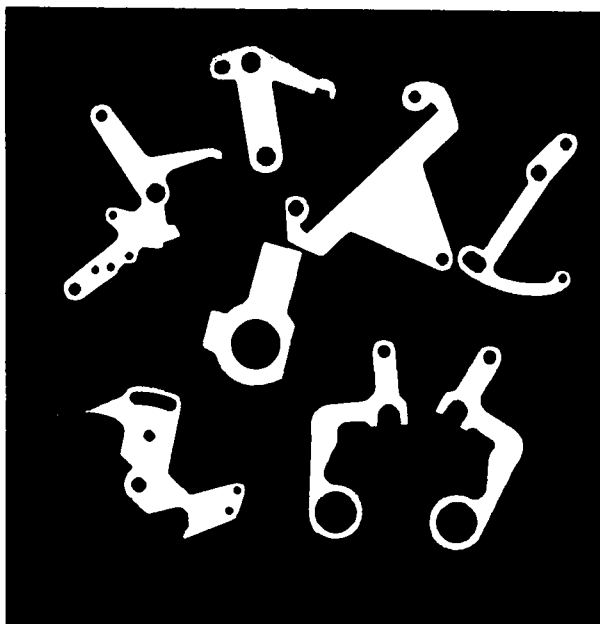


Fig.7.32 Pièces industrielles typiques utilisées pour nos expériences

La Fig.7.32 montre les pièces couramment utilisées dans nos expériences. Elles forment une configuration de classes très robuste, qui dans une large plage de variation de l'éclairage, de la distance focale de l'objectif, et du taux de distorsion de la caméra, ne conduit à aucune fausse détection et à très peu de rejet (1%).

Dans un contexte industriel, les résultats obtenus sont très bons puisque les fausses détections sont en général dangereuses, alors que le taux de rejet diminue simplement le débit moyen.

7.7.2 Contours partiels

Lorsqu'une pièce est partiellement cachée, ses caractéristiques globales ne sont plus accessibles. On a recours alors à des informations locales et en particulier, pour nos applications, à la fonction de courbure.

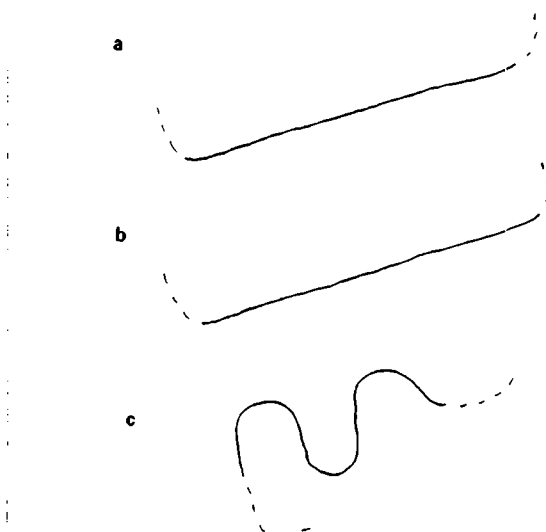


Fig.7.33 Le segment de contour a est tiré de la scène. On en estime la courbure, puis on identifie le segment correspondant de la référence par IDVA (b) et par corrélation (c). L'erreur dans ce dernier cas vient du fait qu'une droite a une courbure nulle

Dans un contexte industriel, les distorsions sont rarement admissibles. C'est pourquoi une mesure de similitude du type corrélatif est indiquée. Nous avons utilisé l'IDVA (cf. 6.1.2). La Fig.7.33 compare, pour un segment de contour tiré de l'image, les segments de la référence reconnus par l'intercorrélation (classique) et par l'IDVA. Clairement, la première méthode n'est pas utilisable pour nos applications.

Les calculs à faire sont nombreux. Pour la comparaison d'une référence et d'un segment de contour, représentés respectivement par N et M échantillons équidistants, $N \times M$ opérations (2 additions et une comparaison) sont nécessaires. Bien que le rapport SNR de l'IDVA en moyenne n'évolue pas, il a une dispersion plus faible lorsque M tend vers N . Dans ce sens, la qualité de la mesure de similitude augmente avec M .

Soit $K(s)$ la fonction de courbure de la référence, $K'(s)$ la courbure d'un segment de contour tiré de la scène et $n(s)$ un bruit additionnel à valeur moyenne nulle, de densité de probabilité $p_n(s)$.

La fonction $K(s)$ peut se modéliser comme une fonction aléatoire de densité de probabilité $p_K(s)$. Sa fonction d'autocorrélation en général décroît monotonement vers zéro lorsque le décalage augmente (ceci est vrai pour une demi-période). L'intercorrélation $R_{KK'}$, de K' et de K jouit donc de la même propriété, si le segment est assez grand.

La différence en valeur absolue entre $K(s+\tau)$ et $(K'(s)+n(s))$ peut se décomposer en quatre parties:

- Changement de signe de $K'(s)$ et de $n(s)$, d'où:

$$P_{-K'} = P_{K'}(-K') \quad (7.13)$$

$$P_{-n} = P_n(-n) \quad (7.14)$$

- Somme de K et de K' . Ici deux cas limites se présentent.

Avec τ_0 tel que $R_{KK'}(\tau_0) = 0$, on a indépendance de K et de K' , d'où:

$$P_{K-K'} = P_{K'} * P_{-K'} \quad (7.15)$$

pour τ_1 tel que $R_{KK'}(\tau_1) = 1$, on obtient:

$$P_{K-K'} = \delta(0) \quad (7.16)$$

- Addition de $-n(s)$ à la somme déjà formée, d'où s:

$$P_s = P_{K-K'} * P_{-n} \quad (7.17)$$

- Calcul de la valeur absolue:

$$p_{|s|} = \begin{cases} 0 & s < 0 \\ p_s(0) & \text{pour } s=0 \\ p_s(s) + p_s(-s) & s > 0 \end{cases} \quad (7.18)$$

L'intégrale de $p_{|s|}$ est approchée sur M termes par une sommation de termes discrets. En supposant l'indépendance des échantillons (ceci n'est pas tout-à-fait vrai, mais nous donne une borne supérieure), on en déduit que l'intégrale tend vers la valeur moyenne de $p_{|s|}$ suivant la loi des grands nombres. La variance de l'estimation faite sur M échantillons diminue proportionnellement à $1/M$ (Fig.7.31). L'algorithme adopté, de comparaison d'une référence avec une fonction partielle, est fondamentalement le même que celui présenté comme optimal dans /80/.

Dans la mesure où M est plus petit que N , en plus du fait que la probabilité d'erreur s'accroît, une partie de la référence n'est pas reconnue, et une autre mesure de similitude doit en principe se faire sur le segment restant.

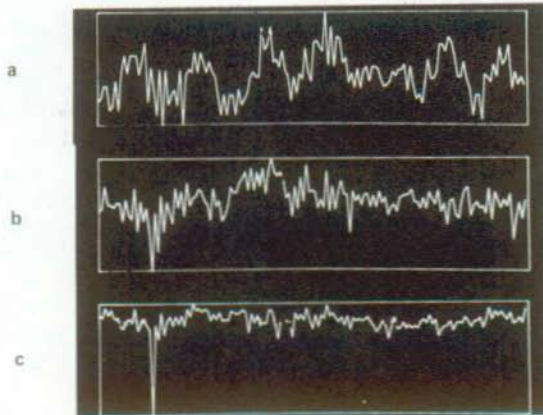


Fig.7.34 Influence de la longueur du segment tiré de la scène sur la comparaison avec la référence, en utilisant l'IDVA. La référence est faite de 147 échantillons. Le segment de la scène a 10 (a), 40 (b) ou 100 échantillons (c)

7.7.3 Repères d'orientation

Les repères d'orientation sont caractérisés par des segments de la fonction de courbure. Leur reconnaissance se confond donc avec l'identification des contours partiels.

Pourtant, la longueur du segment de courbure qui les caractérise peut en général se définir plus librement: Avec notre miniordinateur, il est considérablement plus rapide de reconnaître des segments courts. Un contrôle structural ultérieur détecte les éventuelles fausses reconnaissances et exige alors la comparaison sur un segment plus long, voire sur la fonction complète.

La longueur des segments couramment retenue pour nos expériences est d'environ 25%.

7.8 Test de superposition

Lorsque des objets se superposent, ils engendrent deux (super-)classes de formes sur l'image. La première classe regroupe les objets entièrement visibles, isolés ou au-dessus des autres; la seconde contient les combinaisons de segments de contours provenant de différentes pièces, le fond et les objets partiellement cachés.

L'algorithme de poursuite livre les contours par segments. Une étape de génération de courbes fermées (formes de l'une ou l'autre classe) par combinaison conduit à un grand nombre de formes, identifiables par un classificateur sur la base de caractéristiques globales. Adoptant cette approche, on se heurte toutefois à une double difficulté. D'une part on ne peut pas garantir que les pièces au-dessus du lot offrent un contour détectable, d'autre part les classes ne sont pas disjointes car on a un continuum de formes possibles en fonction du recouvrement.

Une alternative consiste à opérer une corrélation entre image (objet) de référence et scène. En se limitant à des scènes bidimensionnelles, on ne permet aux objets que trois degrés de liberté: x, y , et θ . L'intercorrélation doit donc être tridimensionnelle, semble-t-il. La référence est représentée par $M \times M$ points et M orientations sont possibles (car la définition

angulaire n'est pas indépendante de la résolution fixée par l'échantillonnage), et la scène se traduit par $N \times N$ points. M_e est le nombre d'éléments de contour de la référence et M_o le nombre d'occurrences de ce type d'objet dans la scène.

Les trois degrés de liberté d'un objet dans le plan sont réduits trivialement à zéro lorsqu'on intègre l'information de translation et de rotation à l'image. On a alors une représentation de la scène invariante à x , y , et θ .

En d'autres termes, seul le maximum de la corrélation est important, et on l'obtient pour un décalage selon x , y et θ , qui est connu à ce stade de l'analyse. Le calcul est alors praticable puisqu'on a un nombre de multiplications relativement faible:

$$Q_1 = M_o \times M_e \quad (7.19)$$

Par une application naïve de la corrélation tridimensionnelle on aurait au contraire:

$$Q_2 = M^3 \cdot N^2 \quad (7.20)$$

Le gain est d'autant plus remarquable qu'on a généralement la relation suivante:

$$M_o < M_e < M < N \quad (7.21)$$

7.9 Contrôle et comparaison de structures

Différents "tests de qualité" commandent le rejet éventuel des pièces et limitent ainsi le nombre des fausses détections. Ils sont répartis le long de la chaîne de traitement pour éviter aussi tôt que possible les calculs inutiles.

- Lors de la classification (si le contour est fermé), la distance entre la scène et la référence la plus proche donne un critère évident.

- Lors de la comparaison des courbures, l'indice de corrélation doit être proche de "un". Un test similaire se fait si on utilise l'IDVA.
- Les repères d'orientation sont séparés par une distance connue a priori (corde ou longueur d'arc). Une mauvaise détection de l'un ou l'autre repère se répercute en général sur la distance entre eux.
- Enfin, après reconnaissance d'un segment de contour, après estimation de position et d'orientation, la référence traduite et tournée se corréle avec la scène, il en résulte un indice qu'on confronte à un seuil.

Un éventuel rejet peut provenir d'une vue mal prise, d'un objet inconnu, ou d'une pièce connue qui dépasse les tolérances.

Si les objets se superposent, un rejet peut également intervenir lorsque le manque de contraste ou une pièce superposée cachent une partie déterminée du contour (typiquement 5% ou plus). On peut donc, dans une certaine mesure troquer de la fausse alarme contre de la non-détection. En robotique industrielle, la non-détection rallonge simplement le temps de traitement moyen. Et souvent le temps de traitement de l'équipement de reconnaissance des formes n'est pas celui qui détermine la cadence du poste de travail. Les fausses alarmes par contre sont généralement catastrophiques, imposant un contrôle ultérieur à défaut duquel blocage du poste de travail ou même dégâts aux machines-outils peuvent survenir.

8. PROBLEMES OUVERTS

Les résultats de nos recherches ont été publiés pour la plupart (cf. en particulier /45,58,74,75,82/) et même en partie illustrés par un film largement diffusé (Cambridge/Boston, Kingston R.I., Lausanne, Lyon, Paris, Milan, Wetzlar, Zürich). Ils apportent leur contribution au traitement d'images et à l'automatisation industrielle. Ces disciplines souffrent encore d'énormes lacunes, et dans la direction particulière jalonnée par nos travaux, on peut en noter plusieurs maintenant à combler:

- Intégration au traitement présent de l'information de texture. La texture est souvent associée à des régions fermées dans l'image. Une difficulté supplémentaire vient du changement progressif de la texture à l'intérieur de régions non-fermées (contours ouverts), qu'on rencontre dans la plupart des images à tons de gris. Les contours devraient être représentés par 4 fonctions paramétriques: $x(s)$, $y(s)$ pour les coordonnées et $t_g(s)$, $t_d(s)$ pour la texture de part et d'autre du contour (Fig.8.1a-d). Cette solution doit s'avérer extrêmement intéressante pour la représentation synthétique de scènes, pour la combinaison significative des segments de contours disjoints et, en dernier ressort, pour l'analyse des scènes complexes. De plus, ceci se rapproche de ce que font l'oeil et le cortex humains: Certaines expériences montrent que l'information de couleur n'est perçue qu'au voisinage des contours, et qu'ensuite, le cortex l'étend aux surfaces /81/. L'interpolation de la texture entre contours ouverts est encore un problème sans solution (Fig.8.1e).
- Généralisation pour les images à plusieurs niveaux de gris de la notion de connexité. Il en résulterait une poursuite et une squelettisation optimale des contours dans les images à tons de gris.
- Définition de bonnes caractéristiques "régionales" le long des contours pour accélérer la reconnaissance de scènes complexes. Les paramètres locaux (par ex. courbure en chaque point) entraînent un grand nombre de calculs, alors que les globaux (par ex. longueur de périmètre) sont rarement disponibles. Un ensemble de ces

informations régionales formerait le symbole de départ pour une méthode structurale d'identification. Bien que certains auteurs utilisent déjà les angles le long des contours comme caractéristiques régionales, un gros effort de recherches est encore nécessaire pour trouver un mode de définition de telles caractéristiques, qui soit peu sensible au bruit additionnel et aux distorsions, et qui conduise ainsi toujours au même ensemble de caractéristiques pour un contour donné. Le problème peut en particulier se formuler ainsi: Selon quels critères la fonction de courbure d'un contour peut-elle se découper en segments?

- Identification et localisation de scènes vraiment tridimensionnelles (objets à six degrés de liberté!). Différentes stratégies s'esquissent, utilisant des éclairages par réseaux de lignes ou de points, des prises de vue multiples ou plusieurs capteurs, mais les solutions actuelles sont partielles et ne concernent que quelques problèmes très particuliers. Pour une solution aussi complète que pour les scènes bidimensionnelles, il faudra probablement attendre que se répande l'usage de capteurs tridimensionnels /14/, de très grandes mémoires, et des processeurs encore plus rapides qu'aujourd'hui.

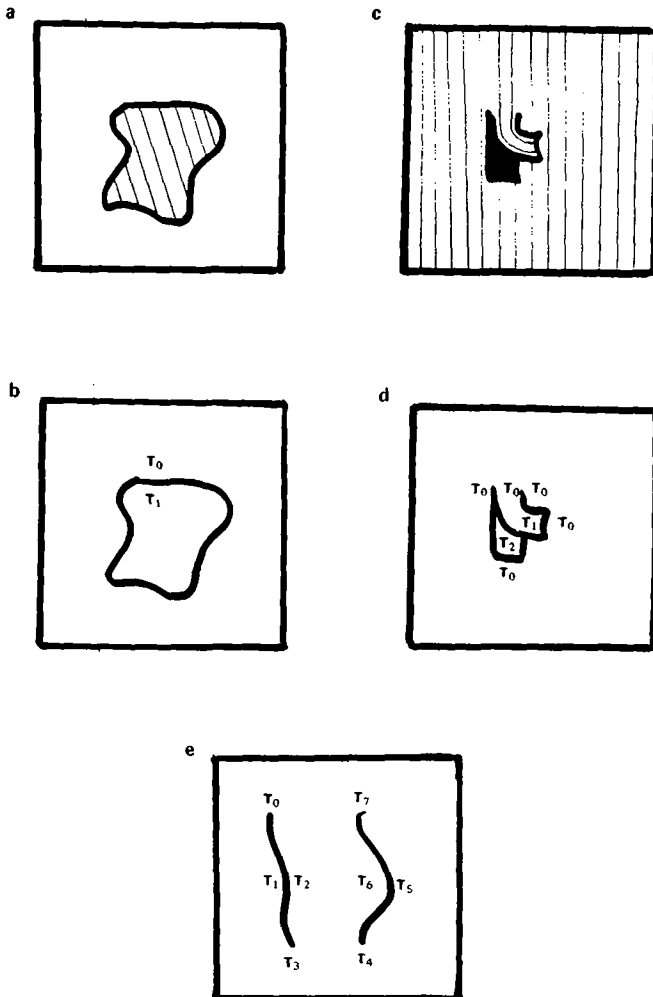


Fig.8.1 Lorsque une scène est formée de régions uniformes (a), l'information de texture peut être associée aux régions aussi bien qu'aux contours (b). Lorsque la texture varie dans la scène (c), on peut la décrire en chaque point le long des deux côtés des contours (d). Hors des contours, la texture varie peu, et il semble possible dans une certaine mesure de la retrouver par interpolation (e)

REFERENCES

- 1 G.N.Saridis, "Toward the Realization of Intelligent Controls", Proc. of the IEEE, vol 67, No. 8, Aug. 79.
- 2 G.Giralt, M.Ghallab, P.Stuck, "Object Identification and Sorting with an Optimal Sequential Pattern Recognition Method", 9th Intern. Symp. on Industr. Robots, Whashington, March 13-15, 1979, pp379-389.
- 3 M.Yashida, S.Tsuji, " Industrial Computer Vision in Japan", IEEE, Computer, Special issue on Machine Vision, May 1980, pp50-63.
- 4 R.Kelley, J.Birk, D.Duncan, H.Martins, R.Tella, "A Robot which Feeds Workpieces Directly from Bins into Machines", 9th Int. Symp. on Industr. Robots, Whashington, March 13-15, 1979, pp339-355.
- 5 Business Week, "Robots join the labor force", USA, May 1980
- 6 H.G.Barrow and R.C.Popplestone, "Relational Descriptions in Picture Processing", Artificial Intelligence Vol. 2, pp377-396, 1971.
- 7 W.A.Perkins, "A Model-Based Vision System for Industrial Parts", IEEE Trans. on Comp., Vol c-27, 2, pp126-143, 1978.
- 8 J.E.Bowie, "Syntactic Descriptions of Biological Shape", Ph. D. Thesis, Mass. Inst. of Technology, 1973.
- 9 R.J.Woodham, "Photometric Stereo: A Reflectance Map Technique for Determining Surface Orientation from Image Intensity", Proc. Soc. of Photo-Optical Instr. Eng., vol. 155, San Diego,, Aug. 30-31, 1978, pp136-143.
- 10 M.Bernasconi, R.Delazer, M.Gini, "Precision Measurement By Stereo Vision System", 10th Intern. Symp. on Industr. Robots, Milan, March 1980, pp349-360.
- 11 J.M.Zurcher, "Etude des capteurs d'images électro-optiques", Rapport LTS 79.03, Ecole Polytechn. Fédérale de Lausanne, 1979, pp64

- 12 W.W.Marden, R.L.Steinberger, P.B.Bracco, "Pulsed illumination, closed circuit television system for real-time viewing of Unsteady (>1 micros) events", Rev. Sci. Instrum., USA, vol. 49, no.10, Oct. 1978, pp1392-8.
- 13 Y.Nitta, "Visual Identification and Sorting with TV-Camera applied to automated Inspection Apparatus", 10th Int. Symp. on Industr. Robots, Milan, Italy, March 1980, pp141-152.
- 14 R.O.Duda, D.Nitzan, P.Barret, "Use of Range and Reflectance Data to find Planar Surface Regions", IEEE Trans. PAMI-1, No.3, July 79, pp259-271.
- 15 M.R.Ward, L.Rossol, S.W.Holland, R.Dewar, "Consight: A Practical Vision-Based Robot Guidance system", 9th Int. Symp. on Industr. Robots", Whashington, March 13-15, 1979, pp195-212.
- 16 M.Oshima, Y.Shirai, "A Scene Description Method using Three-Dimensional Information", Agency of Industrial Science and Technology, PIPS-R-13, March 1977, pp32-47.
- 17 G.J.VanderBrug, J.S.Albus, E.Barkmeyer, "A Vision System for Real-Time Control of Robots", 9th Int. Symp. on Industr. Robots", Whashington, March 13-15, 1979, pp213-231.
- 18 W.K.Pratt, "Digital Image Processing", J.Wiley, New-York, 1978, pp780.
- 19 D.P.Petersen and D.Middleton, "Sampling and reconstruction of wave number limited function in N-dimensional Euclidian spaces", Info. Control, Vol. 5, pp279-323, Dec. 1962
- 20 M.J.E.Golay, "Hexagonal Parallel Pattern Transformation", IEEE Trans. on Comp., vol C-18, pp733-740, 1969.
- 21 R.M.Mersereau, "The Processing of Hexagonally sampled Two-Dimensional Signals", Proc. of the IEEE, vol 67, 6, pp930-949, Jun. 1979.

- 22 G.B.Shaw, "Local and Regional Edge Detectors: Some Comparisons",
Comp. Graphics and Im. Proc., 9, 1979, pp135-149.
- 23 G.R.Nudd, P.A.Nygaard, G.D.Thurmond, S.D.Pouse, "A Charge-Coupled
Device Image Processor for Smart Sensor Applications", Proc. Soc.
of Photo-Optical Instrum. Eng., Vol 155, Aug.30-31, 1978, San Diego.
- 24 L.Mero, Z.Vassy, "A Simplified and Fast Version of the Hueckel
operator for finding optimal edges in Pictures", Proc. 4th Intl
Conf. on Artif. Intell., Tbilisi, USSR, Sept. 75, pp650-655.
- 25 T.S.Huang, G.Y.Yang, G.Y.Tang, "A Fast Two Dimensional Median
Filtering Algorithm", IEEE Trans. on Acoustics, Speech and Signal
Processing, vol. ASSP-27, Febr. 79, pp13-18.
- 26 B.Justusson, "Median Filtering: Statistical Properties", Mathematical
Statistics, Royal Inst. of Techn., Stockholm, Aug. 79, pp65.
- 27 H.Blum, "A transformation for extracting new descriptors of shape",
Models for the perception of speech and visual form, W.Wathen-Dunn
ed., MIT Press, 1967, pp362-380
- 28 C.Arcelli, L.Cordelli, S.Levialdi, "Parallel thinning of binary
pictures", Electron. Lett. (GB), 11(7), 1975, pp148-149
- 29 H.Enomoto, N.Yonezaki, K.Nitta, "Schematic representation and
perception of curved surface and picture", Dept. of Comp. Science,
Tokyo Inst. of Techn., 2-12-1, pp31
- 30 V.Goetcherian, "From binary to grey-tone image processing using fuzzy
logic concepts", Pattern Recognition, vol. 12, Pergamon Press, GB,
1980, pp7-15
- 31 W.Frei, C.C.Chen, "Fast boundary detection: a generalization and a
new algorithm", IEEE Trans. Comput. (USA), vol c-26, no.10, 1977,
pp988-998
- 32 G.Robinson, "Edge detection by compass gradient masks", Comp. Graph.
and Im. Proc. 6, 1977, pp492-501

- 33 R.M.Haralick, "Edge and region analysis for digital image data",
Comp. Graph. and Im. Proc. 1, vol.12, Jan. 1980
- 34 M.Basseville, B.Espiau, J.Gasnier, "Détection séquentielle de
contours pour capteurs d'images linéaires et applications", 2ième
Congrès APCET-IRIA, Toulouse, 12-14 Sept. 1979, pp233-241
- 35 R.M.Haralick, K.Shannumgan, I.Dinstein, "Texture features for image
classification", IEEE Trans. on syst. man and cybern., vol SMC-3,
Nov. 1973, pp610-621
- 36 J.Sklansky, "Image segmentation and feature extraction", IEEE Trans.
on Syst. Man and Cyb. SMC-8, No. 4, Apr. 1978, pp237-247
- 37 F.Veillon, "Une methode en 1 passage de calcul simultané de propriété
morphométriques d'objets dans des images", Congrès APCET-IRIA 1978,
Tome 1, pp411-420
- 38 D.L.Milgram, "Constructing trees for region description", Comput.
Graphics and Im. Proc., vol.11, No.1, Sept. 1979
- 39 A.K.Aggrawala, A.V.Kulkarni, "A sequential approach to the extraction
of shape features", Comput. Graphics and Im. Proc., 1977, pp538-557
- 40 F.Attneave, "Some informational aspects of visual perception",
Psychology Review, No.61, 1954, pp183-193
- 41 D.Profitt, D.Rosen, "Metrication errors and coding efficiency of
chain-encoding schemes for the representation of lines and edges",
Comput. Graphics and Im. Proc. 10, 1979, pp318-332
- 42 H.Freeman, "Boundary encoding and processing", Picture Processing and
Psychopictonics, B.S.Lipkin and A.Rosenfeld (Eds), Academic Press,
New-York, 1970
- 43 J.D.Dessimoz, "Specialised edge-trackers for contour extraction and
line-thinning", Ecole Polytechn. Fédérale de Lausanne, Rapport
LTS-79.05, 1979, pp65

- 44 M.Berthod, J.P.Maroy, "Learning and syntactic recognition of symbols drawn on a graphic tablet", *Comp. Graph. and Im. Proc.*, vol. 9, pp166-182, 1979.
- 45 J.D.Dessimoz, "Curve smoothing for improved feature extraction from digitized pictures", *Signal Processing* 1, No.3, July 1979, pp205-210
- 46 J.Max, "Quantizing for Minimum Distortion", *IRE Trans. Inf. Theory*, Vol. IT-6, pp7-12, Mar. 1960.
- 47 J.Oswald, "Fonctions à accroissement borné: Application à l'échantillonnage, à la quantification et à l'interpolation des signaux", *Ann. Télécomm.* 30, No.3-4, 1975, pp1-11
- 48 W.R.Bennett, "Spectrum of quantized signals", *Bell Syst. Techn. J.* Vol. 27, pp446072, Jul. 1948.
- 49 H.Steinhaus, "Length, Shape and Area", *Colloq. Math.* 3, 1, 1954.
IEEE Trans. on Comp., Vol C-26, No. 8, Aug. 1977.
- 50 B.Mandelbrot, "Fractals: Form, Chance, Dimension", Freeman, San Francisco, 1977.
- 51 I.T.Young, et al., "An Analysis Technique for Biological Shape - I", *Info. and Control* 25, pp357-370 1974.
- 52 T.Pavlidis, "Polygonal Approximations by Newton's Method" *IEEE Trans. on Comp.*, Vol C-26, No. 8, Aug. 1977.
- 53 J.E.Midgley, "Isotropic Four-Point Interpolation", *Comp. Graphics and Image Proc.* 11, pp192-196, 1979.
- 54 W.S.Rutkowski, "Shape Completion", *Computer Graphics and Image Proc.* 9, pp89-101, 1979.
- 55 P.L.Bookstein, "Fitting Conic Sections to Scattered Data", *Comput. Graphics and Im. Proc.* 9, pp56-71, 1979.

- 56 A.J.Jerri, "The Shannon Sampling Theorem - Its Various Extensions and Applications: A Tutorial Review" Proc. of IEEE, Vol. 65, No. 11, pp1565-1595, Nov. 1977.
- 57 P.Kammenos, "Performances of Polar Coding of Visual Localisation of Planar Objects", Proc. 8th Intern. Symp. on Indust. Robot, Stuttgart, Germany, pp143-154, May.-Jun. 1978.
- 58 J.D.Dessimoz, "Visual Identification and Location in a Multi-object Environment, by Contour Tracking and Curvature Description", Proc. 8th Intern. Symp. on Industr. Robots, Stuttgart, Germany, pp764-777, May-Jun. 1978.
- 59 E.Persoon and K.S.Fu, "Shape Discrimination Using Fourier Descriptors", IEEE Trans. on Sys., Man, and Cyb., Vol SMC-7, 3, Mar. 1977.
- 60 P.C.A.Groen and P.W.Verbeek, "Freeman-code Probabilities of Object Boundary Quantized Contours" Comp. Graphics and Image Proc. 7, pp391-402, 1978.
- 61 S.A.Dudani, K.J.Breeding, R.B.MacGhee, "Aircraft identification by moment invariants", IEEE Trans. on Computers, Vol.C-26, No.1, Jan.1977, pp39-45
- 62 S.Gaglio, G.Marino, P.Morasso, V.Tagliasco, "A linguistic approach to the measurement of 3-D motions of kinematic chains", 10th Intern. Symp. on Industr. Robots, March 5-7, 1980, Milan, Italy, pp387-398
- 63 H.C.Andrews, C.L.Patterson, "Outer product expansions and their uses in digital image processing", IEEE Trans. on Computers, vol.C-25, No.2, Febr. 1976, pp140-148
- 64 "Hitachi Hand-Eye System", Hitachi Review, vol.22, No.9, pp362-365
- 65 H.Freeman, "Shape Description via the use of critical points", Proc. of the IEEE Conf. on Pattern Recogn. and Im. Proc., New-York, June 6-8 1977, pp168-174

- 66 T.Agui, M.Nakajima, "Picture processing by shape contour functions", Tokyo Inst. of Technology, pp22, Communication privée
- 67 R.W.Ehrich, J.P.Foith, "Representation of random waveforms by relational trees", IEEE Trans. on Comp. C-25, No.7, July 1976, pp725-736
- 68 V.V.Mottl', I.B.Muchnik, "Linguistic Analysy of Experimental Curves", Proc. of the IEEE, vol.67, No.5, May 1979, pp714-736
- 69 K.S.Fu, "Sequential Methods in Pattern Recognition and Machine Learning", Academic Press, New-York, 1968
- 70 T.W.Anderson, "An introduction to multivariate statistical analysis", Wiley, New-York, Chap.6, 1958
- 71 Y.Z.Tsyppkin, "Foundations of the theory of learning systems", Academic Press, New-York, 1973
- 72 R.Deutsch, "System analysis techniques", Prentice-Hall, Englewood Cliffs, 1969, pp464
- 73 N.Otsu, "A threshold selection method from grey-level histograms", IEEE Trans. on Systems, Man, and Cybern., Vol.SMC-9, No. 1, Jan. 1979, pp62-66
- 74 J.D.Dessimoz, "Identification et localisation visuelle d'objets multiples par poursuite de contours et codage de courbure", Comptes-rendus des Journées de Microtechnique, Ecole Polytechn. Fédérale de Lausanne, 1978, pp194-204
- 75 J.D.Dessimoz, M.Kunt, G.H.Granlund, J.M.Zurcher, "Recognition and handling of overlapping industrial parts", 9th Intern. Symp. on Industr. Robots, Whashington, March 13-15, 1979, pp357-366
- 76 J.M.Zurcher, "Extraction de contours en traitement électronique des images II: Processeur spécialisé pour signal video", Bull. de l'ASE, Zürich, No.11, Juin 1979, pp532-536

- 77 J.R.Bennet, J.S.McDonald, "On the measurement of curvature in a quantized environment", IEEE Trans. on Computers, vol.C-24, No.8, Aug. 1975, pp803-820
- 78 J.R.Bennet, "On the measurement of the curvature of the boundaries of two-dimensional quantized shapes", Ph.D. Dissertation, Dept. Elec. Eng. Univ. of British Columbia, Vancouver B.C., 1972
- 79 T.J.Ellis, D.Profitt, D.Rosen, W.Rutkowski, "Measurement of the lengths of digitized curved lines", Comp. Graphics and Im. Proc. 10, 1979, pp333-347
- 80 J.K.Dixon, "Pattern Recognition with partly missing data", IEEE Trans. on Syst., Man, and Cybern., vol.SMC-9, No.10, Oct. 1979
- 81 A.L.Gilchrist, "The Perception of surface blacks and whites", Scientific American, vol.240, No.3, March 1979, pp112-124
- 82 J.D.Dessimoz, Logiciel spécialisé pour la reconnaissance et la localisation de pièces industrielles plates", Rapport LTS-79.02, Ecole polytechn. fédérale de Lausanne, 1979
- 83 S.Y.Lu, "A tree-to-tree distance and its application to cluster analysis", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.1, No.2, Apr. 1979, pp219-224

ANNEXE 1

CURVE SMOOTHING FOR IMPROVED FEATURE EXTRACTION FROM DIGITIZED PICTURES*

Jean-Daniel DESSIMOZ

Laboratoire de Traitement des Signaux, Ecole Polytechnique Fédérale de Lausanne, 16 Chemin de Bellerive, Switzerland

Received 6 November 1978

Revised 20 February 1979 and 30 March 1979

Abstract. Curves can be expressed by parametrized functions. If a curve is represented by a sampled picture, its parametrized functions appear quantized. The noise induced by quantization can drastically be reduced by an iterative curvilinear filtering of the parametrized functions. At each iteration, this filtering processes the curve coordinates and a better arc length is estimated.

Zusammenfassung. Kurven können durch parametrische Funktionen dargestellt werden. Wenn eine Kurve durch ein abgetastetes Bild dargestellt wird, erscheinen ihre parametrische Funktionen quantisiert. Der Quantisierungsfehler kann durch eine iterative Filterung über die Bogenlänge beschränkt werden. Bei jeder Iteration verarbeitet diese Filterung die Kurvenkoordinaten und erlaubt eine präzisere Abschätzung der Bogenlänge.

Résumé. Les courbes peuvent s'exprimer par des fonctions paramétrées. Lorsqu'une courbe est décrite par une image échantillonnée, ses fonctions paramétrées apparaissent quantifiées. Le bruit introduit par la quantification peut être réduit par un filtrage curviligne itératif des fonctions paramétrées. A chaque itération, le filtre traite les coordonnées de la courbe et l'on en tire une meilleure estimation de la longueur d'arc, ce qui permet un meilleur filtrage.

Keywords. Feature extraction, digitized pictures, quantization noise, parametrized curves, coordinate filtering.

1. Introduction

Boundaries generally convey most of the information of an image. Therefore in our applications, pattern recognition is performed on pictures after extraction or enhancement of the edges.

Arc length, tangent angles, curvatures, area of blobs or moments of inertia are common features extracted from boundaries. If pictures are digitized, how to extract curve features that are really insensitive to the orientation of the sampling grid? Although this basic problem arises very often, it has not been fully solved yet: usually, features are extracted, that are sensitive to the orientation of the sampling grid. However, to reduce sensitivity, some heuristics are often applied [1, 2, 3]. In particular cases an appropriate filtering can also be performed on particular features [4]. In this paper,

it will be shown that all features are optimally extracted from a digitized curve when noise has been reduced in the coordinates describing that curve.

2. Parametrized curves

A curve is conveniently described by parametrized functions: two orthogonal coordinates x and y are defined as functions of a curvilinear abscissa s .

$$x = f(s),$$

$$y = g(s).$$

* This work was supported by the Swiss "commission pour l'Encouragement des Recherches Scientifiques" (CERS, projet 880).

Consider for instance a circle of radius R at the origin of a reference system, with s starting at $(R, 0)$.

$$\begin{aligned}x &= R \cos\left(\frac{s}{R}\right), \\y &= R \sin\left(\frac{s}{R}\right),\end{aligned}\quad (1)$$

When R decreases, the spatial frequency f (number of cycles per unit length) increases, but the magnitude of both x and y decreases. This means that a long curve (a large blob) has a relatively large low frequency content. The amplitude limit for high frequency components decreases proportionally to $1/f$.

3. Quantization of curves

When the picture of a curve is sampled in space with constant steps Δx_i and Δy_i , the curve appears as a set of points $\{P_i\}$. If the parametrized functions of a continuous curve are quantized without being sampled, they show for any value of the parameter, an integer number of elementary levels $\Delta x_q, \Delta y_q$. If $\Delta x_q = \Delta x_i$ and $\Delta y_q = \Delta y_i$, the parametrized functions lead to the same set of points $\{P_i\}$. Even in pictures that are actually sampled, it is then possible to represent curves by piecewise continuous parametrized functions. Doing so, one can take advantage of the continuity relations between neighbouring points of curves. Let us call the quantized functions $\hat{x}(s)$ and $\hat{y}(s)$. The quantization noise that is added to $x(s)$ and $y(s)$ is usually a random white noise (the power σ_n^2 of this white noise increases when the quantization grid becomes coarser: $\sigma_n^2 = \Delta^2/12$ where Δ denotes the minimum quantization interval). Even when it is not random as for a straight line, it has however an upperbound at $\Delta^2/4$.

4. Filtering

Arc element length, tangent angles and curvatures can be accurately measured from $x(s)$ and

$y(s)$; but, they are strongly affected by quantization noise when estimated from $\hat{x}(s)$ and $\hat{y}(s)$. This stems from the fact that they all are functions of derivatives of the coordinates; and it is a truism to state that derivation enhances high-frequency components. It is therefore important to estimate the frequency limit f_i above which the signal-to-noise ratio is likely to become very small and to lowpass filter the parametrized curve.

The filtering is expressed by:

$$\begin{aligned}\hat{x}^{(1)}(s) &= h(s) * \hat{x}(s), \\ \hat{y}^{(1)}(s) &= h(s) * \hat{y}(s)\end{aligned}\quad (2)$$

where $*$ denotes convolution and $h(s)$ is the impulse response of a low-pass filter (cut off spatial frequency f_i [cycles per length unit]).

In these equations, $\hat{x}^{(1)}(s)$ and $\hat{y}^{(1)}(s)$ are improved estimates of the parametrized curve.

Of course, the impulse response of this filter is chosen differently for the recognition of a particular curve, than for the common enhancement of many classes of curves. It should be noted that the parametrized curves are piecewise continuous and $h(s)$ need not be sampled.

5. Estimation of abscissa

In practical situations, the problem remains, to estimate the values of the parameter s along which the filtering should be performed. The parameter s , which is the actual arc length can obviously be expressed by $x(s)$ and $y(s)$:

$$|ds| = \sqrt{\left(\frac{dx}{ds}\right)^2 + \left(\frac{dy}{ds}\right)^2} ds. \quad (3.a)$$

However, in digitized pictures, we have only estimates of the actual arc length:

$$|d\hat{s}| = \sqrt{\left(\frac{d\hat{x}}{d\hat{s}}\right)^2 + \left(\frac{d\hat{y}}{d\hat{s}}\right)^2} d\hat{s}. \quad (3.b)$$

As mentioned above, $d\hat{s}$ is strongly affected by the quantization noise imbedded in $\hat{x}$ and $\hat{y}$. However, along the relatively noisy $\hat{s}$ (obtained by integra-

tion of $d\hat{s}$, a filtering can be performed, that gives better estimates of x and y , $\hat{x}^{(1)}(\hat{s})$ and $\hat{y}^{(1)}(\hat{s})$.

The same filtering can be applied again along the new curvilinear abscissa $\hat{s}^{(1)}$ defined by $\hat{x}^{(1)}$ and $\hat{y}^{(1)}$ (see 3.b). As iteration goes on, s can be better approached. Along $\hat{s}$, because of the noise components, the *apparent* length of the filter impulse response is not constant. This implies a variation in the frequency response of the filter. Therefore, in the first iterations, the filter should not be too selective in order to avoid distortions in the signal bandwidth. For instance, a low-pass filter would be designed with a relatively high cut-off frequency taking into account the largest variation of the *apparent* cut-off frequency.

For $v = x, y$, let us call $\hat{v}^{(n)}$ the observed variable and $\hat{v}^{(n)}$ the same variable after n iterations. The iterative procedure can then be described as follows:

$$\begin{aligned}\hat{x}^{(n)}(\hat{s}^{(n-1)}) &= h(\hat{s}^{(n-1)}) * \hat{x}^{(n)}(\hat{s}^{(n-1)}), \\ \hat{y}^{(n)}(\hat{s}^{(n-1)}) &= h(\hat{s}^{(n-1)}) * \hat{y}^{(n)}(\hat{s}^{(n-1)})\end{aligned}\quad (4)$$

and

$$\begin{aligned}|\mathrm{d}\hat{s}^{(n)}| &= \sqrt{\left(\frac{\mathrm{d}\hat{x}^{(n)}}{\mathrm{d}\hat{s}^{(n-1)}}\right)^2 + \left(\frac{\mathrm{d}\hat{y}^{(n)}}{\mathrm{d}\hat{s}^{(n-1)}}\right)^2} \mathrm{d}\hat{s}^{(n-1)}, \\ \hat{s}^{(n)} &= \int_0^{\hat{s}^{(n-1)}} \frac{\mathrm{d}\hat{s}^{(n)}}{\mathrm{d}\hat{s}^{(n-1)}} \mathrm{d}\hat{s}^{(n-1)}.\end{aligned}\quad (5)$$

Notice that until the last iteration the filter processes the observed coordinates. In principle, iteration is used here in order to obtain a noise-free curvilinear abscissa. When a good estimation of s has been found, any curvilinear filtering can be applied with predictable results because the impulse response is no longer affected by jitter.

6. Convergence

When iteration is used, care should be taken that the process firstly converges, and secondly towards a non-zero limit.

Consider the circle of Section 2. It is a simple pattern which has sinusoidal parametrized

functions. This example is of interest as any function can be viewed as a weighted sum of sine waves.

As the filter processes the circle, the radius varies. The radius resulting from the k th iteration is represented by $R^{(k)}$. It is equal to $x^{(k)}(0)$, if the impulse response of the filter is even. Equation (4) leads then to the following expression:

$$R^{(k+1)} = \int_{-\infty}^{+\infty} h(s^{(k)}) x^{(k)}(-s^{(k)}) \mathrm{d}s^{(k)} \quad (6)$$

Let us define

$$A^{(k)} = R^{(k+1)} / R^{(k)} \quad (7)$$

which appears as the gain of the recursion.

For a convergent non-zero process, $A^{(k)}$ should equal 1 for some values of $R^{(k)}$. Moreover its derivative should be negative for these values.

For our applications (see Section 7), $h(s)$ has been chosen as a rectangular window:

$$h(s) = \begin{cases} \frac{1}{L} & \text{for } -\frac{1}{2}L < s < \frac{1}{2}L, \\ 0 & \text{elsewhere.} \end{cases}$$

This is a satisfactory lowpass filter, often called "averager". It can be implemented easily and efficiently. For our smoothing purpose, the common values of L are between 5 and 9 sampling steps. Equations (6) and (7) become in this particular case:

$$R^{(k+1)} = \int_{-L/2}^{L/2} \frac{1}{L} \cdot R \cos\left(\frac{s}{R^{(k)}}\right) \mathrm{d}s \quad (8)$$

and

$$A^{(k)} = 2 \frac{R}{L} \sin\left(\frac{L}{2R^{(k)}}\right) \quad (9)$$

The process is convergent towards a non-zero limit for:

(a) $R > \frac{1}{2}L$,

(b) $L/\pi < R^{(n)} < \infty$ where $R^{(n)}$ is the starting value of the iteration. Moreover, the filtering leads to the asymptotic value R_∞ :

$$R_\infty = \frac{1}{2}L \operatorname{Arcsin}(L/2R). \quad (10)$$

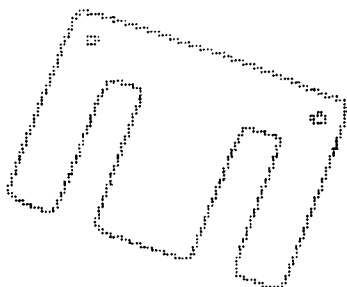


Fig. 1. This curve is very sensitive to the cartesian sampling grid. See in (Fig. 2) the errors in some features extracted from it before and after filtering.

Fig. 3 shows $A^{(k)} = f(R^{(k)})$ for this particular example.

In practical situations, L is usually small with respect to R , and these conditions are not restrictive.

Notice that the filter is non-linear in the sense that the spatial frequencies are shifted towards higher values.

7. Experiments

The processing described here has been applied for over one year in pattern recognition for

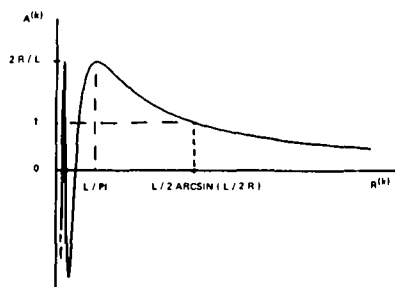


Fig. 3. Convergence domain ($R^{(k)} > L/\pi$) and final radius ($R_{\infty} = \frac{1}{2}L \arcsin(L/2R)$) for a circle of radius R after iterative filtering (rectangular impulse response of length L).

robotics [5]. In our experiments, iterative curve filtering is proved very efficient (Figs. 1 and 2). The extracted curve characteristics become less sensitive to quantization noise than to very low spatial frequency distortions induced by the common TV vidicon-camera used as image acquisition device. The sampling grid has 256 steps in x and y ; it covers most of the TV picture.

Iteratively, a simple averager is applied along curves in order to cut off the high frequency components of the quantization noise (see Fig. 6). When performed on 3 grid units, it reduces

	no filtering	averaging with a 3 grid unit square window	averaging with a 9 grid unit square window
maximum error in contour length estimation	20%	10%	<2%
maximum error in curvature standard- deviation estimation	45°	10°	<2°

Fig. 2. Effect of iterative filtering on some curve features. The filtering was applied for 2 iterations on the curve of Fig. 1. The picture was digitized with 128×128 picture elements on a cartesian grid. The error is defined as the half of the maximum difference in measurements, when the object rotates with respect to the digitization grid.

from 20% to 10% the maximum error in length estimation (Fig. 4). With a 9 grid unit averager, the length estimation error decreases to one percent. However, some high frequency components of the signal may now be affected (Fig. 5). As the iterative filtering described above usually converges very

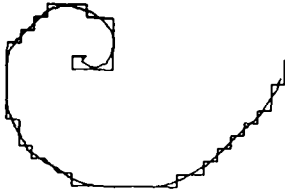


Fig. 4. Effect of iterative filtering on a curve. With a 3 grid unit square window, the quantization noise in x and y is reduced.

fast, the results of the second iteration is used in our pattern recognition applications.

Besides the improvement in feature estimation, the iterative filtering of parametrized functions presented here allows a regularly spaced sampling of curves at relatively low frequency.

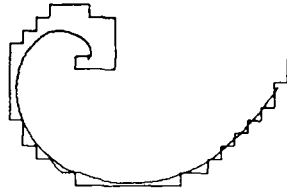


Fig. 5. Effect of iterative filtering of a curve. With a 9 grid unit square window, the quantization noise in x and y tend to disappear, but the highly curved segments of the spiral are strongly attenuated.

```

*** CONFIL.SAC *** 16-FEB-79 JDD: LTS: EFF.
C -- CONFIL.FOR
C 10: PROLOG: ABRONTOIRE DE TRAITEMENT DES SIGNAUX.
C 11: ECOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE
C 12: LAUSANNE, SWITZERLAND
C
C 20: SEF-78
C
C 30: THIS ROUTINE FILTERS ITERATIVELY PARAMETRIZED
C 40: CONTOURS. FILTERING WITH A RECTANGULAR
C 50: WINDOW. SAMPLING STEPS: WIDE WINDOW INTRADES
C 60: BOTH COORDINATES.
C 70: WITH THE NEW COORDINATES, A NEW CURVILINEAR
C 80: APPROXIMATION IS ESTIMATED.
C
C 90: 1.1:1: PARAMETRIZED: CONTOUR TO BE FILTERED: INTEGER ARRAY(S).
C 100: 1.1:2: NUMBER OF COORDINATE LINE(S).
C 110: 1.1:3: WIDTH OF THE RECTANGULAR WINDOW THAT PROCESSES COORDINATES.
C 120: 1.1:4: LOGICAL VALUE: .TRUE. -> THE WINDOW IS CLOSED. .FALSE. IF THE WINDOW IS OPEN
C 130: 1.1:5: NUMBER OF ITERATION TO BE EXPANDED
C 140: 1.1:6: FILTERED COORDINATES: FLOATING ARRAY(S).
C 150: 1.1:7: CURVILINEAR ABSCISSA: FLOATING ARRAY(S).
C
C 160: SUBROUTINE CONFIL (I1,I2,I3,N,MINING,BOUCLE,INTER,X,Y,TEST)
C 170: DIMENSION I1(I1,I2,I3),X(1),Y(1),TEST(1)
C 180: LOGICAL BOUCLE
C
C 190: MINING=MINING
C 200: BOUCLE=.FALSE.
C 210: INTER=INTER
C 220: X=X
C 230: Y=Y
C 240: TEST=.FALSE.
C 250: IF (TEST) GO TO 1.2
C 260: 1.2: TESTING, FIRST ITERATION.
C 270: 1.2:1: I1=1
C 280: 1.2:2: DO 4 A=1,M
C 290: 1.2:3: ISUM=X(I1)
C 300: 1.2:4: ISUM=Y(I1)
C 310: 1.2:5: IF (I2,LT,1) GO TO 3
C 320: 1.2:6: DO 2 A2=1,M2
C 330: 1.2:7: ANDIM=N2
C 340: 1.2:8: N2=N2+1
C 350: 1.2:9: IF (.NOT. BOUCLE .AND. ANDIM.LE.0) ANDIM=N2
C 360: 1.2:10: IF (ANDIM.LE.0) ANDIM=ANDIM+N
C 370: 1.2:11: IF (.NOT. BOUCLE .AND. N2.LT,1) N2=N2+1
C 380: 1.2:12: IF (N2.LT,1) N2=N2+1
C 390: 1.2:13: ISUM=ISUM+X(I1+ANDIM)*I2+Y(I1+ANDIM)*I3
C 400: 1.2:14: ISUM=ISUM+Y(I1+ANDIM)*I2+X(I1+ANDIM)*I3
C 410: 1.2:15: X(I1)=ISUM/XLENG
C 420: 1.2:16: Y(I1)=ISUM/YLENG
C 430: 1.2:17: CONTINUE
C 440: 1.2:18: GO TO 4
C 450: 1.2:19: COMPUTATION OF CURVILINEAR ABSCISSA
C 460: 1.2:20: DO 3 A=1,M
C 470: 1.2:21: I1=1
C 480: 1.2:22: N2=N2+1
C 490: 1.2:23: IF (.NOT. BOUCLE .AND. A1.GT,M) A1=M
C 500: 1.2:24: IF (.NOT. BOUCLE .AND. A2.LT,1) A2=1
C 510: 1.2:25: IF (A1.GT,M) A1=M
C 520: 1.2:26: IF (A2.LT,1) A2=M

```

Fig. 6. A Fortran program for iterative curve filtering. This program is not optimized in speed and in size, but it is believed to be legible.

Acknowledgement

The author is deeply grateful to Prof. F. de Coulon and to his colleagues at the LTS for many helpful discussions.

References

- [1] J.E. Bowie, "Syntactic descriptions of biological shape", *Ph.D. Dissertation*, Mass. Inst. of Techn., USA., July 1973, pp. 60.
- [2] Z. Kulpa, "Area and perimeter measurement of blobs in discrete binary pictures", *Computer graphics and image processing*, Vol. 6, 1977, pp. 434-451.
- [3] P. Kammenos, "Performances of polar coding for visual localisation of planar objects", *Proc. 8th Int. Symp. on Industrial Robots*, Stuttgart, Germany, May-June 1, 1978, pp. 143-149.
- [4] M.J. Eccles, M.P.C. MacQueen, D. Rosen, "Analysis of digitized boundaries of planar objects", *Pattern Recognition*, GB, Vol. 9, pp. 31-41.
- [5] J.D. Dessimoz, "Visual identification and location in a multi-object environment by contour tracking and curvature description", *Proc. 8th Int. Symp. on Industrial Robots*, Stuttgart, Germany, May 30-June 1, 1978, pp. 764-777.

ANNEXE 2

The arc length of a curve increases when zero-mean independent noise is added. The arc length is defined by equation 4.7. Replacing the $x(s)$ and $y(s)$ contributions by their sums with a noise expressed by $n_x(s)$ and $n_y(s)$ along the x-axis and the y-axis respectively, it becomes:

$$ds = \sqrt{(dx^2 + dn_x^2 + 2dx dn_x) + (dy^2 + dn_y^2 + 2dy dn_y)}$$

The only terms which can be negative are $2dn_x dx$ and $2dn_y dy$. As the quantization noise has zero-mean, and under the assumption of independance, both terms have also zero-mean. The radical has a positive mean value larger than in 4.7 due to the positive contributions of dn_x^2 and dn_y^2 . The arc length of the noisy curve, which is the integral of ds is then larger than the one of the noise-free curve.

BIOGRAPHIE

Jean-Daniel Dessimoz est né en 1953, à Conthey (Valais, Suisse).

Ses études secondaires à Sion se sont terminées par une Maturité Classique en 1971. Il a ensuite étudié à l'école polytechnique fédérale de Lausanne, où il a obtenu un Diplôme d'ingénieur-électricien en 1976.

De 1976 à 1980, J.D. Dessimoz a collaboré comme assistant de recherches et d'enseignement au Laboratoire de traitement des signaux de l'EPFL. Entre 1978 et 1980, il a également effectué un travail de doctorat.

Jean-Daniel Dessimoz est Professeur Assistant à l'université de Rhode Island (U.S.A) depuis juillet 1980.