

## Analyse d'images :

## le modèle pyramidal

## Image analysis : The pyramidal framework



Jean-Michel JOLION

Laboratoire d'Informatique Graphique et d'Intelligence Artificielle, Bat. 710, Université Claude Bernard, Lyon I, 69622 VILLEURBANNE CEDEX. Tél. : 72 84 40 00.

Diplôme d'ingénieur de l'institut National des Sciences Appliquées de Lyon, option informatique (1984). Thèse de Doctorat en informatique (analyse d'images en microscopie électronique, 1987). Séjour post-doctoral au Computer Vision Laboratory, Université du Maryland, USA (1987-1988). Maître de Conférences à l'Université Claude BERNARD, Lyon I depuis 1988. Activités de recherche au sein du laboratoire LIGIA centrées sur la vision artificielle et la modélisation géométrique. Tout particulièrement, recherche de techniques d'extraction de primitives géométriques incluant la notion de multirésolution. Application à la saisie intelligente de documents pour la CAO.

## RÉSUMÉ

L'analyse d'images numériques est un domaine où les notions de parallélisme et de récursivité sont fréquemment utilisées. L'architecture pyramidale permet de mettre efficacement en œuvre ces deux notions. Dans cet article, nous présentons trois aspects des pyramides prises tour à tour comme un modèle architectural pour une machine massivement parallèle, un modèle pour la représentation de données hiérarchisées et enfin un modèle algorithmique pour les processus d'extraction rapide d'information.

**MOTS-CLÉS :** Analyse d'Images, Pyramide, Modélisation, Parallélisme, Récursivité, Architecture.

## SUMMARY

*Image analysis is a field intensively using the notions of parallelism and recursivity. The pyramidal architecture is a proposed framework to efficiently use these notions. We present in this paper the different views of pyramids as a model for a parallel computer's architecture, a model for a hierarchical data structure and a model for a bidimensional parallel and recursive process.*

**KEY WORDS:** Image Analysis, Pyramid, Parallelism, Recursivity, Architecture.

## Introduction

La manipulation des images numériques par un ordinateur implique de traiter de manière similaire des volumes de données très importants. Heureusement, nombreuses sont, en analyse d'images, les techniques dites parallèles (lorsque le traitement effectué en un point ne dépend pas de celui effectué en un de ses voisins). C'est pourquoi la recherche d'architectures spécialisées, tant logicielles que matérielles, adaptées à ce domaine a été très féconde ces dernières années. On peut ainsi améliorer considérablement les temps de réponse des systèmes d'analyse d'images numériques.

La notion de multirésolution a été très rapidement mise en évidence grâce tout particulièrement aux

études relatives à l'extraction de contours dans une image [39]. Dans cette approche, le noyau de l'analyseur d'images doit avoir facilement accès à plusieurs images représentant la même scène à des résolutions différentes. Une structure de données permettant d'atteindre cet objectif était donc nécessaire.

Enfin, l'on sait que tout humain peut reconnaître, en une fraction de seconde, un objet connu, mais inattendu. Cette opération est réalisée en quelques étapes (10 à 15, chacune étant massivement parallèle) à l'intérieur du système visuel [54]. Pour intégrer ce phénomène de reconnaissance rapide d'objets chez l'homme, un modèle algorithmique puissant est nécessaire. Il devra prendre en compte : le parallélisme, la rapidité de transmission, la récursi-

vité des processus et des données (différents niveaux dans l'interprétation d'une scène) ainsi que la flexibilité structurelle (apprentissage, adaptativité à un contexte).

La notion de **pyramide** a été introduite dans le domaine de l'analyse d'images par Tanimoto et Pavlidis en 1975 [50]. Cette nouvelle architecture tente de satisfaire aux besoins que nous venons d'évoquer en permettant d'une part, des traitements très rapides, et d'autre part, la mise en application de techniques utilisant la notion de multirésolution. Depuis cette date, de très nombreuses recherches ont été menées pour développer tant l'aspect théorique que les applications de cette technique comme en témoignent deux livres exclusivement consacrés à cette nouvelle approche [40] et [51].

Nous proposons dans cet article une présentation de l'existant relatif au modèle pyramidal. Nous avons choisi de structurer cette étude autour de trois thèmes. Tout d'abord, nous introduirons le formalisme propre à ce domaine et nous présenterons les éléments essentiels qui constituent une pyramide (§ 1). Nous évoquerons à cette occasion les réalisations matérielles d'une telle structure. Par la suite, nous étudierons les deux applications principales de ce modèle dans le domaine de la multirésolution d'une part (§ 2) et de l'extraction rapide d'information d'autre part (§ 3). Enfin, nous aborderons, à travers la présentation de quelques modèles étendus, le problème de la flexibilité d'une telle structure (§ 4).

## 1. Un modèle architectural

### 1.1. LES CONSTITUANTS D'UNE PYRAMIDE

Introduisons tout d'abord quelques notions du domaine de l'analyse d'images nécessaires à notre discussion.

Un *élément d'image* est un couple  $(x, y)$  où  $x \in X$  et  $y \in Y$ ;  $X$  et  $Y$  sont les deux domaines d'existence (inclus dans l'ensemble  $\mathbb{N}$  des entiers) pour les coordonnées définissant la position d'un élément. L'ensemble des éléments d'image défini par  $I = X \times Y$  est appelé *image*.

Le plus souvent,  $X = Y = [0, 2^n - 1]$  où  $2^n$  caractérise la taille de l'image (nombre de lignes ou de colonnes). L'image traitée est alors carrée et l'on parle de structure arborescente quaternaire pour désigner la pyramide. Il est cependant tout à fait possible d'envisager la manipulation d'images rectangulaires comme nous le verrons par la suite. On désigne la structure obtenue par le nom de structure arborescente binaire. Des exemples de telles structures seront présentés par la suite dans le contexte des réalisations matérielles.

Un *pixel* est un triplet  $(x, y, v)$  où  $(x, y)$  désigne l'élément d'image associé au pixel et où  $v \in V$  est la valeur du pixel. Très souvent,  $v$  est appelé niveau de gris ou intensité lumineuse et est une valeur entière. Cependant, la valeur  $v$  pouvant également résulter d'un traitement, nous considérerons par la suite que

$V$  est un domaine réel ( $V \subset \mathbb{R}$ ). L'ensemble des pixels est donc  $I \times V$ .

Dans un premier temps, une pyramide peut être définie simplement comme une architecture informatique particulière pour une machine de vision combinant les notions de parallélisme et de récursivité [53].

La formalisation mathématique d'une pyramide est la suivante. Une pyramide  $P$  est une suite d'images  $I_h$ ,  $h \in [0, n]$ .  $h$  est appelé *niveau* et  $n + 1$  est le nombre de niveaux ou *hauteur* de la pyramide. Dans le cas d'une structure arborescente quaternaire, la taille de l'image  $I_h$  et son niveau sont liés par la relation :  $X_h = Y_h = [0, 2^{n-h} - 1]$ . Un élément de la pyramide sera donc un triplet  $(x, y, h)$  où  $(x, y)$  est un élément de l'image  $I_h$ . La pyramide  $P$  possède donc :

$$1 + 2 \cdot 2 + 4 \cdot 4 + \dots + 2^{n-1} \cdot 2^{n-1} + 2^n \cdot 2^n = 4/3 \cdot 2^n \cdot 2^n \text{ éléments.}$$

L'image  $I_n$  est appelée *apex* ou sommet de la pyramide. Cette image est constituée d'un seul élément. L'image  $I_0$  est appelée *base* de la pyramide. Le plus souvent, cette image est l'image initiale avant traitement. Dans la suite de cet exposé, nous supposons une base de  $2^n \times 2^n$  pixels.

Une machine pyramidale a trois composantes :

— Un ensemble de processeurs indépendants possédant leur propre mémoire. Chaque processeur est un élément de la pyramide.

— Un contrôleur central dont le rôle est le séquençement des opérations effectuées par les processeurs. Il sert également d'interface avec le monde extérieur pour lequel la pyramide n'est le plus souvent qu'une unité arithmétique et logique. Cette fonction peut être assurée par le processeur situé au sommet de la pyramide.

— Un réseau de communication reliant les différents processeurs entre eux et combinant deux techniques classiques : le mesh et le réseau arborescent (i.e. quadtree).

### 1.2. LE RÉSEAU DE COMMUNICATION

L'architecture interne de communication se traduit par deux types de relations directes entre processeurs : les relations horizontales (le mesh) et les relations verticales (le réseau arborescent).

*Relations horizontales* : Chaque élément de la pyramide  $(x, y, h)$  est connecté à ses voisins de même niveau :  $(x - 1, y, h)$ ,  $(x, y - 1, h)$ ,  $(x + 1, y, h)$ ,  $(x, y + 1, h)$ ,  $(x - 1, y - 1, h)$ ,  $(x + 1, y - 1, h)$ ,  $(x - 1, y + 1, h)$  et  $(x + 1, y + 1, h)$  si l'on adopte une 8-connexité. Si l'on choisit une 4-connexité, alors seuls les quatre premiers liens sont effectifs (cf. fig. 1). Les éléments du bord de l'image sont bien sûr des cas particuliers que l'on peut traiter de la même manière que les autres éléments en donnant aux ensembles  $X_h$  et  $Y_h$  la propriété de périodicité si cela est compatible avec la nature de l'image à traiter.

*Relations verticales* : Chaque élément  $(x, y, h)$  est relié avec des éléments du niveau  $h - 1$  appelés ses *fil*s et un (ou des) élément(s) du niveau  $h + 1$  appelé(s) son (ou ses) *père(s)*. Tout élément, exception faite de ceux appartenant à la base, a au moins 1 fils. Tout élément, exception faite du sommet, a au moins 1 père. Le nombre et la nature des relations verticales définissent une structure arborescente caractérisant l'architecture pyramidale.

La structure arborescente est dite *bin*aire lorsque chaque élément à 2 fils. Pour permettre une couverture totale de la surface de l'image, on alterne une partition en lignes puis en colonnes (cf. fig. 2). De ce fait, la pyramide ainsi obtenue a  $2n - 1$  niveaux.

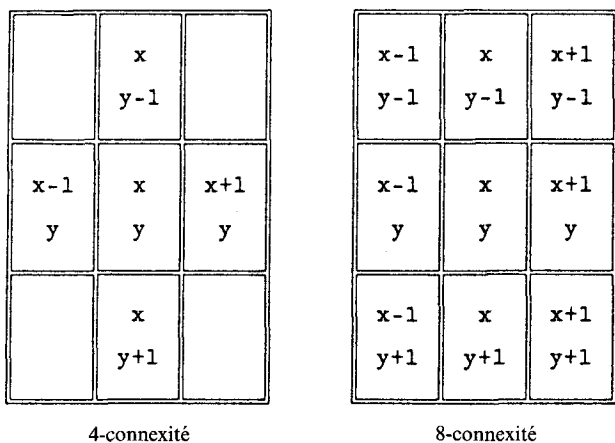


Fig. 1.

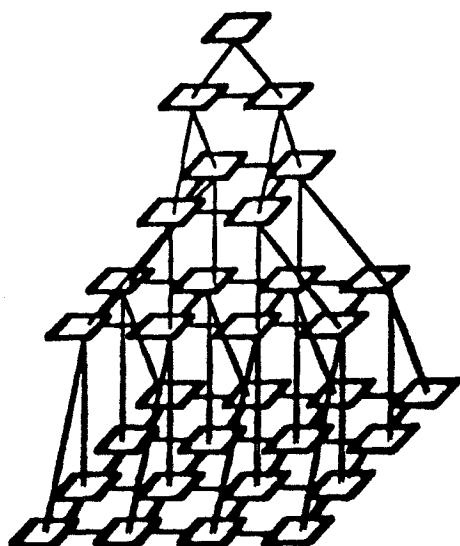


Fig. 2. — Structure arborescente binaire.

La structure la plus employée, est dite *quaternaire sans recouvrement*. Chaque élément  $(x, y, h)$  a 4 fils  $(2x, 2y, h - 1)$ ,  $(2x + 1, 2y, h - 1)$ ,  $(2x, 2y + 1, h - 1)$  et  $(2x + 1, 2y + 1, h - 1)$  et 1 père  $(\lfloor x/2 \rfloor, \lfloor y/2 \rfloor, h + 1)$  où  $\lfloor \ \rfloor$  désigne l'opérateur partie entière. La figure 3 illustre cette structure dans un cas de 4-connexité.

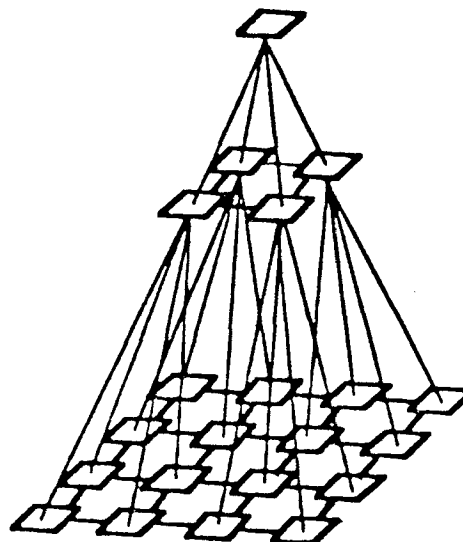


Fig. 3. — Structure arborescente quaternaire 4-connexe sans recouvrement.

On obtient une structure *quaternaire avec recouvrement* si l'on ajoute les voisins des 4 fils centraux. De ce fait, l'élément  $(x, y, h)$  a 16 fils (4 principaux et 12 secondaires) et 4 pères (1 principal et 3 secondaires). La figure 4 illustre cette configuration dans le cas d'une structure monodimensionnelle.

La structure arborescente binaire présente l'avantage de la simplicité de l'élément de la pyramide (3 liens contre 9 dans une structure quaternaire 4-connexe sans recouvrement et 28 dans une structure quaternaire 8-connexe avec recouvrement). L'implémentation est donc simplifiée. Par contre, une structure binaire est plus coûteuse car elle comporte  $2^{2n+1} - 1$  éléments au lieu de  $4/3 \cdot 2^{2n}$  dans une pyramide quaternaire, et le temps d'extraction de l'information est plus important  $(2n + 1)$  niveaux au lieu de  $(n + 1)$ . De plus, la structure quaternaire est mieux adaptée pour la mise en œuvre des algorithmes les plus couramment utilisés de par sa structure isotropique. Cependant, il est tout à fait possible d'envisager l'implantation d'algorithmes, utilisant comme modèle de traitement (cf. § 3) une pyramide quaternaire, dans une structure binaire. L'adaptation est alors réalisée par une couche logicielle intermédiaire.

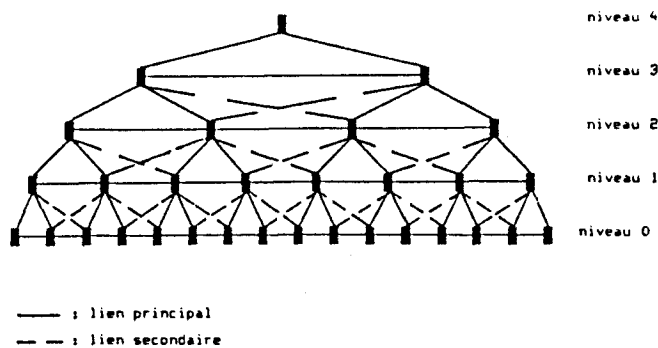


Fig. 4. — Structure arborescente quaternaire 4-connexe avec recouvrement.

## 1.3. LES RÉALISATIONS MATÉRIELLES

Plusieurs projets de réalisations matérielles d'une telle machine sont en cours d'achèvement en laboratoire. Une analyse détaillée de ce type de structure peut être trouvée dans [11]. Une étude plus générale des architectures parallèles dont fait partie l'architecture pyramidale est présentée dans [27].

Parmi les principaux projets, on peut citer les machines PAPIA (Pyramidal Architecture for Parallel Image Analysis, Italie) [9], SPHINX (Système Pyramidal Hiérarchisé pour le traitement d'Images Numériques, France) [34], GAM (USA) [44], HCL (USA) [52].

Pour toutes ces machines, l'élément de pyramide est un processeur 1 bit disposant de 128 bits de mémoire propre pour SPHINX et GAM et 256 bits pour PAPIA. Les machines GAM et PAPIA sont conçues selon le modèle arborescent quaternaire 4-connexe sans recouvrement. La machine HCL est également une machine quaternaire mais 8-connexe sans recouvrement. Par contre, SPHINX est une machine à structure binaire. Pour toutes ces machines la base est une image de  $128 \times 128$  pixels.

Il existe d'autres moyens pour simuler matériellement le comportement d'une pyramide. La machine PVM (Pyramid Vision Machine) développée au David Sarnoff Research Center [55] est constituée de trois modules de mémoire de  $512 \times 512$  éléments chacun et de deux modules de calcul dont un est spécialisé dans les opérations de convolution et l'autre dans les manipulations d'adressage permettant de simuler les différents niveaux de la pyramide. Ces modules sont reliés par un réseau piloté par un ordinateur hôte (habituellement un PC/AT).

Cette machine traite 30 images ( $256 \times 240$  pixels) par seconde. Elle est donc très bien adaptée à la surveillance et au suivi d'objets en temps réel. La pyramide, pour une image donnée, est construite en  $4/3$  fois le temps d'acquisition de la base. Ce facteur  $4/3$  est lié à la simulation quasi séquentielle de tous les niveaux de la pyramide (cf. § 1.1). Les mécanismes mis en œuvre dans cette machine reposent sur l'utilisation des pyramides laplacienne et gaussienne définies par Burt [8] (cf. § 2.1 et § 3.3).

Une pyramide peut également être simulée sur une architecture de type « array processor » [11]. Tous les niveaux sont stockés dans une seule matrice d'éléments grâce à un jeu d'adresses (cf. fig. 5). Chaque processeur appartient au plus à deux niveaux. Cette structure porte le nom de pyramide plate.

Similaire à cette approche, on peut citer la *Connection Machine*<sup>TM</sup> [37] (architecture hypercube de 64 K processeurs développée par la société Thinking Machine en collaboration avec le M.I.T.). Les liaisons verticales et horizontales peuvent être établies dynamiquement au niveau logiciel. Plus souple, cette solution permet de simuler différentes architectures pyramidales et tout particulièrement la pyramide plate.

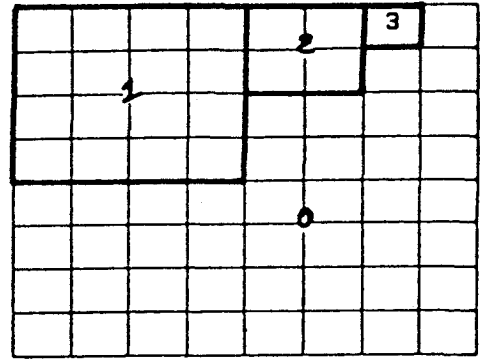


Fig. 5. — Pyramide plate.

Enfin, plus proche du modèle initial, on peut envisager une structure pyramidale à base de transputers. Cette solution présente le double avantage de la puissance des éléments pyramidaux (processeurs 32 bits) et de la mémoire locale importante (4 mégaoctets par processeur). Cependant, cette technologie ne permet pas encore un nombre élevé d'éléments (le réseau T. Node TN364 comporte 64 transputers T800). Cet inconvénient peut être amoindri si l'on utilise comme entité de la base, non pas un pixel mais une sous-image de l'image de départ. De plus, un transputer ne possède que 4 liens ce qui complique et limite son utilisation pour construire des pyramides.

Ce principe, processeurs puissants en nombre limité, est à l'origine de la machine EGPA (Erlangen General Purpose Array) développée en RFA [15] pour laquelle la base de la pyramide est limitée à  $4 \times 4$  éléments. Chaque processeur (32 bits) possède une mémoire locale de 4 mégaoctets. Les communications entre éléments sont similaires à une communication entre tâches (système de boîte aux lettres en mémoire partagée).

Comme le remarquent Hwang et Fü [21], une telle architecture est surtout adaptée aux tâches de vision dites de haut niveau, c'est-à-dire celles aboutissant à une interprétation de l'image par des mécanismes complexes. La solution idéale est peut être celle utilisant une architecture à base de nombreux éléments simples pour les premiers niveaux de la pyramide, et des éléments plus complexes au sommet de celle-ci.

Une synthèse des développements de ces différentes machines peut être trouvée dans [10].

Ces différentes machines sont encore très fortement tributaires de périphériques rarement parallèles. Ceci rend non négligeable le coût en temps de chargement des données en mémoire et en récupération d'information.

Le principal avantage du modèle pyramidal est de combiner parallélisme et récursivité. La nature de l'information transmise à travers le réseau de communication caractérise la pyramide. Nous considérerons tout d'abord deux utilisations de ce modèle où la pyramide est tour à tour modèle pour la représentation de données et modèle de traitement.

## 2. Un modèle pour la représentation de données

Les techniques utilisant la multirésolution sont très importantes pour au moins deux raisons. Tout d'abord, elles réduisent le nombre d'opérations nécessaires au traitement d'une image. La pyramide est alors un outil de compression de données. Ensuite, elles permettent la mise en œuvre du principe de focus d'attention reposant sur le fait que pour chaque entité de l'image, il existe une résolution donnée dans laquelle la recherche (i.e. détection) et l'extraction (i.e. localisation) sont facilitées.

Cette idée n'est pas nouvelle. En effet, en 1971, Rosenfeld et Thurston [39] proposaient une approche de la détection de frontières grâce à des opérateurs de tailles variables. Ils mettaient ainsi en évidence la notion de hiérarchie entre contours.

### 2.1. CONSTITUTION D'UNE REPRÉSENTATION EN MULTIRÉSOLUTION

Dans l'approche en représentation de données, à chaque niveau de la pyramide correspond une version à résolution réduite de l'image initiale contenue dans la base de la pyramide. Pour cela, nous supposons une structure arborescente quaternaire.

En chaque élément du niveau  $h$  de la pyramide, la nouvelle valeur est obtenue par une moyenne pondérée des valeurs de ses  $k^2$  fils. On caractérise cette construction par son masque de pondération  $W(i, j)$  ( $i \in [0, k-1]$ ,  $j \in [0, k-1]$ ) ou *noyau*. Une analyse fréquentielle visant à définir le noyau optimal peut être trouvée dans [29]. Bien qu'optimaux, les noyaux proposés sont difficilement utilisables car le plus souvent impairs et de trop grande taille. Adoptant une approche spatiale, Burt [6] a défini un ensemble de contraintes sur les noyaux :

$$(1) \text{ normalisation : } \sum_{i=0}^{i=k-1} \sum_{j=0}^{j=k-1} W(i, j) = 1.$$

Il y a ainsi conservation du domaine de variation de l'information associée à l'image. Ceci est important lorsque cette information est une mesure physique telle que l'intensité lumineuse.

$$(2) \text{ symétrie : } W(i, j) = W(k-1-i, j) = W(i, k-1-j) = W(k-1-i, k-1-j)$$

pour tout  $(i, j)$ .

Cette deuxième contrainte est induite par l'absence de repère absolu dans une image. Aucune direction ne doit donc être *a priori* favorisée.

$$(3) \text{ unimodalité : } 0 \leq W(i, j) \leq W(p, q) \text{ pour } i \leq p \leq k/2 \text{ et } j \leq q \leq k/2.$$

L'unimodalité permet d'assurer la présence des poids les plus forts au centre de la zone considérée localement. Par cette contrainte, on introduit une conservation (relative) de la continuité.

(4) équi-contribution au niveau supérieur :

$$\sum_{i=0}^{i=1} \sum_{j=0}^{j=1} W(i+2x, j+2y) = 1/4 ; \text{ pour tout } (x, y) \\ (W(a, b) = 0 \text{ si } a \text{ (ou } b) > k-1).$$

Cette dernière contrainte, conçue dans le cadre des architectures quaternaires avec recouvrement où un pixel est pris en compte dans la construction de plus de un élément du niveau supérieur, impose que tout élément d'un niveau donné intervienne pour le même poids dans la construction du niveau supérieur. En effet, en l'absence de règle particulière, aucune zone de l'image ne doit être artificiellement privilégiée. Cette contrainte ne peut être validée que si le noyau utilisé est le même pour tout élément d'un niveau donné.

Selon les objectifs à atteindre, de nouvelles contraintes peuvent être introduites.

Considérons le cas de la pyramide quaternaire en recouvrement présentée au paragraphe précédent. Nous avons  $k=4$ . Le type de noyaux suivant est souvent employé :

$$\begin{bmatrix} B^2 & AB & AB & B^2 \\ AB & A^2 & A^2 & AB \\ AB & A^2 & A^2 & AB \\ B^2 & AB & AB & B^2 \end{bmatrix}$$

Il vérifie les contraintes si :

$$(1) \quad 4 \cdot (A+B)^2 = 1 \Rightarrow A+B = \pm 1/2$$

(2) de par sa construction

$$(3) \quad |A| \geq |B|$$

$$(4) \quad (A+B)^2 = 1/4 \Rightarrow A+B = \pm 1/2.$$

De nombreux noyaux satisfont à ces contraintes :

$(A, B) = (1/2, 0)$  : équi-contribution sans recouvrement ;

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0,25 & 0,25 & 0 \\ 0 & 0,25 & 0,25 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$(A, B) = (1/4, 1/4)$  : équi-contribution avec recouvrement ;

$$\begin{bmatrix} 0,625 & 0,0625 & 0,0625 & 0,0625 \\ 0,0625 & 0,0625 & 0,0625 & 0,0625 \\ 0,0625 & 0,0625 & 0,0625 & 0,0625 \\ 0,0625 & 0,0625 & 0,0625 & 0,0625 \end{bmatrix}$$

$(A, B) = (0,37, 0,13)$  : distribution gaussienne.

$$\begin{bmatrix} 0,0169 & 0,0481 & 0,0481 & 0,0169 \\ 0,0481 & 0,1369 & 0,1369 & 0,0481 \\ 0,0481 & 0,1369 & 0,1369 & 0,0481 \\ 0,0169 & 0,0481 & 0,0481 & 0,0169 \end{bmatrix}$$

La figure 6 montre un exemple de pyramide où l'image initiale, le « cul de sac », est successivement réduite de  $256 \times 256$  à 1 pixel par un noyau d'équi-contribution sans recouvrement. Dans ce cas particulier, la valeur associée à chaque élément est l'intensité lumineuse.

2.2. APPROCHE ASCENDANTE DE L'EXTRACTION D'INFORMATION

Haralick *et al.* [17] proposent l'utilisation des concepts de la morphologie mathématique lors de la construction de la pyramide. Dans un premier temps, l'image du niveau  $h$  est traitée par un opérateur de morphologie [45] (érosion, dilatation, ouverture, ...). Un échantillonnage permet alors d'obtenir l'image du niveau  $h + 1$ . Cette approche est le premier pas vers l'introduction d'un processus de décision dans la construction de la pyramide puisque le choix de l'opérateur et de l'élément structurant associé à cet opérateur permet de mettre en évidence une forme particulière. La pyramide n'est plus une simple représentation de données mais devient un outil spécialisé dans la mise en évidence d'une caractéristique particulière.

Cette approche *ascendante* fut antérieurement proposée par Hong et Rosenfeld [19, 20] pour la détection de zones homogènes. Pour cela, à chaque élément  $P$  du niveau  $h$ , on associe trois mesures :

- $v(P)$  : niveau de gris moyen ;
- $\sigma^2(P)$  : variance des  $v(P)$  calculée sur tous les fils  $P'$  de  $P$  (faible pour une zone homogène) ;
- $a(P)$  : aire de la zone de l'image initiale rattachée à l'élément  $P$ .

Entre  $P$  et chacun de ses fils, on construit une mesure de la qualité de la liaison :

$$q(P, P') = (1 + d(P, P')) \cdot \exp[-1/2 \cdot (v(P) - v(P'))^2 / \sigma^2(P)] / (\sqrt{2\pi} \cdot \sigma(P'))$$

où  $d(P, P')$  est la distance spatiale normalisée entre un père et un de ses fils (cf. fig. 7).

L'aire  $a(P)$  est exprimée en fonction de la mesure  $q$  :

$$a(P) = \sum_{P'} q(P, P') \cdot a(P') / Q(P')$$

où  $P'$  est un fils de  $P$  et  $Q(P') = \sum_{P''} q(P'', P')$  avec  $P''$  père de  $P'$ .

Répartition spatiale des fils (■) et pères (\*) pour une structure  $4 \times 4$  (avec recouvrement). Le fils  $S$  est relié aux pères  $F_1, F_2, F_3$  et  $F_4$ . Les distances normalisées sont respectivement 1,  $\sqrt{5}, \sqrt{5}$  et 3.

De même, les valeurs  $v(P)$  et  $\sigma(P)$  sont calculées en fonction des mesures connues sur les fils  $P'$  par :

$$v(P) = \sum_{P'} v(P') \cdot a(P')$$

$$q(P, P') / \sum_{P'} a(P') \cdot q(P, P')$$

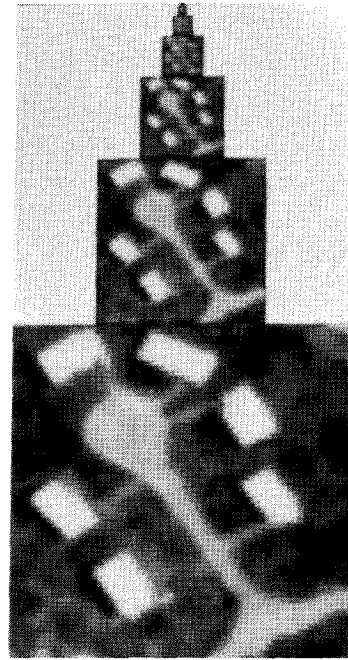


Fig. 6. — Cul de Sac.

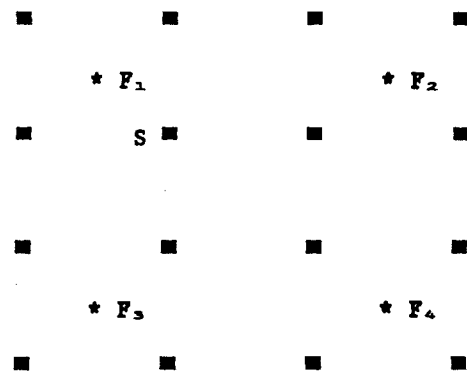


Fig. 7.

$$\sigma^2(P) = \sum_{P'} (v(P) - v(P'))^2 \cdot a(P')$$

$$q(P, P') / \sum_{P'} a(P') \cdot q(P, P')$$

Un élément  $P$  sera dit *racine* s'il se situe au sommet de la pyramide ou si la somme des liens avec tous ses pères (potentiels) n'excède pas un certain seuil. On peut ainsi obtenir une décomposition de l'image en zones homogènes en niveaux de gris.

Cette notion de racine apparaît dans les techniques de détection d'objets compacts sur un fond homogène. Dans cette approche, un opérateur de type laplacien permet de détecter « la meilleure racine » correspondant à un objet [41]. Cette détection peut également être réalisée par des tests statistiques d'homogénéité entre différents niveaux [14].

Jolion *et al.* [22] ont amélioré les performances de cette technique en remplaçant la notion de racine

unique associée à un objet (et donc sensible au bruit) par celle de racines multiples associées au contour de cet objet. La reconnaissance même partielle du contour suffit pour assurer la détection de l'objet.

L'approche ascendante, associée au concept de détection, est complétée par l'approche descendante permettant de délimiter la position exacte, dans l'image de départ, de l'objet associé à la (ou les) racine(s) [14]. Dans cette deuxième étape, chaque élément de la pyramide descendant d'une racine décide de son appartenance à l'objet, le plus souvent par application d'un critère de similarité avec son père. Des contraintes de consistance locale peuvent également être introduites.

### 2.3. REPRÉSENTATION HIÉRARCHIQUE DE CONTOURS

Il est bien sûr tout à fait possible de construire une pyramide résumant une information autre que l'intensité lumineuse. Ainsi Meer *et al.* [33], Kropatsch [26] et plus récemment Clermont *et al.* [12], proposent une structure permettant la prise en compte d'une information contour (cf. fig. 8).

En effet, il n'est pas possible de construire une pyramide contour en appliquant un opérateur de

détection de frontières à chaque niveau d'une pyramide résumant l'information intensité lumineuse. Ceci est une conséquence du filtrage qui masque les transitions. Il est donc nécessaire que l'image initiale, stockée dans la base de la pyramide, contienne une information relative au contour [47].

On peut cependant adopter une structure de réduction par utilisation de noyaux en se servant de l'information d'amplitude du gradient, si l'information de direction du gradient n'est pas significative. Une telle approche a été utilisée par Hanson et Riseman dans le système VISION [16].

### 3. Un modèle de traitement

#### 3.1. PARALLÉLISME ET RÉCURSIVITÉ

Hormis son utilisation dans le domaine de la multirésolution, la pyramide est un outil de traitement très performant alliant parallélisme et récursivité. Soit  $V(I_h)$ , une information que l'on veut extraire d'une image  $I_h$ . Cette information pourra être calculée récursivement si l'on peut décomposer le processus de calcul de la manière suivante :

$$V(I_h) = f(V(I_{hk}), k = 1, K)$$

où  $f$  représente une fonction simple des  $K$  sous-images  $I_{hk}$  qui constituent une partition de  $I_h$ . La caractéristique de récursivité sera réellement effective si l'on peut appliquer le même raisonnement à chaque sous-image  $I_{hk}$ .

Le processus de calcul sera de plus dit parallèle si le calcul de  $V(I_{hk})$  ne dépend pas de celui de  $V(I_{hj})$  pour  $k \neq j$ .

La structure pyramidale permet l'extraction rapide de toute information pour laquelle le processus de calcul est récursif et parallèle. A titre d'exemple, examinons le cas du calcul du niveau de gris moyen d'une image  $I_n$  défini par :

$$m = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} v_{ij} / N^2 = S_n / N^2$$

avec  $N = 2^n$ .

Pour réaliser ce calcul, on utilise une pyramide quaternaire sans recouvrement (chaque élément du niveau  $h$  a 4 fils au niveau  $h-1$ ). De plus, les liaisons horizontales ne sont pas nécessaires. Chaque processeur n'est donc relié qu'à 5 autres processeurs (4 fils et un père), ce qui facilite l'implantation matérielle d'une telle structure. La somme  $S_n$  peut être décomposée en :

$$S_n = \sum_{i=0}^1 \sum_{j=0}^1 S_{n-1}(i, j)$$

où  $S_{n-1}(i, j)$  représente la somme des niveaux de gris pour l'un des quadrants principaux de l'image  $I_n$ . Chacune des sommes  $S_{n-1}(i, j)$  peut être calculée par l'élément  $(i, j, n-1)$  de la pyramide, indépendamment de celles calculées par ses voisins (parallélisme). On peut également répéter cette opération

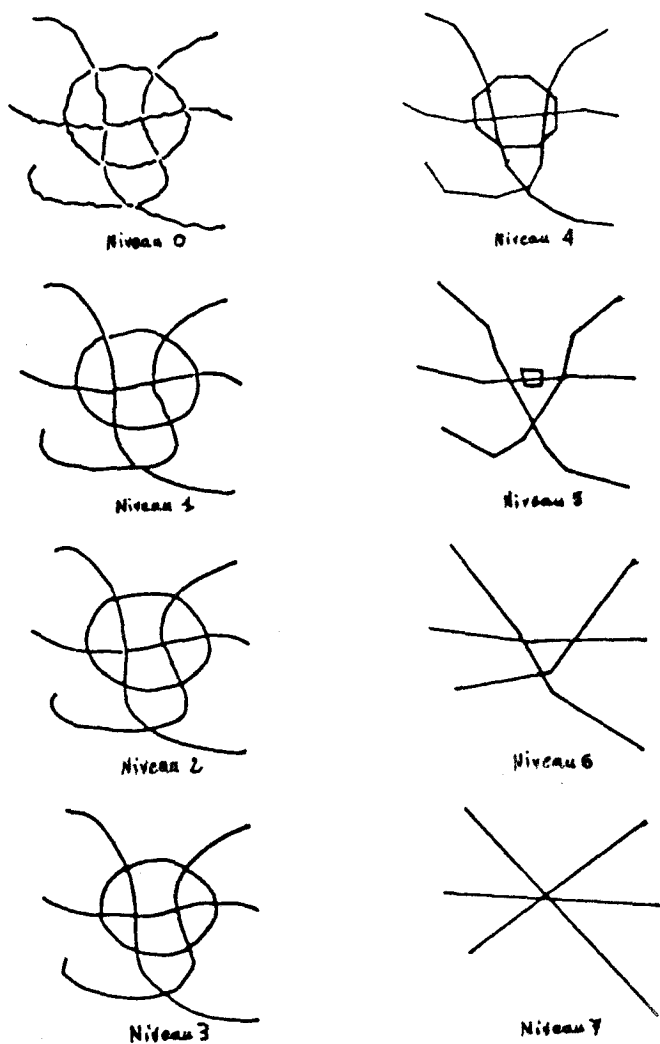


Fig. 8. — Représentation hiérarchique de contours d'après [33].

de décomposition pour les niveaux inférieurs de la pyramide (récursivité) :

$$S_{h+1}(i, j) = \sum_{x=0}^i \sum_{y=0}^j S_h(2i+x, 2j+y).$$

La somme  $S_n(0, 0)$  ( $= S_n$ ) peut donc être calculée récursivement en  $n = \log_2(2^n)$  étapes. Le nombre d'étapes ne dépend que de la hauteur de la pyramide (i.e. de la taille de l'image initiale). On peut, selon le même procédé et en  $n$  étapes de calcul, extraire les autres moments statistiques de l'image (variance, ...), faire du comptage, de la modélisation par les moindres carrés (modélisation par facettes), ... [42].

Une étude complète de la complexité des algorithmes élémentaires en fonction de la structure de la pyramide est proposée dans [1]. Une étude plus générale de l'implantation d'algorithmes spécialisés de vision dans des architectures parallèles est due à Q. Stout [49].

Meer *et al.* [32] ont récemment proposé un algorithme utilisant une architecture pyramidale pour estimer rapidement et avec une très grande fiabilité et précision, la variance d'un bruit gaussien additif à une image. Cet algorithme déduit la variance du bruit par analyse d'une séquence de variances associées aux différents niveaux de résolution d'une image. Chacune des variances est obtenue en  $\log_2$  (taille image) pas de calcul.

### 3.2. EXTRACTION RAPIDE D'INFORMATIONS

Le modèle algorithmique pyramidal est applicable pour des processus plus complexes que le simple calcul des moments statistiques d'une image.

Blanford et Tanimoto proposent dans [3] un algorithme pour la détection de points brillants dans une image. L'information à extraire est de type booléen (présence ou absence) et est obtenue récursivement en  $O(\log_2$  taille de l'image)) dans une structure pyramidale classique. L'information recherchée est propagée de la base vers le sommet grâce à un opérateur max. Cette approche est similaire à celle proposée par Haralick [17].

Blanford d'une part [4], Jolion et Rosenfeld d'autre part [23] proposent un algorithme pyramidal pour la transformée de Hough. L'information à extraire est la liste des segments de droites présents dans une image. Au niveau le plus bas, chaque élément dresse une liste des segments possibles par interprétation de son voisinage local. Les relations horizontales sont donc nécessaires. La fonction  $f$  (extraction de l'information) est alors une fusion de plusieurs listes fournies par le niveau inférieur en une nouvelle liste transmise au niveau supérieur. Une implantation de l'algorithme conçu par Jolion et Rosenfeld sur la Connection Machine<sup>TM</sup> (64 K processeurs bits ayant chacun 64 K bits de mémoire locale) a permis d'obtenir un gain en temps d'un facteur 50 par rapport à l'algorithme séquentiel de la transformée de Hough implanté sur un VAX 11/785 au Computer Vision Laboratory [46].

### 3.3. SIMULATION DU MÉCANISME DE L'ŒIL

La recherche de processus visant à simuler le comportement visuel de l'œil humain a conduit Burt [7] à proposer un mécanisme pyramidal permettant de prendre en compte les phénomènes d'acuité et d'attention (vision dynamique).

Pour cela, Burt utilise la très célèbre pyramide laplacienne [6] qui représente une décomposition fréquentielle de l'image. Les composantes de plus basses fréquences se trouvent au sommet de la pyramide. Le niveau  $h$  de la pyramide laplacienne est défini par :

$$L_h = G_h - W * G_{h+1}$$

où  $G_h$  et  $G_{h+1}$  représentent les niveaux  $h$  et  $h+1$  d'une pyramide construite à l'aide d'un noyau gaussien (cf. § 2.1).  $W * G_{h+1}$  est la prédiction du niveau  $h$  à partir du niveau  $h+1$  et d'un noyau  $W$ .  $L_h$  est donc l'information perdue lorsque l'on passe du niveau  $h$  au niveau  $h+1$ . Le noyau  $W$  étant un noyau gaussien, il réduit toutes les transitions, c'est pourquoi la pyramide laplacienne est caractéristique des hautes fréquences.

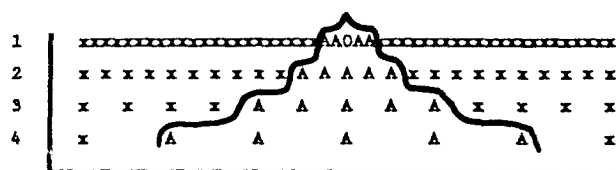


Fig. 9. — Échantillonnage sélectif. x : point non conservé ; A : point conservé ; O : point de fixation.

La différence d'acuité dans le champ visuel peut être simulée par un filtrage des hautes fréquences (valeurs des niveaux les plus bas) fonction de la distance au centre de l'image (assimilé au point de fixation, cf. fig. 9). La figure 10 présente un exemple de simulation de la fovea sur l'œil de Léna.



Fig. 10. — Simulation de la fovea sur l'œil de Léna.



Cependant, l'échantillonnage reste le même dans tout le champ visuel ce qui n'est pas conforme avec les résultats obtenus par les spécialistes de l'œil. Par contre, cette simulation permet de prendre en compte les phénomènes de détection rapide en fonction de la position de l'objet dans le champ visuel et de la taille du détail à repérer.

### 3.4. AUTRES APPLICATIONS

Certaines applications combinent la construction de niveaux de résolution réduite avec une extraction parallèle et récursive d'une information. Dans ces différentes approches, l'utilisation de la multirésolution a pour but la simplification de l'information de départ (essentiellement en réduisant son volume). Un processus classique permet alors d'extraire une information grossière. Dans un deuxième temps, on affine le résultat en retournant récursivement vers la résolution de départ.

On peut par exemple citer la détermination d'un seuil optimal dans un histogramme [38], [24]. Il s'agit dans ce cas d'une application monodimensionnelle. Dans un premier temps, on construit une version à résolution réduite de l'histogramme. Le niveau de résolution final est choisi en fonction de la dynamique de la courbe de départ. A ce niveau, un seuil grossier est déterminé par un algorithme classique de maximisation de la distance de Fisher entre deux populations.

La valeur de ce seuil est ensuite affinée par la prise en compte des niveaux inférieurs. La distance de Fisher est de nouveau utilisée mais uniquement pour les éléments de la pyramide fils du seuil obtenu au niveau supérieur. Le gain en temps est d'autant plus important que la réduction de résolution est grande. Cependant, le seuil obtenu n'est pas forcément optimal. En effet, plus la résolution est faible, plus l'incertitude augmente. Un compromis est donc nécessaire.

Le même principe est appliqué pour la discrimination de textures [48]. La réduction de résolution a pour but de faire apparaître les composantes essentielles des formes au détriment des textures locales. La méthode de relaxation a également été redéfinie dans un contexte de multirésolution [13].

### 3.5. ROBUSTESSE DES ALGORITHMES PYRAMIDAUX

Dans [31], Meer *et al.* proposent une étude très intéressante de la robustesse des algorithmes implantés dans une architecture pyramidale. Trois types de perturbations sont étudiés selon qu'elles s'appliquent : aux éléments des noyaux permettant la construction de la pyramide des différentes résolutions de l'image (bruit additif aux coefficients), aux liens reliant un père à ses fils (mélange de liens), à la structure même de la pyramide (non conservation de la relation donnant le nombre d'éléments par niveau). Même dans ce dernier cas, qui correspond à des perturbations très importantes, les algorithmes choisis par les auteurs (détection d'objets compacts [41] et recherche de seuils [38]) continuent de fournir des résultats de bonne qualité. Ceci permet

de conclure en l'efficacité des processus implantés ainsi qu'en la stabilité des structures pyramidales.

Cependant, l'utilisation d'une structure rigide vis-à-vis des données présente l'inconvénient majeur de la non-invariance en translation (essentiellement pour les structures sans recouvrement). Une très bonne étude critique détaillée de ces problèmes peut être trouvée dans [2]. C'est pourquoi des recherches ont été entreprises ces dernières années dans le but de concevoir une structure plus flexible qui pourrait prendre en compte la spécificité des données.

## 4. Les modèles étendus

Il est tout à fait possible d'envisager des pyramides n'utilisant pas la structure classique. En effet, le modèle pyramidal est défini par un ensemble de contraintes (maillage et noyaux carrés, nombre fixe de niveaux, relations horizontales et verticales préétablies, facteur de réduction de la résolution constant, ...). Par la relaxation d'une ou plusieurs de ces contraintes, on obtient des modèles différents que nous regroupons sous le vocable de modèles étendus.

### 4.1. PYRAMIDES À MAILLAGE NON CARRÉ

La première des contraintes que l'on peut relaxer est l'utilisation d'un support à maillage carré. On trouve dans [5] une analyse des structures hiérarchiques (arbres et pyramides) dans le contexte du maillage hexagonal. La figure 11 illustre deux exemples de structure pyramidale hexagonale. Dans le premier cas (*fig. 11 a*), la tessellation du plan est formée par un recouvrement d'hexagones dont le point central est le père pour le niveau supérieur. La structure proposée par Hartman et Tanimoto [18] (*fig. 11 b*), utilise une tessellation par partition en triangles. Quatre triangles voisins sont associés pour former un élément de la tessellation du niveau supérieur par suppression des trois points milieux des segments du nouveau triangle.

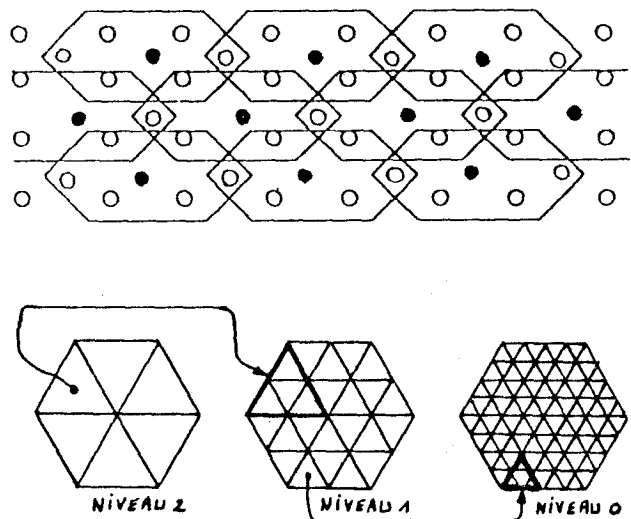


Fig. 11. — Structures hexagonales. (a) 6-connexe; (b) 3-connexe (d'après [18]).

Ce type de maillage est très utilisé dans le domaine de la morphologie mathématique [45]. Il permet de résoudre les problèmes liés au choix de la connexité (4 ou 8) dans les images à maillage carré. L'apparition prochaine de capteurs adaptés (caméras) à cette tessellation du plan permet d'envisager un développement rapide de son utilisation.

On peut également construire une structure pyramidale grâce à des noyaux rectangulaires et non plus carrés. On privilégie ainsi une des deux directions principales associées à la 4-connexité. Une étude de ces pyramides a été réalisée par Peleg *et al.* [36]. Les auteurs proposent une succession de noyaux évoluant de niveau en niveau et permettant ainsi de tenir compte de l'information contenue dans l'image. La pyramide ainsi construite sera dépendante d'un objectif bien particulier, ce que les auteurs résumant par le vocable « pyramide faite main ».

Cette approche pourrait être étendue aux autres directions principales associées à la 8-connexité. Cependant, la complexité de la structure pyramidale serait fortement accrue au moins dans le voisinage des bords de l'image. Or, il ne faut jamais négliger ces effets car, à résolution réduite, ils deviennent prépondérants dans le résultat final. La technique consistant à assimiler l'image à un tore pour bénéficier ainsi d'un domaine infini ne peut s'appliquer sur des scènes naturelles. Par contre, dans le domaine du contrôle qualité, cette hypothèse peut être très facilement vérifiée.

#### 4.2. PYRAMIDE STOCHASTIQUE

A l'opposé de cette approche, on doit citer la pyramide stochastique proposée par P. Meer [28]. Dans cette approche, la structure pyramidale est « aléatoire ». Cette approche a le mérite de relaxer les contraintes les plus fortes tout en conservant le principe de la réduction du volume d'information. Ce travail est, dans le domaine pyramidal, un des plus novateurs de ces dernières années.

Soit l'image initiale définie sur un maillage carré. Le support d'un élément  $(i, j)$  de l'image, noté  $S_{ij}$  est composé par les voisins de l'élément en utilisant la 8-connexité. Trois variables sont associées à chaque élément : deux variables d'état binaires  $p_{ij}$  et  $q_{ij}$  et une réalisation  $x_{ij}$  d'une variable aléatoire uniformément répartie entre 0 et 1. Un processus de décimation (i.e. choix des éléments du niveau supérieur) est mis en œuvre et les éléments restant constituent le niveau supérieur de la pyramide. A l'état initial, toutes les variables  $p_{ij}$  ont pour valeur 0. Un élément est conservé si la variable  $p_{ij}$  est positionnée à 1 durant le processus de décimation défini par :

$$p_{ij} = 1 \text{ si et seulement si}$$

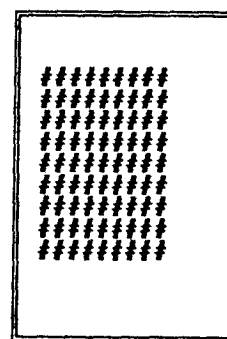
$$x_{ij} = \text{Max } x_{nm} \text{ pour } (n, m) \in S_{ij}.$$

La probabilité de transition de l'état 0 à l'état 1 durant une itération du processus est donc  $1/9$ . Cette probabilité étant trop faible, Meer propose la procédure itérative de consolidation suivante. A l'itération  $k$ , la deuxième variable d'état  $q_{ij}$  est positionnée à 1 si aucun élément du support  $S_{ij}$  n'a été retenu après

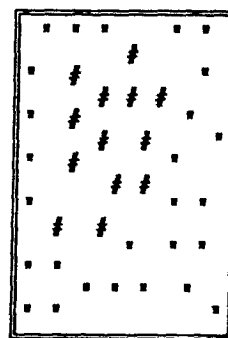
l'itération  $k - 1$ . A chaque itération, une nouvelle variable  $x_{ij}$  est engendrée. La règle de mise à jour de  $p_{ij}$  n'est alors appliquée qu'aux éléments  $(i, j)$  tels que  $q_{ij} = 1$ .

Meer et Connelly [30] ont montré que ce processus se stabilise après trois itérations. Le ratio moyen de décimation (nombre de pixels de  $I_h$ /nombre de pixels de  $I_{h+1}$ ) est 5,44 avec un écart type de  $6,7 \times 10^{-2}$ . La hauteur d'une telle pyramide est approximativement la même que celle d'une structure régulière.

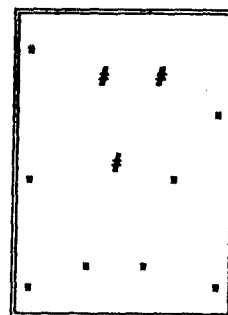
La figure 12 représente un exemple des deux premiers niveaux de la pyramide stochastique. Sur cet exemple, on obtient des ratios de décimation de 5,37 pour le premier niveau et de 4,66 pour le second. Ces valeurs sont plus fortes que celles obtenues avec une structure régulière où un pixel sur 4 est conservé.



(a)



(b)



(c)

Fig.12. — Pyramide stochastique. (a) image initiale ; (b) premier niveau, les pixels non conservés sont représentés en blanc ; (c) deuxième niveau.

Comme dans une structure classique, il existe une relation père-fils. Un fils est raccordé à l'élément conservé le plus proche qui devient son père. Ainsi, un père peut avoir un nombre très variable de fils. Ce type de structure a servi pour la génération de motifs aléatoires [30] ainsi que pour l'étude de la stabilité des algorithmes pyramidaux comme nous le soulignons dans le paragraphe précédent [31].

Montanvert *et al.* ont récemment conçu une structure proche de la pyramide stochastique permettant la prise en compte de concepts liés aux données tels que la proximité, la similarité entre valeurs associées aux pixels (label, niveaux de gris) [35]. Cette structure peut être considérée comme une solution intermédiaire entre la pyramide stochastique et la pyramide adaptative (§ 4.3).

#### 4.3. LA PYRAMIDE ADAPTATIVE

La détermination automatique de la structure pyramidale adéquate pour un objectif donné n'est pas possible dans l'état actuel des connaissances sur ces structures. Cependant, c'est très vraisemblablement la caractéristique qui pourra permettre aux systèmes de vision d'acquérir les propriétés de flexibilité et d'adaptativité. C'est pourquoi l'avenir des structures pyramidales réside sans aucun doute dans les recherches sur les structures dites floues ou non rigides telles que les pyramides stochastiques proposées par P. Meer.

En effet, si l'on conserve la faculté pour un père d'avoir un nombre variable de fils, on peut remplacer la variable aléatoire  $x_{ij}$  par un prédicat caractérisant localement les critères recherchés. La structure obtenue dépend alors de l'objectif caractérisé par le prédicat et prenant en compte les informations teinte et forme. Ce mécanisme est à l'origine de la pyramide adaptative définie par Jolion et Montanvert [25].

L'information teinte peut être extraite par un mécanisme de liens père-fils similaire à celui défini par Hong et Rosenfeld dans [19] et décrit en détail dans le chapitre 2. Ce mécanisme s'adapte parfaitement à une structure non rigide. De plus, il peut très facilement être généralisé au cas des images couleur (3 composantes).

## Conclusion

Nous avons abordé dans cet article l'application du concept de pyramide au domaine de l'analyse d'images. Nous avons vu que cette notion peut être un modèle pour représenter à la fois une architecture matérielle de processeurs, une succession d'images à résolutions réduites ou bien encore un processus de calcul ou plus généralement d'extraction rapide d'information.

L'utilisation d'opérateurs morphologiques dans un tel contexte est une solution d'avenir pour les problèmes de détection rapide d'objets. Pour cela, une structure à maillage hexagonal est souhaitable.

A plus long terme, dans le souci de modéliser le mécanisme de l'œil, des structures moins rigides,

obtenues par relaxation des contraintes définissant le modèle initial, doivent être développées. La pyramide stochastique est un premier pas très important dans cette voie.

Ce dernier type de structure, dans son principe, sert de base à nos essais d'implantation d'un processus de segmentation naturelle [25] visant à simuler le comportement humain en reconnaissance rapide d'objets connus mais inattendus tel qu'il est défini par Rosenfeld dans [43].

## Remerciements

Je tiens à remercier tout particulièrement A. Montanvert pour les nombreuses discussions préliminaires à cet article ainsi que pour ses commentaires pertinents sur les versions précédentes de ce texte. Je remercie également les relecteurs pour leurs commentaires avisés.

*Manuscrit reçu le 2 février 1989.*

## BIBLIOGRAPHIE

- [1] N. AHUIA, S. SWAMY, Multiprocessor Pyramid Architectures for Bottom-up Image Analysis, *IEEE trans. on Pattern Analysis and Machine Intelligence*, Vol. PAMI-6, No. 4, July 1984, p. 463-475.
- [2] M. BISTER, *A Critical View on Pyramid Segmentation Algorithms*, Vrije Universiteit Brussel, Informatics and Electronics Lab., Rapport Technique, IRIS-TR-0006, 1989.
- [3] R. P. BLANFORD, S. L. TANIMOTO, Bright-Spot Detection in Pyramids, *Proc. of the 8th Int. Conf. on Pattern Recognition*, Paris, 27-31 octobre 1986, p. 1280-1282.
- [4] R. P. BLANFORD, *Dynamically quantized pyramids for Hough vote collection*, University of Washington, Seattle, Department of Computer Science, Rapport Technique n° 87-07-08. Version plus courte dans *IEEE Workshop on Computer Architecture for Pattern Analysis and Machine Intelligence*, Seattle, WA, 5-7 octobre 1987, p. 145-152.
- [5] P. J. BURT, Tree and pyramid structures for coding hexagonally sampled binary images, *Computer Graphics and Image Processing*, Vol. 14, 1980, p. 271-280.
- [6] P. J. BURT, Fast filter transforms for image processing, *Computer Graphics and Image Processing*, Vol. 16, 1981, p. 20-51.
- [7] P. J. BURT, Attention Mechanisms for Vision in a Dynamic World, *Proc. of the 9th Int. Conf. on Pattern Recognition*, Rome, 14-17 novembre 1988, p. 977-987.
- [8] P. J. BURT, *Smart Sensing in Machine Vision*. In: *Machine Vision*, H. Freeman éditeur, Perspectives in Computing, Vol. 20, Academic Press, 1988, p. 1-30.
- [9] V. CANTONI, M. FERRETTI, S. LEVI ALDI, F. MALOBERTI, *A pyramidal project using integrated technology*, Integrated Technology for Parallel Image Processing, S. Levisaldi, éditeur, Londres, Academic Press, 1985.
- [10] V. CANTONI, S. LEVI ALDI, *Pyramidal Systems for Computer Vision*, Springer-Verlag, Berlin, NATO ASI, Séries F, Vol. 25, 1986.
- [11] V. CANTONI, S. LEVI ALDI, Multiprocessor Computing for Images, *Proc. of IEEE, Special issue on Computer Vision*, August 1988, p. 959-969.

- [12] P. CLERMONT, A. BELAID, A. MERIGOT, A parallel pyramidal algorithm to determine curve orientation, *Image Analysis and Processing II*, V. Cantoni et V. Di Gesu éditeurs (Proc. of the 4th Int. Conf. on Image Analysis and Processing, 23-25 septembre 1987, Palerme, Italie), Plenum Press, 1988, p. 353-360.
- [13] Z. DINGDING, L. JIAN, W. FAGUAN, Multiresolution Relaxation : Experiments and Evaluations, *Proc. of the 9th Int. Conf. on Pattern Recognition*, Rome, 14-17 novembre 1988, p. 712-714.
- [14] A. D. GROSS, A. ROSENFELD, Multiresolution Object Detection and Delineation, *Computer Vision, Graphics and Image Processing*, Vol. 39, 1987, p. 102-115.
- [15] W. HANDLER, U. HERZOG, F. HOFMANN, H. J. SCHNEIDER, Multiprozessoren für breite Anwendungsgebiete : Erlangen General Purpose Array, GI/NTG Fachtagung Architektur und Betrieb von Rechensystemen, *Informatik Fachberichte*, Berlin, Springer-Verlag, 1984, p. 195-208.
- [16] A. R. HANSON, E. M. RISEMAN, Segmentation of natural scenes, *Computer Vision Systems*, Hanson et Riseman éditeurs, Academic Press, 1978, p. 129-163.
- [17] R. M. HARALICK, C. LIN, J. S. I. LEE, X. ZHUANG, Multiresolution Morphology, *1st Inf. Conf. on Computer Vision*, Londres, June 8-11, 1987, p. 516-520.
- [18] P. N. HARTMAN, S. L. TANIMOTO, A Hexagonal Pyramid Data Structure for Image Processing, *IEEE trans. on Systems, Man, and Cybernetics*, Vol. SMC-14, No. 2, March/April 1984, p. 247-256.
- [19] T. H. HONG, A. ROSENFELD, Compact region extraction using weighted pixel linking in a pyramid, *IEEE trans. on Pattern Analysis and Machine Intelligence*, Vol. PAMI-6, No. 2, March 1984, p. 222-229.
- [20] T. H. HONG, M. SHNEIER, Extracting compact objects using linked pyramids, *IEEE trans. on Pattern Analysis and Machine Intelligence*, Vol. PAMI-6, No. 2, March 1984, p. 229-237.
- [21] K. HWANG, K. S. FU, Integrated computer architecture for image processing and data base management, *Computer*, Vol. 16, January 1983, p. 51-60.
- [22] J. M. JOLION, P. MEER, A. ROSENFELD, Border delineation in image pyramids by concurrent tree growing, *Pattern Recognition Letters*, février 1990.
- [23] J. M. JOLION, A. ROSENFELD, An  $O(\log n)$  Pyramid Hough Transform, *Pattern Recognition Letters*, Vol. 9, No. 5, 1989, p. 343-349.
- [24] J. M. JOLION, A. ROSENFELD, Coarse-Fine Bimodality Analysis of Circular Histograms, *Pattern Recognition Letters*, Vol. 10, No. 3, 1989, p. 201-207.
- [25] J. M. JOLION, A. MONTANVERT, La pyramide adaptative : construction et utilisation pour l'analyse de scènes 2D, *7<sup>e</sup> congrès RFA*, AFCET-INRIA, Paris, novembre 1989, p. 197-206.
- [26] W. C. KROPATSCH, Curve Representations in Multiple Resolutions, *Proc. of the 8th Int. Conf. on Pattern Recognition*, Paris, 27-31 octobre 1986, p. 1283-1285.
- [27] M. MARESCA, M. A. LAVIN, H. LI, Parallel Architectures for Vision, *Proc. of IEEE, Special issue on Computer Vision*, August 1988, p. 970-981.
- [28] P. MEER, Stochastic Image Pyramids, *Computer Vision, Graphics and Image Processing*, Vol. 45, 1989, p. 269-294.
- [29] P. MEER, S. BAUGHER, A. ROSENFELD, Frequency domain analysis and synthesis of image pyramid generating kernels, *IEEE trans. on Pattern Analysis and Machine Intelligence*, Vol. PAMI-9, 1987, p. 512-522.
- [30] P. MEER, S. CONNELLY, A fast parallel method for synthesis of random patterns, *Pattern Recognition*, Vol. 22, No. 2, 1989, p. 189-204.
- [31] P. MEER, S. N. JIANG, S. BAUGHER, A. ROSENFELD, Robustness of image pyramids under structural perturbations, *Computer Vision, Graphics and Image Processing*, Vol. 44, 1988, p. 307-331.
- [32] P. MEER, J. M. JOLION, A. ROSENFELD, Fast Algorithm for Blind Noise Variance Estimation, *IEEE trans. on Pattern Analysis and Machine Intelligence*, Vol. PAMI-12, février 1990, p. 216-223.
- [33] P. MEER, A. SHER, A. ROSENFELD, The Chain Pyramid : Hierarchical Contour Processing, Computer Vision Laboratory, University of Maryland, Rapport Technique CS-TR-2072, July 1988, article soumis à la revue *IEEE trans. on Pattern Analysis and Machine Intelligence*.
- [34] A. MERIGOT, B. ZAVIDOVIQUE, F. DEVOS, SPHINX, a pyramidal approach to parallel image processing, *Proc. of CAPAIDM workshop*, Miami, 1985, p. 107-111.
- [35] A. MONTANVERT, P. MEER, A. ROSENFELD, Hierarchical Image Analysis Using Irregular Tessellations, *Computer Vision Laboratory*, University of Maryland, Rapport Technique CS-TR-2322, September 1989.
- [36] S. PELEG, O. FEDERBUSCH, R. A. HUMMEL, Custom-Made Pyramids, *Robotics Research Technical Report*, No. 85, New York University, Computer Science Division, November 1986.
- [37] T. POGGIO, J. LITTLE, E. GAMBLE, W. GILLET, D. GEIGER, D. WEINSHALL, M. VILLALBA, N. LARSON, T. CASS, H. BULTHOFF, M. DRUMHELLER, P. OPPENHEIMER, W. YANG, A. HURLBERT, The MIT Vision Machine, *Proc. of Image Understanding Workshop*, Cambridge, Mass., USA, April 6-8, 1988, Morgan Kaufmann, p. 177-198.
- [38] T. Y. PHILLIPS, A. ROSENFELD, A. SHER,  $O(\log n)$  Bimodality Analysis, *Pattern Recognition*, Vol. 22, N° 6, 1989, p. 741-746.
- [39] A. ROSENFELD, M. THURSTON, Edge and curve detection for visual scene analysis, *IEEE trans. on Computer*, Vol. TC-20, 1971, p. 512-569.
- [40] A. ROSENFELD, *Multiresolution Image Processing and Analysis*, Springer, Berlin, 1984.
- [41] A. ROSENFELD, A. SHER, Detection and Delineation of Compact Objects Using Intensity Pyramids, *Pattern Recognition*, Vol. 21, 1988, p. 147-151.
- [42] A. ROSENFELD, Some Pyramid Techniques for Image Segmentation. In : *Pyramid Systems for Computer Vision*, V. Cantoni et S. Levialdi éditeurs, Berlin Springer, 1986, p. 261-271.
- [43] A. ROSENFELD, Recognizing Unexpected Objects : A Proposed Approach, *Int. Journal of Pattern Recognition and Artificial Intelligence*, Vol. 1, No. 1, April 1987, p. 71-84.
- [44] D. H. SCHAEFER, P. HO, P. BOYD, C. VALLEJOS, The GAM pyramid. In : *Parallel Computer Vision*, L. Uhr éditeur, Orlando, Academic Press, 1987, p. 15-42.
- [45] J. SERRA, *Image Analysis and Mathematical Morphology*, Academic Press, 1982.
- [46] A. SHER, A. ROSENFELD, A pyramid Hough transform on the connection machine, *Computer Vision Laboratory*, University of Maryland, Rapport Technique CS-TR-2182, February 1989, 12 p.
- [47] M. SHNEIER, Two hierarchical linear feature representations : edge pyramids and edge quadtrees, *Computer Graphics and Image Processing*, Vol. 17, 1981, p. 211-224.
- [48] M. SPANN, R. WILSON, A quad-tree approach to image segmentation which combines statistical and spatial information, *Pattern Recognition*, Vol. 18, Nos. 3/4, 1985, p. 257-269.
- [49] Q. F. STOUT, Mapping Vision Algorithms to Parallel Architectures, *Proc. of IEEE, Special issue on Computer Vision*, August 1988, p. 982-995.
- [50] S. TANIMOTO, T. PAVLIDIS, A hierarchical data structure for picture processing, *Computer Graphics and Image Processing*, Vol. 4, No. 2, June 1975, p. 104-119.

- [51] S. TANIMOTO, A. KLINGER, *Structured Computer Vision*, Academic Press, 1980.
- [52] S. TANIMOTO, T. J. LIGOCKI, R. LING, A prototype pyramid machine for hierarchical cellular logic. In: *Parallel Computer Vision*, L. Uhr éditeur, Orlando, Academic Press, 1987, p. 43-83.
- [53] S. TANIMOTO, From pixels to predicates in pyramid machines, *Proc. of Workshop on « From the pixels to the feature »*, Bonas, France, August 22-27, 1988.
- [54] S. THORPE, Traitement d'images par le système visuel de l'homme, Traitement d'Images : du Pixel à l'Interprétation, CNRS, *TUPI, Atelier Scientifique*, Aussois, 19-22 avril 1988, p. IX1-IX.12.
- [55] G. S. VAN DER WAL, J. O. SINNIGER, Real time pyramid transform architecture, *SPIE Proc. of Intelligent Robots and Computer Vision*, Boston, 1985, p. 300-305.