REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Mouloud MAMMERI de Tizi-Ouzou

Faculté de Génie Electrique et d'Informatique
Département d'Automatique

PROJET DE FIN D'ETUDES

En vue de l'obtention du diplôme

D'INGENIEUR D'ETAT EN AUTOMATIQUE

Thème

Segmentation d'images texturées à base des modèles Autorégressifs bidimensionnels

Proposé par :

M^r^. HAMMOUCHE. K

Présenté par :
M^lle^. LAGAB SONIA
M^r^. AFETTOUCHE FARID

Soutenu le : 21/09/2011

Promotion 2011

Remerciements

A travers ce modeste travail, nous tenons à remercier vivement notre promoteur M^r^ HAMMOUCHE KAMAL pour l'intéressant documentation qu'il a mis à notre disposition, pour ses conseils précieux et pour toutes les commodités et aisances qu'il nous a apportés durant notre étude et réalisation de ce projet.

Nous exprimons également notre gratitude à tous les professeurs et enseignants qui ont collaboré à notre formation depuis notre premier cycle d'étude jusqu'à la fin de notre cycle universitaire.

Sans omettre bien sur de remercier profondément tous ceux qui ont contribué de prés ou de loin à réalisation du présent travail.

Et enfin, que nos chers parents et familles, et bien avant tout, trouvent ici l'expression de nos remerciements les plus sincères et les plus profonds en reconnaissance de leurs sacrifices, aides, soutien et encouragement afin de nous assurer cette formation d'ingénieur dans les meilleures conditions.

Sommaire

Chapitre II : Modélisation et méthodes d'estimations du modèle AR (2D)

Introduction Générale

Le traitement d'images est une discipline qui connait actuellement un essor considérable grâce au développement des outils informatiques et du matériel visuel (caméra, scanner, appareillage médical ect.). Il touche des domaines aussi variés que le médical, la télévision, l'imagerie satellitaire, le multimédia, ect.. et où il ne s'agit plus uniquement de traiter les images pour les améliorer mais aussi de les comprendre et de les interpréter. Détecter, localiser et reconnaître un objet dans une image est une opération relativement subjective qui peut différer d'une personne à une autre. Et pour reconnaître des objets, il faut au préalable les segmenter, c'est-à-dire partitionner l'image en un ensemble de région pour pouvoir interpréter leur contenue. Cette tache devient très difficile lorsque les zones qui constituent l'image son texturées. Pour segmenter convenablement ce genre d'images, il est nécessaire d'analyser leurs textures.

Le but de notre travail consiste à segmenter des images texturées en se basant sur l'analyse de la texture basée sur le principe de la modélisation. Comme technique de modélisation, on s'est intéressé aux modèles autorégressifs bidimensionnels. Ces modèles nécessitent la définition d'un support de causalité dans le domaine spatial pour définir des notions de $\ll passé, présent\ et\ futur \gg$ et qui caractérise la causalité du signal image. Trois types de support sont étudiés dans ce mémoire : le support quart de plan (QP), le support semi causal (SC) et le support non causal (NC). Pour estimer les paramètres autorégressifs, nous avons fait appel à la méthode des moindres carrées.

Le présent mémoire est divisé en trois chapitres.

Le premier chapitre est dédié aux généralités sur les traitements d'image texturées. Nous commençons par les différentes définitions et types de textures puis nous présenterons les différentes approches de segmentation. Dans le deuxième chapitre, nous allons d'abord introduire la définition du modèle autorégressif (AR) mono et bidimensionnels et les différents choix de causalité. Par la suite nous allons aborder la modélisation et les méthodes d'estimations des coefficients autorégressifs. L'utilisation de ces coefficients comme attributs de texture dans le processus de segmentation est exposée dans le troisième chapitre. Une conclusion générale clôtura ce mémoire.

I.1) Introduction :

Nous présenterons dans ce chapitre la notion de la texture et les différentes approches d'analyses de celles-ci. La texture constitue une caractéristique importante dans le domaine de l'analyse d'image. L'étude de cette dernière dans le cadre de la segmentation demeure toujours un problème d'actualité.

Par la suite, nous donnerons la définition de la segmentation et nous présenterons ses différentes approches. La segmentation est une étape très importante dans une chaine d'analyse car c'est à partir de l'image segmentée que des mesures sont effectuées pour l'extraction des paramètres en vue de la classification et l'interprétation.

I.2) Définitions de la texture :

La texture est utilisée pour traduire un aspect homogène des objets d'une image, donc elle peut être considérée comme une information visuelle qui peut être associée à des adjectifs tels que : lisse, madrée, fine, grossière……..etc.

Unser [2] définit la texture comme suite :

<< une texture est une région d'une image pour laquelle il est possible de définir une fenêtre de dimension minimale , telle qu' une observation au travers de celle-ci se traduit par une perception (impression) visuelle identique pour toutes les translations possibles de cette fenêtre à l'intérieure de la région considérée >>

Cette propriété d'invariance par translation de perception visuelle d'une texture est exploitée dans la plupart des méthodes de classification à partir d'attributs de texture [3], [4], [5], [6].

Selon Picard [7], il est difficile d'établir une définition de la texture, il est cependant possible de dégager trois propriétés essentielles :

- La complexité d'une texture est difficilement quantifiable.
- Une texture est caractérisée par des variations importantes dans les hautes fréquences. Une zone uniforme ou quasiment lisse n'est pas une texture pour Picard.
- La notion de texture n'est valable que pour un ensemble de résolutions et d'observations données.

Selon Zhang [8], la notion de la texture découle de la nécessité de décrire des informations d'une image de la manière la plus simple possible :<< *une texture peut être considérée comme une façon de décrire les corrélations entre les différentes zones d'une région de l'image* >>.

La définition littéraire de la texture : est comme suite << répétition spatiale d'un même ou plusieurs motifs dans une seule ou différente direction de l'espace >>.

En résumé, Les principales propriétés visuelles d'une texture que l'on dégage de la littérature sont les suivantes :

- Une texture ne peut être définie que sur un voisinage dont la taille dépend de la taille des motifs qui la composent.
- La notion de la texture est liée à la résolution de l'observation. En effet, une région peut apparaitre texturée à une certaine échelle et uniforme à une autre.

Aucun chercheur n'a pour l'instant donné une définition claire et satisfaisante de la texture. Chacun essaie d'appréhender cette notion de son centre d'intérêt, plusieurs orientations ont été envisagées pour la définition de la texture.

Les définitions rappelées plus haut soulignent un aspect important des textures liées à la disposition spatiale des objets d'une texture (motif, primitive). Des différentes textures peuvent être distinguées selon la répartition spatiale des motifs, celle-ci permet de distinguer assez naturellement trois types de textures.

I.2.1) Textures macroscopiques (structurelles)

Dans ces types de texture, il est défini comme une répétition plus au moins régulière et périodique de ces primitives ou de ces motifs. Les primitives ou les motifs sont faciles à extraire visuellement, on parle alors de textures structurées.

Elles peuvent également correspondre à des images de synthèse. Ce type exige que l'image soit formée d'une texture régulière et se heurte alors à de nombreux problèmes. Tout d'abord, il est généralement difficile de repérer les motifs élémentaires, d'autant plus que les propriétés de ces motifs ne sont pas nécessairement constantes dans l'image (variation de l'éclairement). Un autre problème est dû à la répartition des primitives qui n'est pas

forcément régulière (un tissu avec des trames plus serrées par endroits), ce qui entraine l'introduction d'un élément aléatoire.

Notons que les textures parfaitement régulières existent rarement dans la réalité donc leur utilisation se limite à des applications très précises. Elles représentent en générale un milieu artificiel, par exemple un mur de briques figure (I.1) :

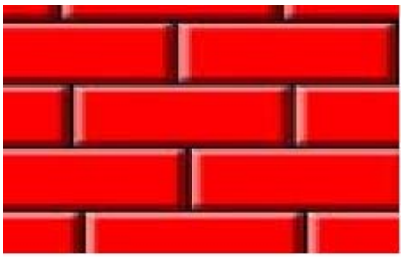


Figure (I.1) : Mure de briques

Figure (I.2) : Grains de café

I.2.2) Textures aléatoires (microscopique) :

Contrairement à une texture structurelle, une micro texture ne comporte ni de primitive isolable ni de fréquence de répétition et on ne peut pas séparer un motif d'un autre.

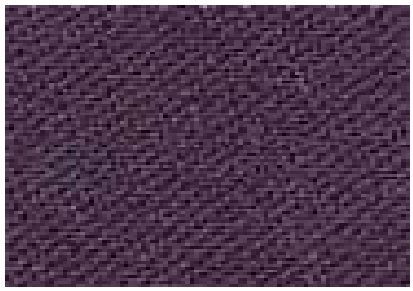
La texture est définie comme la réalisation d'un processus stochastique. On parle alors de texture aléatoire dans la mesure ou la distribution des motifs apparait irrégulière ou désordonnée dans l'image par exemple l'eau (Fig. I.3), l'herbe (Fig. I.4), ...etc. Ces textures décrivent bien les textures naturelles.

Figure (I.3) : Eau

Figure (I.4) : Herbe

I.2.3) Textures directionnelles

Ces textures se caractérisent essentiellement par certaines orientations dominantes sans qu'il y'ait une répétition, vraiment régulière, ni complètement aléatoire, ni de motif de base. Elle est définie comme une texture orientée ou directionnelle. Par exemple un tissu (Fig. I.5).



(I.5) : Tissu

I.2.4) Textures mixtes :

On distingue aussi des textures composées de plusieurs types de textures appelées textures mixtes (Fig. I.6).

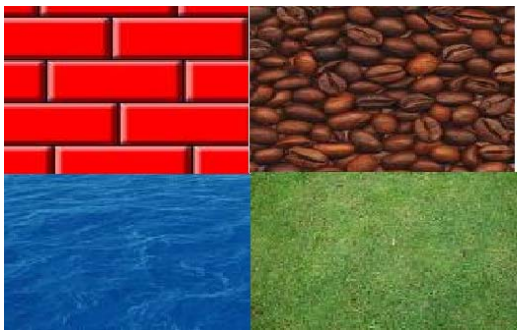


Figure (I.6) : Mixtes

I.3) Définition du motif d'une texture

L'approche formelle tente de définir les composantes d'une texture ainsi que leurs interactions.

D'après Haralick [10] et Francos [11], nous définissons le motif d'une texture en prenant en compte trois composants :

- Le motif de base : c'est la plus petite structure spatiale de la texture.
- La transformation de motif de base : elle caractérise d'éventuelles déformations géométriques et des modifications des niveaux de gris du motif au sein de la même texture.
- La loi de probabilité de la transformation de motif : elle permet de définir la transformation d'un motif donné .Nous définissons donc le motif ou la primitive d'une texture de la façon suivante :

  << *Le motif d'une texture est la plus petite structure spatiale contenue dans une texture dont la forme est régie par la même loi de probabilité au sein d'une texture*>>.

I .4) Finesse d'une texture

La finesse d'une texture est une propriété perceptuelle importante puisqu'elle caractérise la taille du motif d'une texture.

Selon Gagalowicz, une texture fine est caractérisée par une fonction d'autocorrélation tendant vers zéro rapidement en fonction de la distance de déplacement. Cette fonction pour une texture grossière (primitives de grande taille) décroit beaucoup plus lentement.

Dans le cas des textures déterministes, cette analyse ne doit se faire que sur la période de la texture considérée.

Cette définition caractérise la variation de la corrélation entre pixels plus ou moins proches pour une texture donnée.

Nous définissons la finesse de la texture comme la caractérisation de la taille du motif élémentaire composant une texture à une résolution donnée. La résolution est liée à la distance d'observation d'une texture ou à la taille de la fenêtre d'analyse de celle-ci. La finesse contrairement au caractère déterministe ou aléatoire d'une texture, varie en fonction de la résolution de l'observation.

I.5) Méthodes d'analyse de la texture [13]

La diversité des images, ainsi que la complexité de donner une définition précise de la texture a permis l'émergence de plusieurs méthodes d'analyse de la texture[15],[16]. Ces méthodes ont pour but d'extraire un ensemble d'attributs ou de paramètres pouvant décrire les caractéristiques de la texture. Ces attributs doivent être représentatifs, pertinents et discriminants de façon qu'on puisse discerner une texture parmi d'autres. Leur extraction peut être classée essentiellement en cinq approches, l'approche structurelle, fréquentielle, statistique, l'approche basée sur les modèles et celle basée sur la morphologie mathématique (Fig. 7).

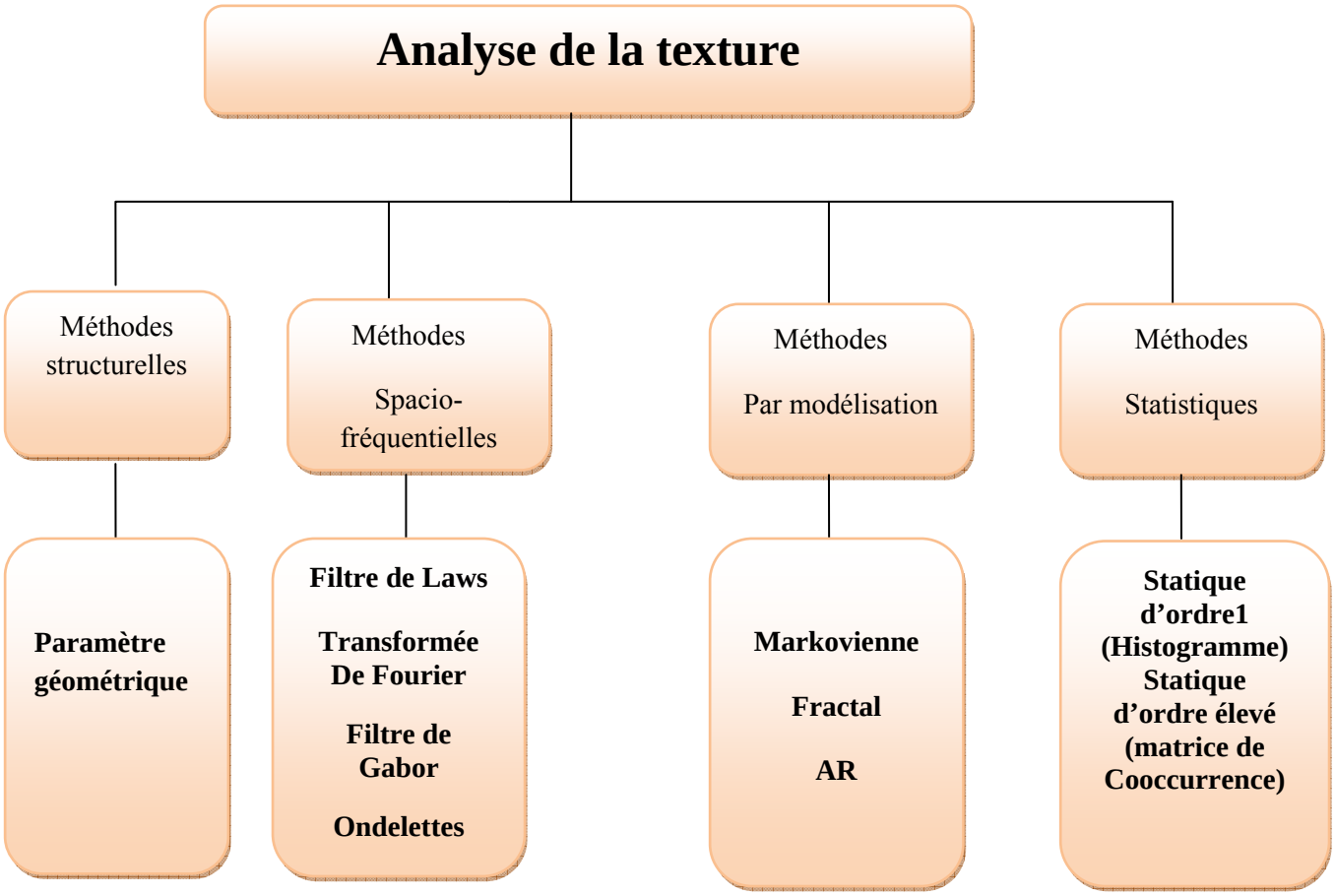


Figure (I.7) Méthodes d'analyse de la texture

I.5.1) Méthode structurelles :[14]

L'approche structurelle cherche à extraire et localiser des primitives des textures [17]. Principalement, ces méthodes utilisent des techniques d'autocorrélation pour retrouver le placement dans la texture de la primitive initialement extraite, afin d'en déduire une règle de placement. Cette approche ne sera pas plus commentée, étant donné les techniques employées sont diverses et très liées à la nature des images. Le trait caractéristique de ces méthodes est que tout se déroule en deux étapes, l'extraction de la primitive puis la recherche de la règle de placement.

Notons qu'en principe, ces méthodes utilisent des technique d'autocorrélation pour retrouver le placement dans la texture de la primitive extraite, afin d'enduire une règle de placement.

- **Fonction d'autocorrélation**

La fonction d'autocorrélation permet de mettre en avant les dépendances spatiales existantes entre certaines paires de pixels et d'obtenir de multiples informations perceptuelles sur la texture.

Cette fonction mesure une distance entre l'image originale et sa translatée. A partir de cette mesure, on peut estimer la taille des éléments d'une texture présente dans une image sur une fenêtre de taille (2W+1) * (2W+1), centrée sur le pixel de coordonnées (*j*, *k*) et pour une translation (d_1, d_2), la fonction d'autocorrélation peut être calculée de la manière suivante :

$$C(d1, d2, j, k) = \frac{\sum_{m=j-w}^{j+w} \sum_{n=k-w}^{k+w} g(m,n) * g(m-d_1, n-d_2)}{\sum_{m=j-w}^{j+w} \sum_{n=k-w}^{k+w} [g(m,n)]^2}$$

Ou :

$$d_1, d_2 = 0, \pm 1, \pm 2, \ldots, \pm T \qquad -T \leq d1, d2 \leq T$$

avec

T : un entier positif et g (m, n) le niveau de gris du pixel (m, n).

La fonction d'autocorrélation contient deux sortes d'informations qui sont résumées comme suite :

- Pour les grandes primitives la fonction d'autocorrélation varie lentement en augmentant la distance (d_1, d_2).
- Pour les petites primitives la fonction d'autocorrélation varie rapidement en augmentant toujours la distance (d_1, d_2) par contre pour les petites translations elles ne produisent pas une grande différence dans la comparaison. Une répétitivité des primitives entraine une périodicité de la fonction d'autocorrélation.

A partir de cette fonction, la signature d'éléments de la texture peut être calculée de la manière suivante :

$$C(j, k) = \sum_{d_1=-T}^{T} \sum_{d_2=-T}^{T} d_1^2 d_2^2 C(d_1, d_2, j, k)$$

Plus C (j, k) augmente, plus la texture et grossière.

I.5.2) Méthodes Spatio-fréquentielles

Cette approche exploite en même temps la représentation spatiale et la représentation fréquentielle de l'image. Une texture douce, pour laquelle on ne décèle des variations significatives du niveau de gris que sur un nombre important de points image, aura un spectre de puissance caractérisé par de fortes amplitudes dans le domaine des bases fréquences. Par contre pour une texture grossière et granuleuse, dans laquelle les variations du niveau de gris sont brutales des hautes fréquences, elles font appel entre autre aux transformées en ondelettes et aux filtres de Gabor.

❖ Filtres de Gabor

Les filtres de Gabor sont des filtres orientés passe-bande qui présentent d'excellentes propriétés de localisation fréquentielles et spatiales. Leur efficacité dans le domaine de l'analyse de la texture a été démontrée dans plusieurs travaux [18] [19] [14]. Ce filtre à une base sinusoïdale modulée par une gaussienne.

Sinus / cosinus : analyse en fréquence

Gaussienne : différents niveaux de lissage (échelles)

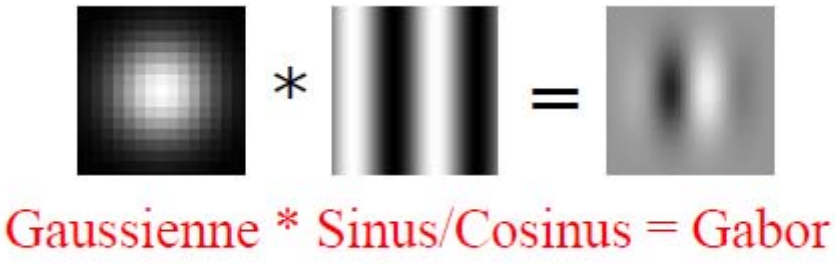


Figure (I.8) : Représentation d'un filtre de Gabor

C'est un filtre passe bande ayant un gabarit gaussien. Il est très rependu par sa propriété de résolution optimale conjointe en fréquence et en temps. La fonction de filtre de Gabor est donnée par :

$$f_{GB}(x', y') = g(x', y') \exp[2\pi j(\mu_0 (x-x_0)^2 + \gamma_0 (y-y_0)^2)]$$

avec

$$g(x', y') = \exp\left[-\frac{1}{2}\left(\frac{x'^2}{\sigma^2_\mu} + \frac{y'^2}{\sigma^2_\gamma}\right)\right]$$

Ou

σ_μ et σ_γ sont des constantes d'espace de l'enveloppe gaussienne qui déterminent l'étendue de l'ordre dans les axes *x* et *y* respectivement. Le point (x_0, y_0) définit le point d'origine où s'applique la fonction f_{Gb}. En ce point, la fonction f_{Gb} est maximale.

μ_0 et γ_0 sont des paramètres de position.

Le filtre de Gabor fait partie d'une famille de techniques d'analyses et caractérisations de texture qui décrivent le rendu visuel par un mélange de signaux de fréquence, d'amplitude et de direction différentes. Il permet l'extraction directe de caractéristiques de texture

localisées en fréquence et en orientation, en utilisant le principe de fenêtres gaussiennes, qui glissent sur l'image en capturant ses propriétés locales. Il s'appuie sur un banc de filtres très sélectifs en fréquence et en orientation.

I.5.3) Méthodes statistiques :

Dans ce type d'approche, la nature statique de la texture est prise en compte dans la procédure d'analyse, car en considère que la texture ne comporte pas des contours forts et ne possède pas un motif de base isolable, mais elle a au contraire une structure aléatoire (la texture est considérée comme la réalisation d'un processus stochastique stationnaire). Cette approche utilise le domaine spatiale et regroupe une multitude de méthodes comme

(Statistique d'ordre 1, matrice de cooccurrence).

- Matrice de cooccurrence

La matrice de cooccurrence de niveaux de gris a été initialement introduite par Haralick. C'est une méthode de second ordre puisqu'elle permet d'analyser la relation entre les niveaux de gris des pixels pris deux à deux. Elle considère comme la méthode de référence dans le domaine de l'analyse de la texture.

L'idée de cette méthode est d'identifier les répétitions de niveaux de gris de deux pixels séparés selon une direction θ et une distance d souvent égale à 1. Les directions les plus utilisées sont 0°, 45°, 90°, et 135°.

La matrice de cooccurrence M d'une région R pour une translation t est définie comme suite :

$$\left\{ M(i,j,d,\theta) = card\ \ (m, m+t) \in R^2 \quad g_m = i, g_{m+t} = j \right\}$$

Avec m représente un pixel

C'est une matrice de taille $N_g \times N_g$ ou :

N_g: st le niveau de gris maximale de l'image.

Quatorze indices ont été extraits de la matrice de cooccurrence pour définir une texture.

Les 5 paramètres les plus connus sont :

- **Contraste**

$$\mathrm{Cont(d\,,\theta)} = \sum_{\mathrm{i=0}}^{\mathrm{N_g-1}} \sum_{\mathrm{j=0}}^{\mathrm{N_g-1}} (\mathrm{i-j})^2 \mathrm{M(i,j,d,\theta)}^2$$

Chaque terme de la matrice M est pondéré par sa distance à la diagonale. On obtient un indice correspondant à la notion usuelle de contraste, il est élevé quand les termes éloignés de la diagonale de la matrice sont élevés, c'est-à-dire quand on passe souvent d'un pixel très clair à un pixel très foncé ou inversement.

- **Entropie**

$$ENT(d,\theta) = \sum_{i=0}^{N_g-1} \sum_{j=0}^{N_g-1} M(i,j,d,\theta)\ log\ M\ (i,j,d,\theta)$$

L'entropie est faible si on a souvent le même nombre de pixels, forte si chaque couple est peut représentéer. Elle fournit un indicateur de désordre que peut présenter une texture.

- **Corrélation**

$$Cor\ (d\,,\theta) = \frac{\sum_{i=0}^{N_g-1} \sum_{j=0}^{N_g-1} (i-\mu_x)(j-\mu_y)\ M\ (i\,,j\,,d,\theta)}{\sigma_x\ \sigma_y}$$

$$M_x(i) = \sum_{j=0}^{N_g-1} M(i\,,j\,) \qquad M_y(j) = \sum_{i=0}^{N_g-1} M(i,j)$$

$$\mu_x\,(i) = \sum_{j=0}^{N_g-1} jM_x\ (j) \qquad \mu_y\,(j) = \sum_{i=0}^{N_g-1} i\ M_y(i)$$

$$\sigma_x^2 = \sum_{i=0}^{N_g-1} (\,i-\mu_x\,)^2\ M_x\ (\mathrm{i}) \qquad \sigma_y^2 = \sum_{j=0}^{N_g-1} (j-\mu_y)^2\ M_y\,(j)$$

Ce paramètre quantifie la dépendance directionnelle des niveaux des gris, il atteint ses plus grandes valeurs lorsque θ est proche de l'orientation des lignes de la texture.

- **Homogénéité locale**

$$H(d,\theta) = \sum_{i=0}^{N_g-1} \sum_{j=0}^{N_g-1} \frac{1}{1+(i-j)^2}\ M(i,j,d,\theta)$$

Il reflète l'existence des plages uniforme de texture et affecte un poids de plus en plus faible au fur à mesure qu'on s'éloigne de la diagonale principale. Plus la valeur de H est élevée plus la texture est grossière.

- **Variance**

$$Var(d,\theta) = \sum_{i=0}^{N_g-1}\sum_{j=0}^{N_g-1}(i-j)^2 \mathrm{M}\,(\mathrm{i},\mathrm{j},\mathrm{d},\theta)$$

Ce paramètre mesure la somme des fréquences pour différents écarts de niveaux de gris.

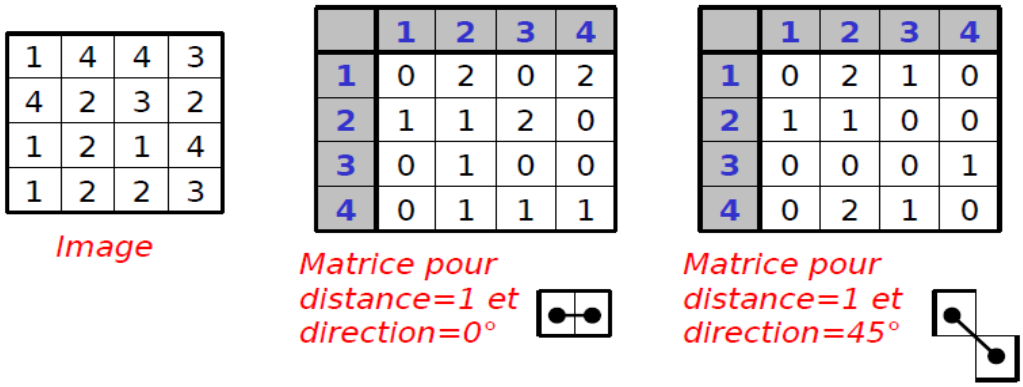


Figure (I .9): Exemple sur le principe de calcule d'une matrice de cooccurrence

I.5.4) Méthodes basées sur un modèle [22]

Le but de ces méthodes est d'obtenir un modèle linéaire de la texture. Les paramètres de ces modèles permettent de caractériser ou de synthétiser une texture. On distingue en générale trois types de modèles. Les modèles linéaires, les modèles autorégressifs, fractals et les modèles Markoviens.

- **modèle d'Autorégressif (AR) :**

On peut caractériser la dépendance entre les niveaux de gris d'une texture par un modèle auto régressif en considérant l'image comme une séquence temporelle de niveaux de gris dont la valeur à tout instant dépond des valeurs précédemment rencontrées au cours d'un balayage ligne par ligne. Un tel modèle permet de prévoir la valeur du niveau de gris d'un pixel en fonction des niveaux de gris de ses voisins.

Par ailleurs, la nature du champ d'entrée $e(i, j)$ définit le type du modèle. Dans le cas où $e(i, j)$ est un bruit blanc, le modèle sera appelé modèle AR 2D excité par un bruit blanc et

dans le cas où $e(i, j)$ est un bruit corrélé, le modèle sera alors un modèle d'auto régressifs à moyen ajustée (ARMA 2D) [Youl , 1994] et sa formule est comme suit :

$$x(i,j) = \sum_{(k,l)\,\epsilon\, D_s} a_{k,l}\,(i-k\,,j-l) + b_{0,0}\; e(\,i\,,j) + \sum_{(\,k\,,l\,)\in D_e} b_{k,l}\;\; e(\,i-k\,,j-l\,)$$

Ou

D_s : Représente le domaine de prédiction lié à la sortie du filtre

D_e : Représente le domaine de prédiction lié à l'entrée du filtre

e : d'écrit un processus de bruit

- Si les coefficients $b_{k,l}$ sont nuls pour tout (k, l) appartenant à D_e , alors le processus est AR
- Si les coefficients $a_{k,l}$ sont nuls pour tout (k, l) appartenant à D_s , alors le processus est MA

Le choix des domaines $D_s\ et\ D_e$ incombe à l'utilisateur. Dans de nombreux cas on choisit $D_s = D_e = D$. Afin de conserver aux modèles ARMA une bonne localité, il est souhaitable que le nombre de termes dans D soit faible, mais pour bien représenter des signaux à longue périodicité, il vaut mieux qu'il soit grand. Ces modèles expriment naturellement une dépendance causale.

Dans le cas d'un modèle AR, les coefficients $a(k, l)$ sont pris comme attributs de texture.

I.6) Segmentation d'images :

La segmentation d'image texturée est la phase la plus importante dans le traitement d'image, elle partitionne l'image en un ensemble de régions dont chacune d'entre elles possède au moins une caractéristique que les autres régions voisines ne possèdent pas.

La segmentation d'image texturée est la détection et localisation des différentes textures qui composent l'image. La difficulté repose dans l'élaboration des critères définissants les objets à segmenter.

De nombreux algorithmes ont ainsi été proposés durant les dernières décennies. Ils sont basé sur différentes approches contour, région, texture.

I.6.1) Définition :

Segmenter une image texturée revient à décomposer celle – ci en un ensemble de région ayant des caractéristiques texturales homogènes. Mathématiquement la segmentation est le partitionnement d'une image I en N régions, R_i (ensemble de pixels) disjointes et homogènes selon un prédicat ou critère *P déterminé (couleur, intensité, texture, niveau de gris, indice,……..) [1]*

l'ensemble des régions $\{R_1, R_2, \ldots\ldots R_N\}$ est une segmentation de l'image I si :

1) $\bigcup_{i=1}^{N} Ri = I$, i ∈ [1, N]
2) $R_i \cap R_j = \emptyset \quad \forall\, i \neq j$
3) P (R_i) = vrais $\quad \forall\, R_i$
4) P ($R_i \cup R_j$) = faux, $\forall\, i \neq j$ et R_i adjacente à R_j

La première condition implique que chaque pixel de l'image doit appartenir à une région R_i et l'union de toutes les régions correspond à l'image entière. La deuxième condition stipule l'intersection de deux régions distinctes égale à l'ensemble vide (c'est-à-dire qu'un pixel ne doit pas appartenir à deux régions différentes. La troisième condition est relative à la structure des régions, elle définit une région comme un ensemble de pixels qui doivent être connexes. La quatrième condition exprime le fait que chaque région doit respecter un prédicat d'uniformité P. La dernière condition implique la non réalisation de ce même prédicat pour la réunion de deux régions adjacentes.

I.6.2) Approches de segmentation d'image:[24],[1],[25]

La segmentation est une étape primordiale en traitement d'image, à ce jour, il existe de nombreuses méthodes de segmentation, que l'on peut regrouper en quatre principales classes :

1. Segmentation fondée sur les régions *(region-based segmentation)*. On y trouve par exemple : la croissance de région (*region-growing*), décomposition/fusion (*split and merge*).
2. Segmentation fondée sur les contours (*edge-based segmentation*)
3. Segmentation fondée sur la classification ou le seuillage des pixels en fonction de leur intensité (*classification* ou *thresholding*)

Les différentes approches de la segmentation sont résumées dans l'organigramme suivant :

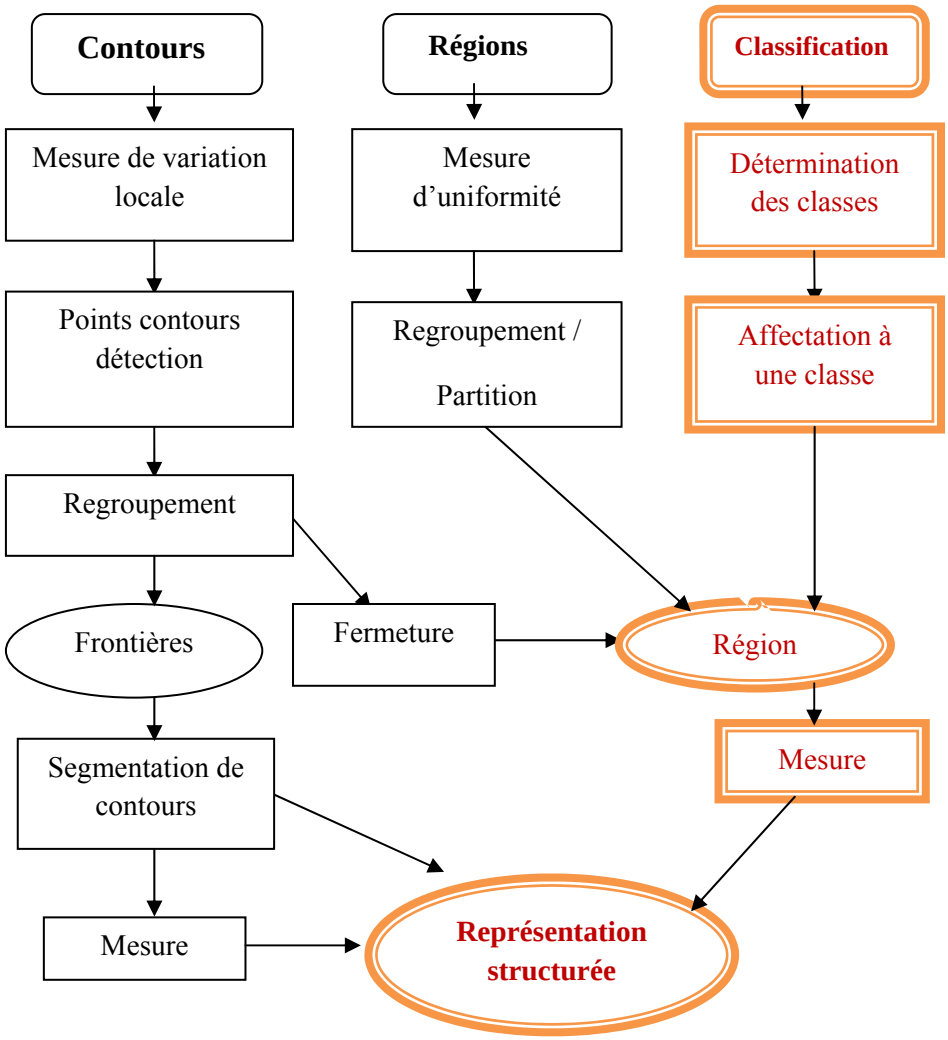


Figure (I.10) : Approche de segmentation d'image

I.6.2.1) Approche région :

1) Segmentation par fusion de région :

L'idée consiste à exploiter une partition initiale de l'image constituée de petites régions, puis ces régions sont fusionnées successivement jusqu'à ce que le critère de fusion ne soit pas vérifié.

Plusieurs règles de regroupement ont été proposées. Certaines de ces règles mettent en Jeu :

- Des propriétés statistiques telles que la moyenne ou la variance des niveaux de gris des régions, le gradient moyen des frontières des régions, le contraste maximum des régions, ou d'autres statistiques locales qui expriment l'état de surface des régions, …….etc.
- Des propriétés géométriques ou morphologiques telles que l'élongation ou la compacité des régions. Deux régions sont regroupées, si par exemple un facteur de forme est conservé ou amélioré après leur fusion.

2) Segmentation par division de régions :

Une autre approche pour réaliser la segmentation d'images est celle qui procède par partitionnement. Cette méthode consiste à diviser l'image, qui constitue la région initiale, en régions de plus en plus homogènes. Le processus est réitéré pour chacune des régions produites jusqu'à ce qu'une certaine homogénéité soit atteinte. L'homogénéité d'une région est souvent contrôlée par sa variance ou son contraste. Ces techniques à caractère descendant ont une faiblesse liée à la nature souvent régulière du découpage. Une région est divisée en sous-régions de niveaux inférieurs, les frontières d'une région sont alors représentées sur différents niveaux.

3) Segmentation par division-fusion (split and merge) :

L'algorithme division-fusion (Split and Merge) à été proposé par Horowitz Palliais [Bres03], il est encore actuellement l'un des plus performants. Le processus est décomposé en deux étapes.

L'image initiale peut être une première partition résultant d'une analyse grossière ou bien l'image brute. Dans la première étape, ou *division*, on analyse individuellement chaque région N. Si celle-ci ne vérifie pas le critère d'homogénéité, alors on divise cette région en blocs (le plus généralement en 4 quadrants) et l'on réitère le processus sur chaque sous-région prise individuellement, le découpage arbitraire peut conduire à ce que cette partition ne soit pas maximale. Dans la deuxième étape, ou *fusion*, on étudie tous les couples de régions voisines (x_k, x_i). Si l'union de ces deux régions vérifie le critère d'homogénéité, alors, on fusionne les Régions.

I.6.2.2) Approche par contours :

La segmentation par approche contours [6] [8] s'intéresse aux contours de l'objet dans l'image. La plupart des algorithmes qui lui sont associes sont locaux, c'est-adire qu'ils fonctionnent au niveau du pixel.

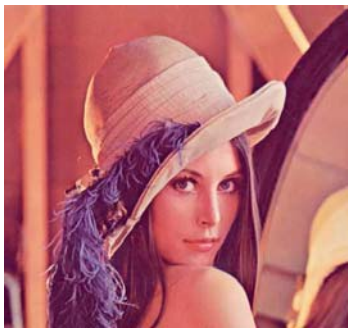
Des filtres détecteurs de contours sont appliqués à l'image et donnent généralement un résultat difficile à exploiter sauf si les images sont très contrastées.

Les contours extraits sont la plupart du temps morcèles et peu précis, il faut alors utiliser des techniques de reconstruction de contours par interpolation ou connaitre à priori la forme de l'objet recherché.

Formellement, ce type d'approche est proche des méthodes d'accroissement des régions fonctionnant au niveau du pixel. Ces techniques purement locales sont en général trop limitées pour traiter des images bruitées et complexes.

Dans la segmentation par approches contours, il y a deux problématiques à résoudre, a savoir :

- caractériser la frontière entre les régions :



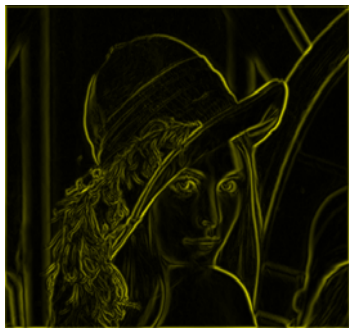


Figure (I. 11) Détection sur Lena de contour

- fermer les contours :

Figure(I.12) Illustration de contours à fermer Lena

1) Détection des contours :

La détection de contours est une étape préliminaire à de nombreuses applications de l'analyse d'images. Les contours constituent des indices riches, au même titre que les points d'intérêts, pour toute interprétation ultérieure de l'image.

Les contours dans une image proviennent des :

- Discontinuités de la fonction de reluctance (texture, ombre).
- Discontinuités de profondeur (bords de l'objet).

Pour les détecter, il existe deux types d'approches :

- Approche gradient : détermination des extrema locaux dans la direction du gradient.
- Approche Laplace : détermination des passages par zéro du Laplace.

Par souci de concision, nous ne développerons que les approches par gradient mais avant tout, définissons ce qu'est le gradient d'une image.

- **Le gradient d'une image :**

Un filtre de Gradient permet de mètre en évidence des variations des niveaux de gris suivant un axe variable, ce qui aura pour effet de mètre en évidence les fronts, révéler les textures. Pour une direction donnée, un filtre Gradient est utilisé pour augmenter ou bien réduire les fronts sur cette direction. Il est utilisé en particulier pour mètre en évidence des contours.

Le gradient d'une image est défini par :

$$\nabla I(x,y) = \frac{\partial I}{\partial x}(x,y) + \frac{\partial I}{\partial y}(x,y)$$

Parmi les opérateurs Gradient, on trouve les filtres de Préwiltt, Sobel et Robert, leurs masques sont définis comme suite

$\mathbf{G_X} = \begin{bmatrix} -1 & -C & -1 \\ 0 & 0 & 0 \\ 1 & C & 1 \end{bmatrix}$: Masque horizontale. $\mathbf{G_Y} = \begin{bmatrix} -1 & 0 & 1 \\ -C & 0 & C \\ -1 & 0 & 1 \end{bmatrix}$ Masque verticale.

C=1 : est un filtre de Perwitt

C=2 : est un filtre de sobel

Le détecteur de contours de *Roberts* est un filtre de convolution travaillant sur un voisinage 2×2 très rapide à calculer, il permet de calculer le gradient en un point en traitant successivement chacune des composantes du gradient calculées, suivants les directions 45° et 135° (au lieu de 0° et 90° comme Perwitt et Sobel)

Les masques de Roberts sont définis comme suite :

A= $\begin{bmatrix} +1 & 0 \\ 0 & -1 \end{bmatrix}$: Masque diagonale (135°)

B= $\begin{bmatrix} 0 & +1 \\ -1 & 0 \end{bmatrix}$: Masque anti-diagonal (45°)

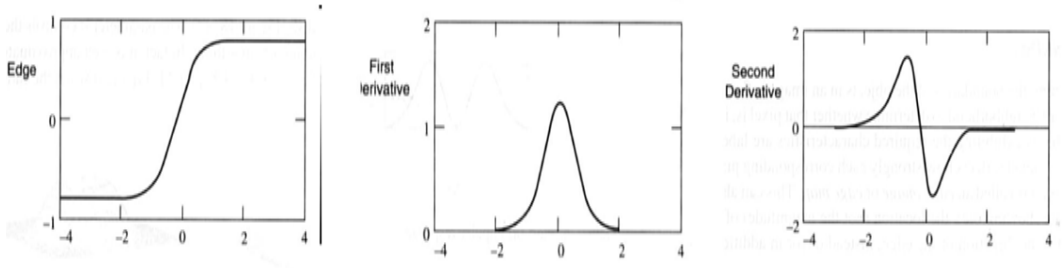
** ***Laplacien :***

Est une fonction scalaire en chaque point de l'image Laplacien est défini comme suite :

$$\nabla I(x,y) = \frac{\partial^2 I(x,y)}{\partial x^2} + \frac{\partial^2 I(x,y)}{\partial y^2}$$

Dans l'approche de *laplacien* le passage par zéro de la dérivée seconde du signal est considéré comme point contours. Le calcul de *laplacien* d'une image est toujours un produit de convolution qui peut être calculé directement par une convolution avec l'image.

La détection du passage par zéro du *laplacien* correspond au maximum ou minimum du gradient. Le *laplacien* agit comme un filtre passe-haut (très sensible aux petites variations non significatives et qui sont dues essentiellement à la présence du bruit), ce qui nécessite l'application d'un filtrage passe-bas avant le calcul du l*aplacien*.



❖ Une entrée b) 1eme dérivée c) 2eme dérivée

Figure (I.13) **:** les deux approches de détection de contours

2) Obtention des contours :

Une fois que les vecteurs gradients et *laplacien* sont calculés, les contours sont obtenus soit :

- En cherchant les points qui présentent un fort gradient qui correspondent aux maxima locaux de la norme gradient.
- En cherchant les points de passage par zéro du *laplacien*.

I.6.2.3) Approche par classification : [26]

La segmentation d'image par classification consiste à partitionner l'image en un ensemble de classe. Chaque classe regroupe des pixels ayants des vecteurs d'attributs de texture aussi similaire que possible et que les classes soient éloignées (en terme d'attributs) que possible les unes des autres. Les méthodes de classification cherchent à identifier les classes de pixels présentes dans l'image et affectent à chaque pixel un label (une Étiquette) indiquant la classe à laquelle il appartient.

Le processus de classification est donc réalisé par l'exécution des étapes suivantes :

- ❖ Représentation des données de la classification (définition des attributs des pixels servant à la classification tels que niveau de gris, paramètre de texture, couleur……etc.
- ❖ Définition d'une distance de similarité entre les pixels (exemple distance euclidienne entre les niveaux des pixels).
- ❖ Regroupement des pixels en classes.
- ❖ Evaluation de la classification obtenue.

Il n'existe pas une méthode de classification qui peut s'appliquer à différents types d'images et qui peut nous donner un nombre de classe exacte, ce qui explique la grande diversité de méthodes de classification qui existe dans la littérature. En effet il existe des méthodes qui prennent en compte qu'un seul attribut (niveau de gris) c'est ce qu'on appelle << *méthodes monodimensionnelles* >>, et des méthodes qui exploitent plusieurs attributs << *méthodes multidimensionnelles* >>.

On distingue deux types d'approches multidimensionnelles: la classification supervisée et la classification non supervisée.

1) Classification supervisée :

La classification est dite *supervisée* lorsque les différentes textures de l'image sont connues et l'appartenance de certains pixels (prototypes) à ces textures est connue à priori.

Dans ce cas, elle consiste à construire à partir de ces pixels (prototype) une fonction d'identification ou discrimination pour les autres pixels. Cette fonction d'identification réalise un découpage de l'espace de représentation. A chaque zone de ce découpage est affectée une classe de texture à priori. Les autres pixels sont ensuite classifiés en fonction de leur position dans l'espace des paramètres. Cette classification concerne les méthodes bayésiennes, telle que la méthode des K plus proche voisin.

- **Les méthodes des k plus proches voisin :**

C'est une méthode supervisée. On compare chaque vecteur d'attributs (niveau de gris du pixel par exemple) à ceux d'un voisinage immédiat en termes d'attributs et non en termes de voisinages partiale (selon un seuil prédéfini), On affecte alors le label de la classe dominante au pixel non assigné.
L'exemple suivant illustre le principe de la méthode :

On veut affecter le pixel X à l'une des classes, parmi les huit plus proches voisins du pixel X (en terme de niveau de gris), trois pixels appartiennent à la classe d'étiquette 3,
5 pixels appartiennent la classe d'étiquette 2. C'est donc l'étiquette 2 qui sera attribué au pixel X.

$$\begin{bmatrix} 6 & 4 & 8 & 10 & 12 \\ 7 & 2 & 3 & 2 & 5 \\ 9 & 3 & X & 2 & 9 \\ 0 & 2 & 2 & 3 & 8 \end{bmatrix}$$

Notons que la complexité de cette méthode réside essentiellement dans le choix de la distance de similarité qui impose le nombre des k plus proches voisins.

2) **Classification non supervisée** :

La classification non supervisée, telle qu'elle est définie en analyse de données, est un processus de regroupement d'éléments en un certain nombre de classes. Ce regroupement peut se faire soit sous forme de partition (on parle alors de classification par partitionnement) soit sous forme d'hiérarchie (on parle alors de classification hiérarchique)

2.1) Classification par partitionnement :

Cette méthode consistent à partitionner l'ensemble des pixels de l'image en N_C classes, l'objectif est alors de pouvoir regrouper automatiquement des pixels similaires dans une même classe (attributs de texture similaire). Dans ce cas il s'agit de définir une fonction de similarité entre pixels qui sera maximale entre les pixels d'une même classe et minimale comparée aux autres classes. La difficulté majeure de ces méthodes repose essentiellement sur la définition du nombre de classes N_C et la taille minimale d'une classe (nombre de pixels dans une classe).

2.2) Classification hiérarchique :

La classification hiérarchique est généralement réalisée par une agglomération ascendante de classe c'est à dire chaque entité (chaque pixel) est considéré comme classe de départ. Elle consiste dans un premier temps à regrouper des pixels dont deux dans les centres de gravité sont plus proches (mesure de la similarité) pour former des sous classes, puis faire regrouper les sous classes ainsi créées pour avoir des classes plus larges et plus homogènes, l'algorithme suivant illustre le principe de la méthode :

1) Etant donnée n entités (n pixels), initialiser $N c$ (nombre de classe) à n.

2) Déterminer les deux plus approches classes C_i et C_j selon une mesure de similarité appropriée.

3) Fusionner C_i et C_j et décrémenter $N c$ de 1.

4) Répéter (2) et (1) tant que les groupes formés ne sont pas homogène

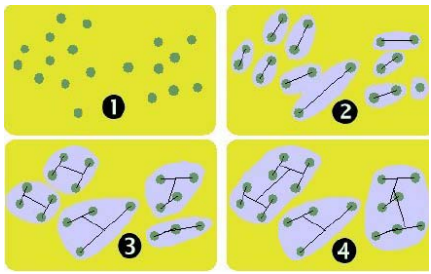


Figure (I.14) : exemple de classification hiérarchique

Dans le cadre non supervisé, les distributions à priori du champ aléatoire modélisant une image ne sont pas connues. Il faut donc une étape préliminaire afin d'estimer les paramètres relatifs à ces distributions.

Cette méthode se divise en deux familles :

- Les méthodes statistiques.
- Les méthodes métriques.

Méthodes statistiques :

Les méthodes statistiques consistent à rechercher les modes de la fonction de densité de probabilité (fdp) sous jacents à la distribution de l'ensemble de pixels à classifier. En effet, les centres des classes correspondent aux régions de l'espace multidimensionnel caractérisées par une forte concentration des données. Ceci se traduit par des fortes valeurs de la (fdp) dans ces régions alors que les régions constituées de peu de données correspondent aux faibles valeurs de la (fdp).

Méthodes métriques :

Parmi les méthodes de classification non supervisées les plus connues, on peut citer la méthode des ≪ centres mobiles ≫ ou la méthode des ≪ *K- means* ≫ **.** Dans cette

méthode, chaque classes C_K *(k = 1, ……, k),* (*k* étant le nombre totale des classes) est représenté par son centre de gravité noté u_k tel que μ_k $[\mu_1, \mu_2, \mu_3, \ldots\ldots, \mu_{knp}]^t$

np: est le nombre d'attributs de texture.

On note par $t_i = (t_{i1}, t_{i2}, t_{i3}, \ldots., t_{inp})^t$ le vecteur caractéristique de pixel « i », t_{ij} constitue alors l'attribut « j » du pixel « i ». Avec j = 1, ……, np.

- **Les k – moyen (k- means) [14], [27]**

La méthode des k-moyennes ou (*k- means*) est basée sur un algorithme itératif, elles consistent dans un premier temps à découper l'image en k zones (de façon aléatoire) représentant des classes de départ, on calcule ensuite le vecteur moyen de chacune des classes (centre de gravité) et on affecte chaque pixel de l'image dans la classe où le vecteur moyen est le plus proche. On répète l'opération jusqu'à un nombre maximal d'itérations ou jusqu'à ce que les vecteurs moyens recalculés ne varient plus de manière significative entre deux itérations. L'algorithme des *K- means* est illustré dans le paragraphe suivant :

- **Etape1 : Initialisation des centres de classes**

Initialiser au hasard ou manuellement les N_C centres de gravites [u_1, u_2, u_3,……,u_{Nc}] correspondant aux N_C classe.

- **Etape2 : Affectation :**

1.) affecter chaque pixel « i » à une classe C_k de centre u_k.

Un pixel « i » appartient à la classe C_k de centre u_k si seulement si $\| t_i - u_k \|$ est minimale pour tout k =1,……,N_c

2.) mettre à jour la position du centre de gravité u_k de la classe C_k .

$$u_{k,j} = \frac{1}{N_k} \sum_{i \in C_k} t_{ij}$$

où N_k est le nombre de pixels de la classe C_k.

- **Etape3** :

Aller à l'étape 2 si les centres des classes changent entre deux itérations successives ou si un nombre maximal n'est pas atteint.

L'algorithme des k-means nécessite la connaissance du nombre de classes de *k* et le choix des centres initiaux influe beaucoup sur la rapidité de la convergence ainsi que sur le résultat final de la classification. Il est à noter que lorsque deux centres de gravité sont proches, les deux classes associées ne sont pas fusionnées. D'autre part, un centre de gravité isolé dans l'espace des paramètres peut être associé à une classe qui ne contient aucun pixel.

Malgré ces inconvénients l'algorithme des *k- means* reste très utilisé en pratique.

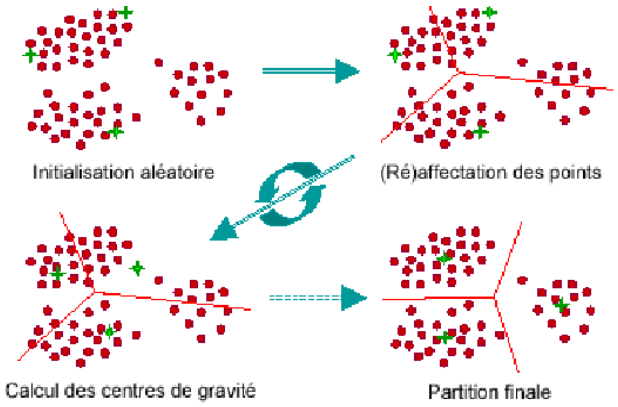


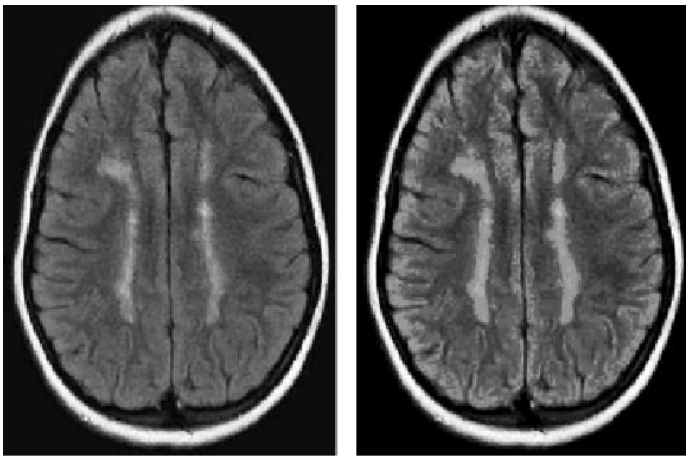
Figure (I.15) **:** Principe de l'algorithme de *K-means*.

❖ **Les principales étapes de l'algorithme *k-means* sont :**

1.) Choix des centres initiaux des K classes (clusters).
2.) (Ré-) Affecter les pixels à la classe la plus proche en utilisant la distance euclidienne.
3.) Recalculer les K centres des classes.
4.) Réitérer les étapes 2 et 3 jusqu'à ce que plus aucune réaffectation ne soit faite.

Cet algorithme est très populaire car elle est extrêmement rapide en pratique. En effet, le nombre d'itérations est typiquement inférieur aux nombres de points. En termes de performances, cet algorithme ne garantit pas un optimum global, la qualité de la solution dépend grandement des ensembles initiaux. Le nombre de classes est également exigé mais

cela n'étant pas réellement handicapant car dans certaines applications comme par exemple la segmentation cérébrale le nombre de classes est le plus souvent connu.



(a) (b)

Figure (I.16) : Résultat de classification obtenue par k-means

(a) image originale
(b) image segmentée en niveau de gris

I.8) Conclusion

Nous avons évoqué dans ce chapitre quelques notions sur la texture d'images. La texture permet de caractériser l'état d'une surface dans une image et qu' on peut la définir pour chaque pixel en fonction des niveaux de gris de ses pixels voisins. L'analyse de la texture est un domaine de recherche important dans le processus de traitement d'images et les domaines potentiels engendrant l'inspection des surfaces de l'imagerie biomédicale, industrielle et satellitaire ou aérienne (télédétection).

Plusieurs méthodes d'analyse de la texture on été développées, parmi ces méthodes, nous avons retenu celle qui est basée sur les modèles Auto régressifs que nous développerons plus en détails dans les chapitres qui suivent. Nous avons aussi présenté des généralités sur la segmentation d'image et nous avons adapté dans notre travail l'approche de la segmentation basée sur la classification. Nous avons utilisé l'algorithme de K-Means c'est le plus utilisé dans les recherches.

II. 1) Introduction :

Les modèles les plus connus sont le modèle Markovien, fractal et autorégressif (AR), le but de ces méthodes est d'obtenir un modèle générateur de la texture.

Le modèle autorégressif bidimensionnel (AR2D) est une extension du modèle monodimensionnel. Les développements théoriques concernant cette classe de modèle ont porté surtout sur les problèmes de stabilité et de réalisation. La stabilité des modèles 2D en général pose de grandes difficultés [Oztu, 1992], [Alat, 2003]. Ceci est dû aux problèmes algébriques de factorisation des polynômes à deux variables [Youl, 1994]. En effet, il n'existe pas de théorème qui assure que tout polynôme 2D est factorisable.

En dépit des problèmes que nous venons de citer, nous avons voulu mené notre propre étude et avons, ainsi, développé et mis en œuvre les applications texturales dérivés des modèles Autorégressifs bidimensionnels.

II.2) Modélisation [28] :

La modélisation réalisée à partir d'un comportement du système et \ou de lois physiques, consiste à déterminer la structure des équations qui régissent le comportement de ce système, et aussi à fixer, a priori la valeur de ces paramètres (longueurs, masses, inerties, capacités, résistances, frottements ……). Mais, il est souvent impossible d'obtenir une connaissance a priori complète et précise de tous les paramètres du modèle.

On distingue deux types de modélisation

- Modélisation non paramétrique.
- Modélisation paramétrique.

II.2.1) Modélisation non paramétrique :

Ce type de modélisation peut être obtenu par une analyse fréquentielle ou par la réponse impulsionnelle du système. Généralement il consiste à estimer sa densité spectrale .

II.2.2) Modélisation paramétrique :

La modélisation d'un signal par des techniques paramétriques nécessite une intervention humaine sur le modèle. Il faut, dans un premier temps, définir le modèle le mieux adapté au signal à modéliser.

Elle consiste à associer à un signal un modèle, représenté par un vecteur paramètre $\theta = [\theta_1 \ , \ \theta_2 \ , \dots , \theta_p]^T$ censé représenter au mieux le signal considéré .Etant donné que l'on choisit un modèle pour le signal, cela signifie qu'en générale on possède des informations a priori sur le signal lui – même qui permettent de sélectionner tel ou tel modèle.

Il s'ensuit que le choix d'un modèle plutôt d'un autre requiert au préalable une analyse du signal.

II.2.2.1) l'intérêt d'une modélisation paramétrique :

Dans de nombreuse applications on peut représenter un ensemble de N échantillons par un vecteur de dimension (P<<<N) et la modélisation paramétrique permet de réduire l'espace de représentation. Parmi ces applications on a :

- **Classification :** lorsque l'on désire classer des signaux suivant certain nombre de classes, on substitue au signal ses paramètres et on classifie suivant ces paramètres.

- **Détection :** si l'on veut détecter les ruptures dans un modèle, il suffit de suivre un petit nombre de paramètres pour déterminer l'instant de rupture.

- **Transmission – Compression :** au lieu de transmettre un signal, on ne transmet que les paramètres de modèle associé.

Elle permet d'extraire de façon plus fine (exacte) certaines informations. Par exemple le cas en analyse spectrale où la modélisation permet d'estimer avec une meilleure résolution le spectre d'un signal.

II.2.2. 2) Etape de la modélisation paramétrique [29] :

La modélisation paramétrique des signaux (et des systèmes) se compose des trois principales étapes représenté sur la figure (II .1)

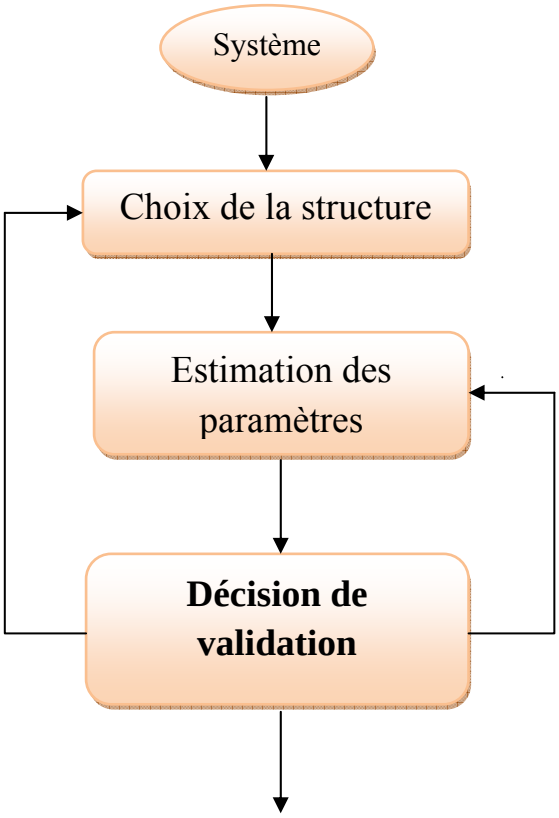


Figure (II.1) : Etape de la modélisation paramétrique

❖ **Choix de la structure du modèle :**

Ce choix est le résultat d'un compromis entre la complexité du modèle (ARMA, AR) et de sont aptitude à représenter correctement le système dans le domaine de l'étude.

❖ **Estimation des paramètres :**

L'estimation des paramètres du modèle sont utilisés de façon à minimiser un critère représentatif de l'équivalence signal- modèle.

❖ **Teste de validation du modèle :**

L'importance de cette étape réside dans la vérification de l'aptitude du modèle obtenu à représenter la classe de signaux considérés (vérification du choix de la structure et la méthode d'estimation). On suppose que le modèle estimé est correct et on utilise les propriétés théoriques du modèle pour extraire l'information utile.

II .3) Modèles linéaire [29], [30],[31],[32]

II.3.1 Modèle ARMA :

Un modèle général Auto- Régressif à Moyenne Ajustée d'ordre (p, q) ou ARMA (p, q),

(**A**uto **R**égressif **M**oving **A**verage), est composé de deux parties, une partie autorégressive AR et une partie mobile MA.

Un modèle ARMA peut être défini par la relation suivante :

$$x(n) = -\sum_{i=1}^{p} a(i)x(n-i) + \sum_{i=0}^{q} b(i)e(n-i) \qquad \text{(II.1)}$$

e(n) : l'entrée du filtre ARMA, considérée comme un bruit blanc.

x(n) : la sortie du modèle ARMA.

a(i), b(i) : paramètre de modèle ARMA.

Le modèle ARMA peut être représenté par le schéma bloc de la figure (II.2)

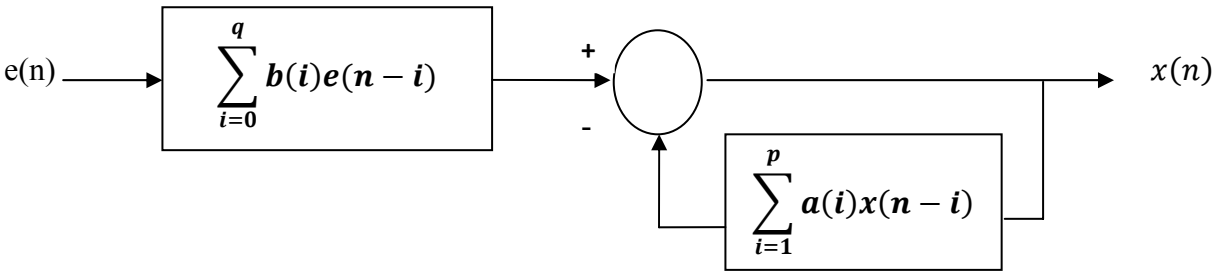


Figure (II.2) **:** schéma bloc de la représentation temporelle du modèle ARMA.

Pour a(0) = 0 le modèle est non causal.

Soit (z) , la transformée en z de la séquence x(n) est définie par :

$$X(z) = \sum_{n=-\infty}^{+\infty} x(n)z^{-n} \qquad \text{(II.2)}$$

Sachant que

$$TZ \quad x(n-1) = z^{-1}\ x(z) \qquad \text{(II.3)}$$

La TZ de l'équation (II.1) nous donne

$a_0 X(z) + a_1 z^{-1} X(z) + \ldots\ldots + a_p z^{-q} X(z) = b_0 E(z) + b_1 z^{-1} E(z) + \ldots\ldots + b_q z^{-q} E(z)$.

Soit :

$$\frac{X(z)}{E(z)} = \frac{b_0 + b_1 z^{-1} + \ldots\ldots\ldots + b_q z^{q}}{1 + a_1 z^{-1} + \ldots\ldots + a_p z^{-p}} = \frac{\sum_{k=1}^{q} b_k z^{-k}}{1+\sum_{k=1}^{p} a_k z^{-k}} = H(z) \qquad \text{(II.4)}$$

H(z) est la fonction de transfert en (Z) du modèle qui peut être représenté par la figure (II.3) suivante : dont B(Z) la sortie et A(Z) l'entrée du système

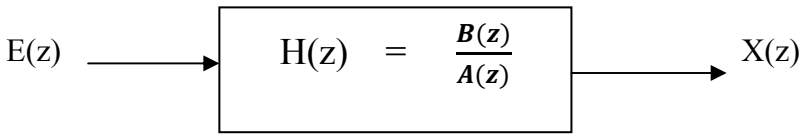


Figure (II.3) **:** Schéma bloc de la représentation spectrale du modèle ARMA

On peut interpréter le modèle ARMA comme un filtre de fonction de transfert H(z) excité par une entrée E(z) et délivrant à sa sortie un signal X(z) . C'est un filtre numérique à réponse impulsionnelle infinie. Pour assurer que le processus x(n) soit stationnaire il faut que H(z) soit stable et causal c'est-à-dire que A(z) ait ses zéro à l'intérieur du cercle unité.

II.3.2) Modèle à moyenne ajustée (MA)

Le processus à moyenne ajustée (MA) d'ordre q est une séquence aléatoire x(n), lorsqu'on peut l'écrire sous forme d'une moyenne pondérée courante de variables aléatoires non- corrélées

$$x(n) = \sum_{i=0}^{q} b(i)\ e(n-i) \qquad \text{(II.5)}$$

Ou :

$b(i)$: sont des paramètres du modèle MA

q : est l'ordre du modèle MA

$e(n)$: est un bruit centré de variance σ^2

Dans cette partie, les $a(i)$ sont nuls $a(i) = 0$ $i > 0$

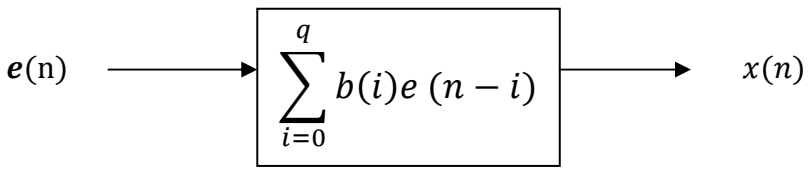


Figure (II.4) : Schéma bloc de la représentation temporelle du modèle MA

Le modèle MA est dit : modèle tout zéro, car ce dernier sera caractérisé par un polynôme ayant que des zéros, c'est aussi un filtre stable et dans le domaine spectrale la fonction de transfert est alors H(z) = B(z) et A(z) = 1.

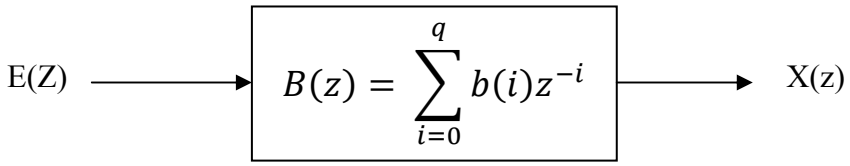


Figure (II.5) **:** Schéma bloc de la représentation spectrale du modèle MA

II.3.3) Modèle AR

Dans le cas où les coefficients b(i) du modèle ARMA est égale à 1 on obtiendra un modèle AR défini comme suit :

$$X(n) = -\sum_{i=1}^{p} a(i)x(n-i) + e(n) \qquad \text{(II.6)}$$

Et ce modèle (AR) est un filtre tout- pôle (composé uniquement de pôles) au travers duquel passe un bruit blanc.

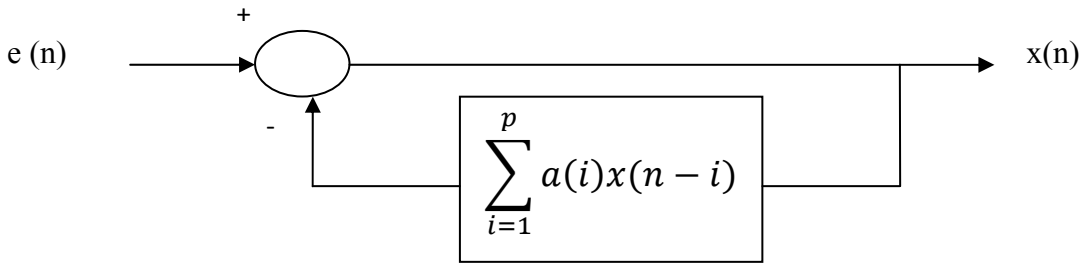


Figure (II.6) : schéma bloc d'une représentation temporelle du modèle AR

$a(i)$: i = 1…… p : sont les paramètres du modèle AR.

p: l'ordre du modèle AR

La fonction de transfert d'un filtre autorégressifs d'ordre p est donnée par :

$$H(z) = \frac{1}{1 + \sum_{i=1}^{p} a(i)\, z^{-i}} \qquad \text{(II.7)}$$

Après avoir étudier les modèles ARMA et MA plusieurs travaux ont comparé les performances obtenues par les types de modèle dans le traitement de signal. Vu la non linéarité de la modélisation par la partie MA et le risque d'obtention de modèles instables par la partie ARMA, la partie AR a pris faveur.

Dans la fonction (II.7), le polynôme B(z) = 1 et la fonction de transfert H(z) ne contient Que des pôles, d'où l'appellation du modèle tout-pôles.

II.4) Utilisation des modèles AR :

On utilise le modèle d'autorégressif dans les cas suivant :

- La modélisation et l'identification des processus.
- Le calcul de la densité spectrale.
- La prédiction linéaire d'un signal.

❖ **Le calcul de la densité spectrale :**

La fonction d'autocorrélation du signal que nous supposerons réel, stationnaire et de moyenne nulle est :

$$r_x(m) = E[x(n)\,x(n+m)] \qquad \text{(II.8)}$$

Ce qui est une fonction symétrique par rapport à l'origine

$$r_x(m) = r_x(-m) \qquad \text{(II.9)}$$

La fonction de Fourier de cette fonction d'autocorrélation est la densité spectrale $S_x(Z)$ qui est une fonction réelle positive ou nulle. Cette densité spectrale est la moyenne du module de la transformée de Fourier du signal x(n), sa formule est donnée comme suit :

$$\begin{cases} S_x(z) = \dfrac{\sigma_e^2}{A_p(z)\,A_p(z^{-1})} \\ Avec \\ A_p(z) = 1 - \sum_{i=1}^{p} a(i)\,z^{-1} \end{cases} \qquad \text{(II.10)}$$

❖ **La prédiction linéaire :**

Il s'agit ici de modéliser le signal directement dans l'espace des temps et non pas dans l'espace des fréquences .

Si x(n) représente la série monodimensionnelle temporelle mesurée, alors le modèle est celui représenté par la formule d'autorégressif (II.6).

On voit qu'un point de la série de données peut être ainsi prédit par une relation linéaire à partir des points précédents.

II.5) Estimation des coefficients AR :

Il existe un problème dans la modélisation AR qui revient à déterminer les coefficients du modèle ou du filtre tout-pôles dont on connait le signal de sortie et celui de l'entrée, il est nécessaire d'adopter un critère .le critère classiquement utilisé est celui de la minimisation de l'énergie de l'erreur de prédiction.

Ce problème peut être résolu par la méthode des moindres carrés, de levinson, de burg ou de covariance.

II.5.1) Equation de Yule Walker

Dans le cas d'un model d'autorégressif (AR) on a :

$$X(n) = -\sum_{i=1}^{p} a(i)x(n-i) + e(n) \qquad \text{(II.11)}$$

Dont x(n) est une sortie d'un système et invariant dans le temps dont l'entrée e(n) est un bruit blanc stationnaire de moyenne nulle et de variance σ_e^2

Le principe des équations de Yule Walker est d'exprimer les paramètres AR inconnus en fonction de la fonction d'auto corrélation $r_x(m)$.

❖ En multipliant ainsi les deux membres de l'équation (II.11) par x(n-m) et en prenant l'espérance, on obtient

$$E[x(n)x(n-m) = E[-\sum_{i=1}^{p} a(i)x(n-i)x(n-m)] + E[e(n)x(n-m)] \qquad \text{(II.12)}$$

$$E[x(n)x(n-m)] = -\sum_{i=1}^{p} a(i)E(x(n-i)x(n-m)] + E[e(n)x(n-m)] \qquad \text{(II.13)}$$

Or $E[x(n)x(n-m)]$ n'est rien d'autre que l'auto corrélation du signal x(n) notée $r_x(m$

Si le signal est ergodique la fonction d'auto corrélation sera comme suite :

$$r_x(m) = \frac{1}{N-m} \sum_{i=0}^{N-m} x(i)\, x(i+m) \qquad \text{(II.14)}$$

Lorsque $m > 0$, $E[e(n)x(n-m) = 0\ (n)$ est décorrèle avec le passé de $x(n)$

Pour $m = 0$ on a :

$$E[(e(n)x(n)]=E[e(n)(-\sum_{n=1}^{p} a(i)x(n-i) + e(n))] = \sum_{i=1}^{p} -a(i)E[e(n)x(n-i)] +$$

$$E\left(e^2(n)\right) = 0 + \sigma_e^2 \qquad \text{(II.15)}$$

On a pour $m \geq 0$, en regroupant les deux cas précédents

$$r_x(m) = E[\sum_{n=1}^{p} -a(i)x(n-i)] + \sigma_e^2 \delta_m \qquad \text{(II.16)}$$

Avec $\delta_m = \begin{cases} 1\ si\ m = 0 \\ 0\ si\ m > 0 \end{cases}$

Par ailleurs, on a :

$$E[-\sum_{i=1}^{p} a(i)\ \ x(n-i)\, x(n-m)] = -\sum_{i=1}^{p} a(i)\ E[\, x(n)\ \ x(n-m+i)\,] =$$
$$-\ \sum_{i=1}^{p} a(i)\, r_x\ (m-i)$$

(II.17)

- En regroupant les deux termes de l'équation (II.17) on obtient les équations dites de **Yule Walker** :

$$r_x(m) = -\sum_{i=1}^{p} a(i)\, r_x\ (m-i) + \sigma_e^2\, \delta_m \qquad \text{(II.18)}$$

$$\Longrightarrow \sum_{i=0}^{p} a(i)\ r_x\ (m-i) = \sigma_e^2\ \delta_m \qquad \text{(II.19)}$$

$$avec \qquad a\ (0) = 1$$

Et elle peut s'écrire sous forme matricielle :

$$\begin{bmatrix} r_x(0) & r_x(-1) & \cdots & r_x(-p) \\ r_x(1) & r_x(0) & & r_x(1-p) \\ \vdots & \vdots & \cdots & \vdots \\ r_x(p) & r_x(p-1) & & r_x(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_1 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} \sigma_e^2 \\ 0 \\ 0 \\ 0 \end{bmatrix} \qquad \text{(II.20)}$$

la fonction d'auto corrélation est paire, pour m < 0

$$r_x(m) = r_x(-m) = \sum_{i=1}^{p} a(i)\, r_x\,(|m| - i) + \sigma_e^2\ \delta_m \qquad \text{(II.21)}$$

Il faut résoudre le système d'équation pour déterminer les coefficients AR ou bien en éliminant la première ligne et la première colonne et le système devient :

$$Ra = -r \qquad \text{(II.22)}$$

$$R = \begin{bmatrix} r_x(0) & r_x(-1) & \dots & r_x(-p+1) \\ r_x(1) & r_x(0) & \dots & r_x(-p+2) \\ \vdots & \vdots & \dots & \vdots \\ r_x(p-1) & r_x(p-2) & \dots & r_x(0) \end{bmatrix} \qquad \text{(II.23)}$$

$$a = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} \qquad \text{(II.24)}$$

$$r = \begin{bmatrix} r_x(1) \\ r_x(2) \\ \vdots \\ r_x(p) \end{bmatrix} \qquad \text{(II.25)}$$

Puisque le système est constitué des équations linéaires donc on peut les résoudre par des méthodes classiques comme (Gausse, jordan , jacobi, gausse seidel, cholesky, ect …….). et ces méthodes exigent un nombre important d'opérations.

la matrice R est définie comme suite :

- ✓ Définie positive
- ✓ Hermitienne $r_x^*(m) = r_x$ (-m) (dans le cas des signaux reels la matrice R est symétrique)
- ✓ Toeplitz : les éléments de chaque diagonale (principale et secondaire) sont égaux.

La résolution du système se fait alors d'une matrice efficace grâce à l'algorithme de **Levinson-Durbin**

II.5.1.1) Algorithme de Levinson-Durbin

Cette algorithme consiste à exprimer les paramètres d'ordre k en fonction des paramètres d'ordre k-1, les paramètres a(i), ainsi les variances de l'erreur sont déterminées d'une manière récursive.

Comme suite :

1) **Initialisation :**

K =1

$\boldsymbol{a_1}[\mathbf{1}] = \frac{-r_1(1)}{r_x(0)}$; $\boldsymbol{\delta_1^2} = (1 - |a_1[1]|^2 r_x(0)$

1) **Recursion:**

Pour k=2,……, p

$$a_k[k] = \frac{-1}{\delta_{k-1}^2}[r_x(k) + \sum_{l=1}^{k-1} a_{k-1}[l] r_x(k-l)]$$

$$a_k[l] = a_{k\_1}[l] + a_k[k] a_{k-1}^*[k-l]$$

$$\sigma_k^2 = (1 - |a_k[k]|^2)\sigma_{k-1}^2$$

2) Finalement, on déduit:

$\sigma_P^2 = \sigma_e^2$ et $a_p[1] = a(1)$

⋮

$a_p[p] = a(p)$

Remarque :

- L'Algorithme exige p^2 operations.
- Les coefficients $a_k[k]$ sont appelés ***coefficients de réflexion***
- Le meilleur prédicteur d'ordre p d'un processus AR (p) est donné par les paramètres AR. Autrement dit celui qui rend l'erreur orthogonale aux données. En effet, Pour minimiser l'erreur entre x(n) une combinaison linéaire des $\{x(n-i)\}$, $i = 1 \ldots \ldots$, p, il faut projeter orthogonalement x(n) sur l'espace engendré par $\{x(n-i)\}$, $i = 1, \ldots \ldots, p$.

De fait, l'erreur de prédiction devient orthogonale aux $\{x(n-i)\}, i = 1, \dots\dots, p$.

Ceci peut se retrouver en écrivant que, $\forall i \in [1, p\ ]$.

$$\left\{ E\ \{e(n)x*(n-i)\} = E\left\{\sum_{i=0}^{p} a(i)*x(n-i)x*(n-i)\right\} = \sum_{i=0}^{p} a(i)r_x(n-i) = 0\right\}$$

(II.26)

Les coefficients optimaux a (i) qui se trouvent dans la fonction(II.19) qui rend l'erreur de prédiction e(n) orthogonale (non corrélée) aux données$\{ x(n-1), \dots\dots, x(n-p) \}$. L'erreur de prédiction e(n) peut être vue comme l'information qui est contenue dans x(n) et qui n'était pas dans $\{x(n-1), \dots\dots\dots, x(n-p)\}$.

II.5.1.2 Algorithme de Burg :

Les résultats obtenus par l'algorithme de Levinson-Durbin (les coefficients de réflexion) peuvent être supérieure à 1 et cela entrainera une instabilité du filtre AR.

pour éliminer cet inconvénient Burg à proposé une autre solution appelée méthode de maximum d'entropie.

Si on suppose que $\{ r_x(i) \}\ i = 0, \dots\dots\dots, p$ sont connus, alors la fonction d'auto corrélation est extraite par les équations de Yule - Walker pour un processus AR.

Pour un processus quelconque Burg à posé la question suivante :

Si l'on connait $\{ r_x\ (i)\}\ i = 0, \dots\dots\dots, p$ comment est il possible d'extrapoler $r_x\ (p+1)$, $r_x\ (p+2) \dots\dots$, de manière à garantir le caractère semi- défini positif de la fonction de corrélation ?

Burg à montré que parmi toutes les extrapolations possibles, celle qui donne l'entropie maximale qui correspond a un processus AR, et ce dernier c'est le processus le plus aléatoire qui permet d'extrapoler la fonction d'auto corrélation à partir de la connaissance de $\{ r_x(i)\}\quad i = 0, \dots\dots\dots, p$

D'après Burg cette propriété est importante dans la mesure où elle permet de faire le moins d'hypothèses possible (la corrélation nulle) sur la fonction d'auto corrélation en dehors de $\{ r_x(i)\}\ i = 0, \dots\dots, p$, la méthode du maximum d'entropie est souvent assimilée a la modélisation AR.

L'algorithme de Burg utilise la contrainte de Levinson- Durbin sur les coefficients Autorégressifs et optimise les coefficients de réflexion par minimisation de la moyenne quadratique des erreurs de prédiction directe (ou progressive) et rétrograde.

$$E = \frac{1}{2} \sum_{n=p}^{N-1} \left| e_p^f(n) \right|^2 + \left| e_p^b(n) \right|^2 \quad \text{(II.27)}$$

Ou :

$e_p^f(n)$: représente l'erreur de prédiction directe à l'ordre p.

$$e_p^f(n) = x(n) - \hat{x}(n) \quad \text{(II.28)}$$

avec

$$\hat{x}(n) = \sum_{m=1}^{p} a(m)\, x(n-m) \quad \text{(II.29)}$$

Burg à aussi montré que :

$$e_p^f(n) = x(n) + \sum_{m=1}^{p} a_p[m] x(n-m) = e_{p-1}^f(n) + k_p e_{p-1}^b(n-1) \quad \text{(II.30)}$$

$e_p^b(n)$: est l'erreur rétrograde définie comme étant l'erreur commise l'ors qu'un échantillon est prédit par une combinaison linéaire des échantillons postérieurs.

$$e_p^b(n) = x(n-p) + \sum_{m=1}^{p} a_p^*[m]\, x(n-p+m) = e_{p-1}^b(n-1) + k_p^* e_{p-1}^f(n) \quad \text{(II.31)}$$

$$e_p^b(n) = x(n-p) - \hat{x}(n-p) \quad \text{(II.32)}$$

avec :

$$\hat{x}(n-p) = -\sum_{m=1}^{p} a_m^*\, x(n+m-p) \quad \text{(II.33)}$$

$$k_p = \frac{-2 \sum_{n=p}^{N-1} e_{p-1}^f(n)\, e_{p-1}^{b*}(n-1)}{\sum_{n=p}^{N-1} \left| e_{p-1}^f(n) \right|^2 + \left| e_{p-1}^b(n-1) \right|^2} \quad \text{(II.34)}$$

❖ L'algorithme de Burg est résumé comme suit :

Initialisation

$$r_x(0) = \frac{1}{N}\sum_{n=0}^{N-1} |x(n)|^2$$

$$\sigma_0 = r_x(0)$$

$$e_0^f(i) = x(i) \qquad i = 1, 2, \ldots\ldots\ldots, N-1$$

$$e_0^b = x(i) \qquad i = 0,1, \ldots\ldots\ldots., N-1$$

Calcule des coefficients de réflexion

Pour $k = 1,2, \ldots\ldots\ldots, p$

$$a_k[k] = k_k = \frac{-2\sum_{i=k}^{N-1} e_{k-1}^f(n)\, e_{k-1}^{b*}(n-1)}{\sum_{i=k}^{N-1} |e_{k-1}^f(n)|^2 + |e_{k-1}^b(n-1)|^2}$$

Calcul de la récurion de levinson

$$a_k[j] = a_{k-1}[j] + a_k[k]\, a_{k-1}[k-j] \qquad j = 1, \ldots\ldots, k$$

$$\sigma_k^2 = (1 - |a_k[k]|^2)\sigma_{k-1}^2$$

Calcul erreurs

$$e_k^f(i) = e_{k-1}^f(i) + a_k[k] e_{k-1}^b(i-1) \qquad i = k+1, \ldots., N-1$$

$$e_k^b = e_{k-1}^f(i-1) + a_k[k]\, e_{k-1}^f(i) \qquad i = k, \ldots\ldots.., N-2$$

II.5.1.3) Méthode de covariance

La méthode de Yule-Walker minimise l'erreur de Prédiction sur l'ensemble des données *x(n)* ce qui oblige à compléter la séquence par des zéros au début et à la fin des données.

Le calcule des coefficients AR par la méthode de covariance consiste à minimiser l'erreur suivante :

$$\sum_{N=0}^{N+p-1} e^{2}(n) = \sum_{N=0}^{N+p-1}(x(n) + \sum_{i=1}^{p} a(i)\, x(n-i))^2 \qquad \text{(II.35)}$$

Cette minimisation aboutit au système d'équation suivant :

$$\begin{bmatrix} c_x(1,1) & c_x(1,2) & \cdots & c_x(1,p) \\ c_x(2,1) & c_x(2,2) & \cdots & c_x(2,p) \\ \vdots & \vdots & \cdots & \vdots \\ c_x(p,1) & c_x(p,2) & \cdots & c_x(p,p) \end{bmatrix} \begin{bmatrix} a(1) \\ a(2) \\ \vdots \\ a(p) \end{bmatrix} = \begin{bmatrix} c_x(1,0) \\ c_x(2,0) \\ \vdots \\ c_x(p,0) \end{bmatrix} \qquad \text{(II.36)}$$

Ou

C_x (j, k) représente le coefficient de la covariance du signal *x(n)*. Si *x(n)* est ergodique, on a :

$$C_x(j,k) = \frac{1}{N-P} \sum_{n=P}^{N-1} x(n-j)x(n-k) \qquad \text{(II.37)}$$

$0 \leq j \leq p$ et $0 \leq k \leq p$

La résolution du système d'équations linéaires nous permet de déterminer les coefficients *a(i)* (i = 1,…….,p) .La variance du bruit d'entrée σ_e^2 peut être estimer par :

$$\sigma_e^2 = \min\left(\sum_{N=p}^{N-1} e^2(n)\right) = c_x(0,0) + \sum_{i=1}^{p} a(i)\, c_x(0,i) \qquad \text{(II.38)}$$

L'algorithme de levison-Durbin ne peut pas être appliqué pour résoudre le système car la matrice de covariance n'est pas du type Toelplitz.

Par contre l'algorithme de Cholesky peut être utilisé pour la résolution de système car la matrice de covariance est symétrique définie positive.

On remarque la stabilité de filtre ainsi calculée par cette méthode n'est pas assurée

Pour éviter ce problème, la méthode de covariance modifiée propose de minimiser la somme des erreurs de prédiction directe et rétrograde comme dans l'algorithme de Burg .

$$\sum_{n=p}^{N-1}[e_p^f(n)]^2 + \sum_{n=0}^{N-1-p}[e_p^b(n)]^2 \qquad \text{(II.39)}$$

La minimisation de cette équation aboutit au même système d'équation précédent sauf que :

$$C_x(j,k) = \frac{1}{2(N-p)}\left[\sum_{n=p}^{N-1} x(n-j)x(n-k) + \sum_{n=0}^{N-1-p} x(n+j)x(n+p)\right] \qquad \text{(II.40)}$$

II.5.1.4 Méthode des moindres carrés

L'estimation des coefficients AR peut être aussi obtenue en minimisant l'erreur :

$$e(n) = x(n) + \sum_{i=1}^{p} a(i)x(n-i) \qquad \text{(II.41)}$$

Nous disposons initialement de N échantillons de *x(n)* avec n=0,1…..,N-1, pour calculer les coefficients *a(i)* (i=1,……,p), on considère différentes valeurs n = p, p+1,…,N-1

On obtient ainsi le système d'équation suivant :

$$e(p) = x(p) + \sum_{i=1}^{p} a(i)x(p-i)$$

$$e(p+1) = x(p+1)\sum_{i=1}^{p} a(i)x(p+1-i)$$

$$\vdots$$

$$\vdots$$

$$e(N-1) = x(N-1) + \sum_{i=1}^{p} a(i)x(N-1-i)$$

Ce système peut s'écrire sous forme matricielle :

$$e = x - XA \qquad \text{(II.42)}$$

Avec :

$e = [e(p)\ e(p+1)\ \ldots.e(N-1)]^T$ le vecteur erreur.

$A = [a(1)\ a(2)\ \ldots.a(p)]^T$ le vecteur des coefficients AR qu'on cherche.

$x = [x(p)\ x(p+1)\ \ldots x(N-1)]^T$ le vecteur d'échantillons de *x(n)*.

et X une matrice telle que :

$$X = -\begin{bmatrix} x(p-1) & x(p-2) & \cdots & x(0) \\ x(p) & x(p-1) & \cdots & x(1) \\ \vdots & \vdots & \cdots & \vdots \\ x(N-2) & x(N-3) & \cdots & x(N-1-p) \end{bmatrix} \qquad \text{(II.43)}$$

L'estimation de vecteur A consiste à minimiser l'erreur quadratique $e^T e$ cela revient à résoudre :

$$\frac{\partial e^T e}{\partial A} = 0 \qquad \text{(II.44)}$$

Ou résoudre le système suivant :

$$X^T\ XA = X^T\ x \qquad \text{(II.45)}$$

Par l'une des méthodes de résolution des systèmes d'équations linéaires ou par inversion directe.

La solution dans ce cas est donnée par :

$$A = y^{-1}\, X^T\, x \tag{II.46}$$

Avec :

$$y = X^T\, X \tag{II.47}$$

$$A = (X^T\, X)^{-1}\, X^T\, x \tag{II.48}$$

II.6) Ordre du modèle AR [31]

L'ordre p du modèle AR peut être fixé arbitrairement. Cependant, il doit être choisi de manière précise

- Si l'ordre est trop faible, le modèle ne représentera pas les propriétés intrinsèques du modèle.
- Si l'ordre est trop élevé, le modèle représentera les propriétés du signal dues aux bruits.

Il existe plusieurs critères de sélection de l'ordre p du modèle AR qui sont basés sur l'analyse statistique, basés sur la variance de l'erreur de prédiction $\sigma_e^2(p)$ pour chaque valeur de P, parmi ces critères on peut citer :

- Critère FPE (Erreur de Prédiction Finale ou Final Prédiction Error) :

$$\text{FPE} = \frac{N+p+1}{N-p-1}\ \sigma_e^2(p) \tag{II.49}$$

N : représente le nombre d'échantillons du signal x(n).

- Critère AIC (Akaike Information Criterion) :

$$\text{AIC (p)} = \text{n log}\,(\sigma_e^2(p)) + 2p \tag{II.49}$$

La valeur minimale de FPE et d'AIC en fonction de p donne l'ordre du modèle. Ces critères sont très utilisés en pratique malgré le fait qu'ils ont tendance à sous estimer l'ordre du système.

II.7) Modèle autorégressif bidimensionnel [31]

Dans une image les pixels sont sur une grille de donnée au lieu d'une séquence, il est plus naturel de baser le traitement sur une représentation bidimensionnelle (dans le domaine spatial) cette représentation entraine un choix sur l'ordonnancement de différents pixels, la définition des notions de $\ll présent \gg$, $\ll passé \gg et \ \ll future \gg$ qui caractérisent facilement la causalité ne présente aucune ambiguité pour un signal monodimensionnel, ici il ya lieu d'opérer un choix qui nous amène à définir plusieurs types de modèle qui auront une structure causale , semi causale et non causale .

II.7.1) Définition

Un modèle autorégressif bidimensionnel est définit par l'équation suivante :

$$x(i,j) = -\sum_{(m,n)\in D} a(m,n,i,j)\, x\,(i-m, j-n) + e(i,j) \qquad \text{(II.50)}$$

Ou

Les quantités : a(m, n, i, j) sont les coefficients du modèle AR

D : C'est la région de prédiction définie par l'ensemble des couples (m, n) qui dépendent du pixel (i, j) et définit aussi le type de causalité du modèle.

e (i, j) est l'entrée du modèle qui peut être un bruit blanc ou une séquence aléatoire de fonction de densité spectrale connue.

Si e(i ,j) est un bruit blanc le modèle sera appelé modèle AR et dans le cas ou e(i ,j) est un bruit corrélé le modèle sera un modèle ARMA , et si e(i, j) est une erreur de prédiction alors le modèle sera appelé modèle de prédiction ou bien modèle de variance minimale

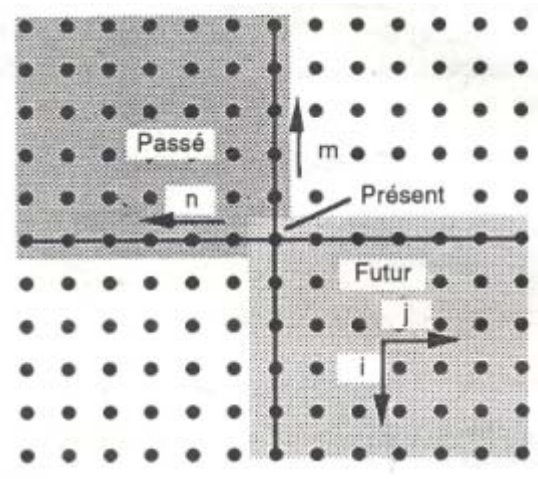
II.7.2) Causalité des systèmes bidimensionnels

- **Causalité quart de plan**

La causalité quart de plan est la plus simple à envisager, car c'est une extension directe du concept de causalité dans le cas monodimensionnel , l'ensemble D est défini comme suite :

$$D = \{\, m \geq 0, n \geq 0 \ et \ m + n > 0 \ \} \qquad \text{(II.51)}$$

Pour un point de l'image x (i, j) on définit le passé et le futur comme cela est représenté sur cette figure (II.1) et l'ensemble D se trouve dans le $\ll passé \gg$.



Figure(II.1) Causalité quart de plan

❖ **Causalité demi plan asymétrique**

Cette définition de la causalité provient de la manière usuelle de lire et de stocker une image dans le sens usuel de lecture, de gauche à droite et de haut en bas. L'ensemble de D est définit comme suite :

$$D = \{ (\ m > 0\ , \forall\, n\ )\ \} \cup \{ (\ m = 0\ ,\ n >> 0\ )\ \} \qquad \text{(II.52)}$$

Et comme le montre la figure II.2 :

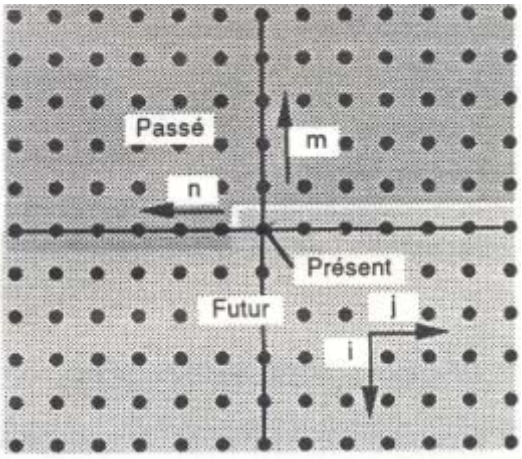


Figure II.2 causalité semi asymétrique

❖ **Semi – causalité et non causalité**

Dans le cas bidimensionnel on peut se trouver dans la situation où le modèle est causal dans une direction et non causal dans l'autre, c'est ce qu'on appelle un modèle semi-causal (Figure. II.3) ou l'ensemble D est définit comme suit :

$$D = \{ (m > 0, \forall n) \} \cup \{ (m = 0, n \neq 0) \} \qquad \text{(II.52)}$$

Dans le cas non causal, le modèle n'est causal dans aucune direction (figure II.2) et l'ensemble D est définit aussi comme suite :

$$D = \{ \quad \forall \ (m, n) \neq (0,0) \} \qquad \text{(II.53)}$$

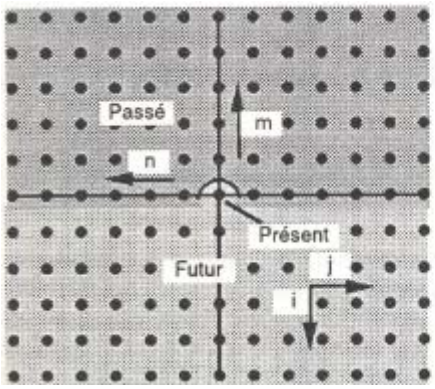


Figure (II.3) semi- causalité

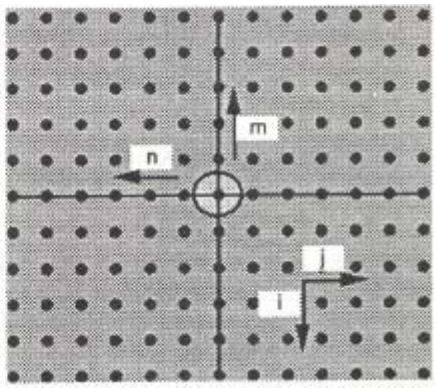


Figure (II.4) non causalité

La causalité quart de plan parait plus naturelle que la causalité demi plan asymétrique car c'est une extension directe du cas monodimensionnel, mais elle est plus restrictive (réduis). la causalité demi plan asymétrique est plus large et peut être obtenue à partir de la causalité monodimensionnelle par une simple transformation d'une grille de données en une séquence monodimensionnelle. Cette transformation est très importante car elle permet d'utiliser plusieurs résultats établis dans le cas monodimensionnel .Elle est définie de la manière suivante :

Soit x (i, j) un champ de données balayé ligne par ligne, a un point (i, j) on fait correspondre un instant ou indice de position noté ***t*** tel que :

$$(i, j) \rightarrow t = i + j\,I \qquad \text{(II.54)}$$

Avec

$$0 \leq i < I \quad et \quad 0 \leq j < J$$

I : nombre de lignes $0 \leq i \leq I$

J : nombre de colonnes $0 \leq j \leq J$

II.7.3 Modèle stationnaires

Dans le cas stationnaire le modèle (II.50) s'écrit :

$$X(i, j) = -\sum_{(m,n)\in S} a(m,n)\, x(i-m, j-n) + e(i,j) \qquad \text{(II.55)}$$

Les coefficients a (m, n) ne dépendent plus de la position du pixel (i, j) de l'image. Cette hypothèse n'est souvent pas réalisable pour un nombre important de points de l'image. Dans de telles situations on fait un découpage de l'image en zone arbitraires, sur chacune d'elles, on suppose que la stationnarité est satisfaite et par conséquent on estimera les paramètres du modèle AR pour chaque zone.

Le modèle (II.55) est intéressant lorsque les paramètres a (m, n) sont non nuls uniquement pour un nombre fini de couples (m, n) de D .Ce sous ensemble de D noté S est appelé fenêtre de prédiction ou bien support du modèle. La dimension de S définit l'ordre du modèle.

II.7.4 Exemples de supports

- ✓ Modèle causal quart de plan

$$S_0 = \{0 \leq m \leq M, 0 \leq n \leq N, (m, n) \neq (0, 0)\} \qquad \text{(II.56)}$$

Le couple (M, N) est appelé ordre du modèle.

- ✓ Modèle causal demi plan asymétrique

$$S_1 = \{m = 0, 1 \leq n \leq M\} \cup \{1 \leq m \leq M, -N \leq n \leq N\} \qquad \text{(II.57)}$$

- ✓ Modèle semi -causal

$$S_2 = \{0 \leq m \leq M, -N \leq n \leq N, (m, n) \neq (0, 0)\} \qquad \text{(II.58)}$$

- ✓ Modèle non causal

$$S_3 = \{-M \leq m \leq M, -N \leq n \leq N, (m, n) \neq (0, 0)\} \qquad \text{(II.59)}$$

II.8) Estimation des paramètres

Pour un support **S** fixé on cherche à estimer les paramètres de modèle (II.55) , pour cela l'entré e(i , j) est une erreur de prédiction donc le modèle est un modèle de prédiction ou bien de variance minimale .le champ de données bidimensionnels x(i , j) est supposé aléatoire, gaussien , stationnaire , de moyenne nulle et de variance finie.

Pour calculer les paramètres a (i, j) en minimisant la variance de l'erreur de prédiction e (i ,j) :

$$E[e(i\,,j)^2] = E[(\,x(i\,,j\,) - \sum_{(m,n\,)\epsilon S} a(m,n)x(\,i-m,j-n))^2] \qquad \text{(II.60)}$$

Ensuite on fait la dérivée partielle par rapport aux paramètres a(m ,n) , c'est-à-dire

$$\frac{\partial E[e(i,j)]^2}{\partial a(m,n)} = 0 \quad \forall\,(\,m,n\,) \in S \qquad \text{(II.61)}$$

L'équation de (II.61) implique la relation d'orthogonalité suivante :

$$E\,[e\,(i,\,j)\,x\,(i\,-k,\,j-l)] = 0,\ \forall\,(k,l)\epsilon\,S \qquad \text{(II.62)}$$

En remplaçant e (i , j) par sa valeur , on obtient le système d'équations normales habituelles :

$$E[(x(i,\,j)-\sum_{(m,n)\epsilon S} a(m,n)x(i-m,j-n))x(i-k,j-l)]= 0\ \forall(k,l)\epsilon S \qquad \text{(II.63)}$$

Ou bien en termes de fonction d'autocorrélation :

$$r(k\,,l\,) - \sum_{(m,n)\epsilon S}\ a(m,n)\,r(k-m,l-n) = 0\,,\forall\,(k,l)\in S \qquad \text{(II.64)}$$

Les paramètres a (m, n) sont obtenus en résolvant le système d'équations (II.64) et la variance minimale de l'erreur sera alors calculée à partir de la relation suivante :

$$\sigma_e = E[e(i,j)^2] = E[e(i\,,j)x(i\,,j)]$$
$$= r(0\,,0) - \sum_{(m,n)\epsilon S} a(m,n)r(m,n) \qquad \text{(II.65)}$$

Le système d'équation (II.64) et l'équation (II.65) peuvent être regroupés sous la forme condensée suivante :

$$r(k,l) - \sum_{m,n\epsilon S} a(m,n) r(k-m, l-n) = \delta_e^2 \ \delta(k,l) \quad \forall (k,l) \in S \qquad \text{(II.66)}$$

Suivant le choix de S, c'est-à-dire de la nature de la causalité du prédicateur, le système d'équation (II.66) peut être résolu en utilisant différents algorithmes. les propriétés du champ d'entrée e(i, j) sont aussi liées à la nature de la causalité et pour le montrer calculons la fonction d'autocorrélation de e(i ,j) pour les différentes géométrie de support **S** , dans le cas d'un modèle à variance minimale c'est-à-dire dans le cas de l'entrée e (i, j) est une erreur de prédiction .

La fonction d'autocorrélation de l'entrée e (i, j) est donnée comme suite :

$$r_e(k,l) = E[e(i,j)e(i-k, j-l)]$$

$$r_e(k,l) = E[\, e(i,j) \left\{ x(i-k, j-l) - \sum_{(m,n)\in S} a(m,n) x(i-k-m, j-l-n) \right\}]$$

$$\forall\, (k,l) \epsilon\, S \qquad \text{(II.67)}$$

En utilisant la relation d'orthogonalité (II.65) on obtient les résultats suivants :

- ✓ Modèle causal

$$r_e\,(k,l) = \sigma_e^2 \ , \ (k,l) = (0,0) \qquad \text{(II.68)}$$

$$r_e(k,l) = 0, \ ailleurs \qquad \text{(II.69)}$$

Qui est la fonction d'auto corrélation d'un bruit blanc.

- ✓ Modèle semi – causal

$$r_e(\,k\,,l) = \sigma_e^2 \ , \ (k\,,l) = (\,0\,,0) \qquad \text{(II.70)}$$

$$r_e\,(k,l) = -\sigma_e^2\, a(O,1) \ , \ (k,l) = (0,1) \epsilon\, S_2 \qquad \text{(II.71)}$$

$$r_e\,(k,\,l) = 0, \text{ ailleurs} \qquad \text{(II.72)}$$

Qui est la fonction d'auto corrélation d'un bruit blanc dans une direction et d'un bruit corrélé dans l'autre direction.

✓ Modèle non causal

$$r_e\ (k\ ,l\ ) = \sigma_e^2\ ,(k,l) = (0,0) \tag{II.73}$$

$$r_e(\ k,l\ ) = -\sigma_e^2 a(k,l), (k,l)\epsilon S_3 \tag{II.74}$$

$$r_e(k,l) = 0\ ,\ \ ailleurs \tag{II.75}$$

Qui est la fonction d'auto corrélation d'un bruit corrélé.

II.8.1) Ecriture matricielle des équations normales

Le système d'équation linéaire (II .66) peut être écrit sous forme matricielle suivante :

$$R_{MN}\ a = \sigma_e^2 L_{M,N} \tag{II.76}$$

R_{MN} : La matrice d'auto corrélation de l'image

a : Vecteur des paramétres

L_{MN} : Vecteur unitaire

Le vecteur de dimension (M× N) dont toutes les composantes sont nulles sauf une composante qui est égale a 1. Dans le cas stationnaire R_{MN} est une matrice symétrique de Toeplitz par bloc

Elle est définie par :

$$R_{M,N} = \begin{bmatrix} R_0 & R_{-1} & \dots & R_{-2N} \\ R_1 & \vdots & & \\ \vdots & \vdots & & R_{-1} \\ R_{2N} & \dots & R_1 & R_0 \end{bmatrix} \tag{II.77}$$

Les blocs R_K sont aussi constitués par des matrices de Toeplitz définies par :

$$R_K = \begin{bmatrix} r(0,k) & r(-1,k) & \dots & r(-2M,k) \\ r(1,k) & & & \\ \vdots & & & r(-1,k) \\ r(2M,k) & \dots & r(1,k) & r(0,k) \end{bmatrix} = R_{-k} \qquad \text{(II .78)}$$

Les dimensions et les valeurs des composantes des matrices R_{MN} , R_k et les vecteurs a , L_{MN} sont fonction de la nature de la causalité (S) du modèle.

II.8.2) Cas d'un modèle causal

❖ Causalité quart de plan

$Dans\ le\ cas\ ou\ S_0$ est le support de quart de plan l'équation (II.66) s'écrit comme suite :

$$r(k,l) - \sum_{m=0}^{M}\sum_{n=0}^{N} a(m,n)r(k-m,l-n) = \sigma_e^2\,\delta(k,l)\ ,\forall(k,l) \in S_0 \qquad \text{(II.79)}$$

$Ou\ \ (m,n) \neq (0\ ,0\ )$

La matrice d'auto corrélation sera donnée comme suite :

$$R_{MN} = \begin{bmatrix} R_0 & R_{-1} & \dots & R_{-N} \\ R_1 & & & \\ & & & R_{-1} \\ R_N & \dots & R_1 & R_0 \end{bmatrix} \qquad \text{(II.80)}$$

Les matrices blocs R_k qui est sont définies par (II.78) avec la dimension (M+1)*(M+1), parcontre le vecteur paramètre a est donné par :

$$a_0^T = [\ a_0^T\ a_1^T\ \dots a_n^T \dots a_N^T\ ] \qquad \text{(II.81)}$$

Ou

$$a_0^T = [1 - a(\ 1,0)\ \dots\ -a(m,0)\ \dots -a(M,0)\ ]\ ,\ \text{a (0, 0) =1}$$

$$a_n^T = [-a(\ 0,n)\ \dots\ -a(m,n)\ \dots\ a(M,n)\ ]$$

Le vecteur unitaire L_{MN} est donné par :

$$L_{MN}^T = \left[1_M^T\ 0^T\ \dots\ 0^T\ \dots\ 0^T\right] \qquad \text{(II.82)}$$

$$L_M^T = \left[1\ \ 0\ \ \dots\ \ 0\ \ \dots\ \ 0\ \right]$$

❖ **Causalité quart de plans avec les autres quadrants**

En supposant que le support S_0 causal de quart de plan est celui de premier quadrant et il est possible d'établir des relations similaires pour les autres quadrants qui sont affichés sur la figure (II .5).

On va définir les coefficients a (m, n) avec les supports du deuxième, du troisième et du quatrième quadrant on aura :

***2éme Quadrant* :**

$$a(m, n) = a^{(2)}(m,n), \quad \{(0 \le m \le M, -N \le n \le 0 \ , (m,n) \ne (0,0)\}$$

***3éme Quadrant* :**

$$a(m, n) = a^{(3)}(m, n), \{(-M \le m \le 0 \ , -N \le n \le 0 \ , (m, n) \ne (0, 0)\}$$

***4éme Quadrant* :**

$$a(m, n) = a^{(4)}(m, n), \{(-M \le m \le 0 \ , 0 \le n \le N \ , (m,n) \ne (0,0)\}$$

En suivant les étapes précédentes pour établir les équations normales relatives aux coefficients a(m, n) dans le premier quadrant , on déduit des équations normales des trois autres quadrants les relations suivantes :

$$a^{(1)}(m, n) = a^{(3)}(m, n) \qquad \text{(II.83)}$$

$$a^{(2)}(m, n) = a^{(4)}(m, n) \qquad \text{(II.84)}$$

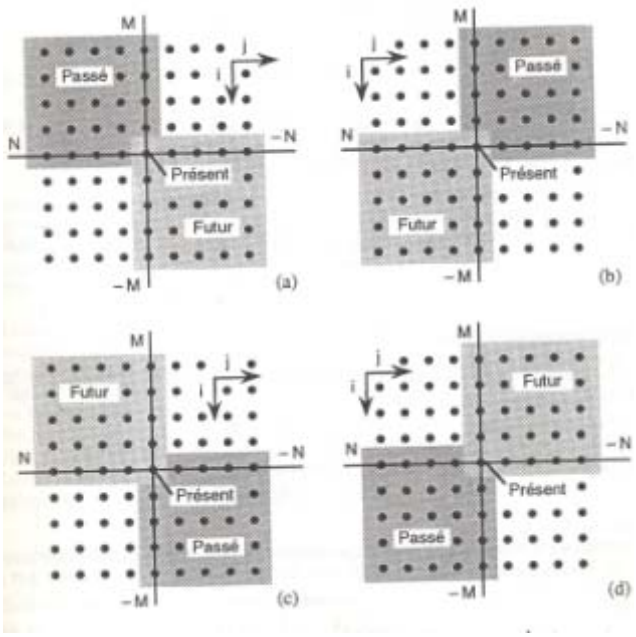


Figure (II.5) : causalité quart de plan dans chacun des 4 quadrants (a) 1er quadrant, (b) 2éme quadrant, (c) 3éme quadrant et (d) 4éme quadrant

❖ Causalité demi – plan asymétrique

Le support de la causalité demi asymétrique noté S_1 qui est défini par l'équation (II.57). L'équation (II.66) peut s'écrire :

$$r(k,\ l) - \sum_{m=1}^{M} a(m,0)\, r(k-m,l) - \sum_{m=-M}^{M} \sum_{n=1}^{N} a(m,n)\ r(k-m,l-n) = \sigma_e^2\, \delta(k,l),\ \ \forall (k,l) \in S_1 \qquad \text{(II.84)}$$

Dans ce cas la matrice d'auto corrélation R_{MN} sera donnée par :

$$R_{MN} = \begin{bmatrix} R'_0 & R'_{-1} & & R'_{-N} \\ R'_1 & R'_0 & R'_{-1} & R'_{-N+1} \\ & R'_1 & & R'_{-1} \\ R'_N & R'_{N-1} & R'_1 & R'_0 \end{bmatrix} \qquad \text{(II.85)}$$

Et les matrices bloc R'_K sont définies par l'équation (II.78) et R_k sont définies par la matrice suivante :

$$R'_k = \begin{bmatrix} r(M,k) & r(M-1,k) & r(-M,k) \\ r(2M,k) & r(2M-1,k) & r(0,k) \end{bmatrix} = R'_{-k} \quad \text{(II.86)}$$

$$a_0^T = [a_0^T \; a_1^T \ldots \quad a_n^T \quad \ldots \quad a_N^T] \quad \text{(II.87)}$$

Ou

$$a_0^T = [1 - a(1,0) \ldots - a(m,0) \ldots - a(M,0)], \qquad a(0,0) = 1$$

$$a_n^T = [-a(-M,n) \ldots - a(0,n) \ldots - a(M,n)]$$

$$l_{MN}^T = [1_M^T \; 0^T \ldots \quad 0^T \ldots \quad O^T] \quad \text{(II.87)}$$
$$L_M^T = [1 \; 0 \; \ldots \; 0 \; \ldots \; 0]$$

❖ **Cas d'un modèle semi -causale**

Le support S_2est défini par la relation (II.58), l'équation (II. 66) s'écrit :

r(k,l) $-\sum_{m=-M}^{M} a(m,0)r(k-m,l) - \sum_{m=-M}^{M}\sum_{n=1}^{N} a(m,n)r(k-m,l-n) = \delta_e^2\,\delta(k,l)$,

$\forall (k,l) \in S_2$ (II.88)

Avec a (0,0) = 1

Dans ce cas la matrice R_{MN} sera équivalente à la matrice définie par l'expression (II.77) dans laquelle les matrices blocs sont données par (II.78).les vecteurs a et L sont donnés par :

$$a^T = [a_0^T a_1^T \ldots \quad a_n^T \ldots \quad a_N^T] \quad \text{(II.89)}$$

$$a_0^T = [-a(-M,0) \ldots 1 \ldots - a(M,0)], \qquad a(0,0) = 1$$

$$a_n^T = [-a(-M,n) \ldots - a(0,n) \ldots - a(M,n)]$$

$$L_{MN}^T = [L_M^T \quad 0^T \quad \ldots \quad 0^T \quad 0^T \;] \quad \text{(II.90)}$$

$$L_M^T = [0 \; \ldots 0 \; 1 \; 0 \; \ldots 0 \;] \quad (N+1)^{\text{éme}} \text{ élément}$$

❖ **Cas non causal**

Le support S_3 est défini par l'équation (II.59) dans ce cas la matrice R_{MN}et les matrices blocs R_k seront équivalentes aux matrices définies par (II.77) et (II.78) et le vecteur a sera donné par :

$$a^T = [a_N^T \ldots \quad a_0^T \ldots \quad a_N^T] \quad \text{(II.91)}$$

$$a_0^T = [-a(-M,0) \ldots 1 \ldots - a(M,0)], \qquad a(0,0) = 1$$

$$a_n^T = [-a(-M,n) \dots - a(0,n) \dots - a(M,n)]$$

$$L_{MN}^T = \left[0^T \dots \quad L_M^T \quad \dots \quad 0^T\right] \qquad \text{(II.92)}$$

(M+1) éme élément

$$L_M^T = [0 \ \dots 0 \ 1 \ 0 \dots 0]$$

(N+1) éme élément

En définitive, pour un support S fixé, les paramètres du modèle et la variance de l'erreur de prédiction sont obtenus en résolvant le système d'équations (II.76) avec les valeurs de R, a et L correspondantes. dans la pratique, on montre qu'une image quelconque peut étre modélisée par un modèle d'ordre relativement faible : (M, N) ≤ (3, 3) et par conséquent la résolution de ces équations sera relativement simple.

A partir de l'équation (II.76) la matrice R est définie positive, il existe une solution unique de cette équation pour un ensemble de paramètres donnés a (m, n) , il n'existe pas une matrice unique pour laquelle ces paramètres sont solution de l'équation (II.76), la stabilité du modèle obtenu n'est pas garantie. Malgré cet inconvénient le système d'équations (II.76) est utile car il peut être résolu par des algorithmes rapides.

II.9) Estimation spectrale bidimensionnelle

Le modèle AR bidimensionnel (II.50) peut être décrit par une fonction de transfert rationnelle z_1 , et z_2 telle que :

$$\mathrm{H}\,(z_1, z_2) = \frac{B(z_1,z_2)}{A(z_1,z_2)} \qquad \text{(II.93)}$$

Ou

$$\mathrm{B}\,(z_1, z_2)=1$$

$$\mathrm{A}\,(z_1, z_2) = 1 - \sum_{(m,n)\in S} a(m,n) z_1^{-m} z_2^{-n} \qquad \text{(II.94)}$$

Avec $z_1 = e^{jw_1}$, $z_2 = e^{jw_2}$

e (i, j) l'entrée du modèle et x (i, j) la sortie du modèle , on définit la fonction densité spectrale (f.d .s.) en (Z_1, Z_2) de la séquence $\{e(i\,,j)\}$ par :

$$S_e(z_1, z_2) = \sum_{k=-\infty}^{+\infty} \sum_{l=-\infty}^{+\infty} r_e\,(k,l) z_1^{-k} z_2^{-1} \qquad \text{(II.95)}$$

$r_e(k,l)$: est la fonction d'autocorrélation de l'entrée, de la même façon on définit la fonction densité spectrale de la sortie :

$$S_x = (z_1, z_2) = \sum_{k=-\infty}^{+\infty} \sum_{l=-\infty}^{+\infty} r_x\,(k,l) z_1^{-k}\, z_2^{L} \qquad \text{(II.96)}$$

A partir de l'équation du modèle (II. 66) et des définitions de (II.96) et (II.95) on montre que $S_e(Z_1, Z_2)$ et $S_x(Z_1, Z_2)$sont liées par la relation suivante :

$$S_x(z_1,z_2) = \frac{S_x\,(z_1,z_2)}{A(z_1,z_2)\,A(z_1^{-1}z_2^{-1})} \qquad \text{(II.97)}$$

Dans le cas d'un modèle à variance minimale, c'est-à-dire dans le cas où e (i ,j) est une erreur de prédiction , A (Z_1,Z_2) est appelé filtre de l'erreur de prédiction .si le support S est causal , la densité spectrale de puissance de la sortie sera donnée par :

$$S_x(\omega_1,\omega_2) = \frac{\sigma_e^2}{\left|A(\omega_1,\omega_2)\right|^2} \qquad \text{(II.98)}$$

Dans le cas semi- causal le fonction densité spectrale en Z de la sortie sera donnée par :

$$S_x(z_1,z_2) = \frac{\sigma_e^2\,(1-\sum_{n=-N}^{N} a\,(0,n)\,Z_2^{-n}}{A(z_1,z_2)\,A(z_1^{-1},z_2^{-1})} \qquad \text{(II.99)}$$

D'une façon similaire on peut voir que pour un modèle non causal $S_x(Z_1,Z_2)$ est donnée par l'équation suivante :

$$S_x(z_1,z_2) = \frac{\sigma_e^2}{A(z_1,z_2)} \qquad \text{(II.100)}$$

Dans laquelle le terme A (Z_1^{-1}, Z_2^{-1}) s'élimine car il apparait à la fois au numérateur et au dénominateur. A partir des équations (II.98), (II.99) et (II.100) il est clair que le modèle à variance minimale semi-causal ou non causal, ne garantit pas la non négativité l'estimateur spectral, par contre le modèle causal donne toujours un estimateur non négatif.

II.10) Algorithme de Levinson bidimensionnel

II.10.1) Rappels et définitions

L'algorithme de levinson a été largement utilisé dans la résolution des équations normales qu'engendrent les problèmes du filtre 1D. La simplicité et l'efficacité numérique de cet algorithme, dues essentiellement à la forme Toeplitz de la matrice impliquée dans l'équation normale, ont motivé l'extension du cas 2D, de cet algorithme.

Notons que pour le cas 2D les équations normales introduisent des matrices de Toeplitz bloc Toeplitz . Avant de décrire les différentes étapes de cet algorithme, rappelons les définitions suivantes (Marezetta[1980]) :

- Un filtre linéaire bidimensionnel A (z_1, z_2) est dit à phase minimale s'il est causal et stable et est à l'inverse causal et stable.
- Une fonction de densité spectrale S (z_1, z_2) est dite analytique positive si S $(\omega_1, \omega_2) > 0$ te S (z_1, z_2) est analytique dans un voisinage du cercle unité, c'est-à-dire dans la région définie par : 1-ε $< |z_1|$, $|z_2| <$ 1+ε
- La séquence d'auto corrélation r (k, l) est analytique et définie positive si elle décroit exponentiellement en tendant vers zéro lorsque (k, l) $\longrightarrow \infty$ et la matrice de Toeplitz suivante est définie positive pour $|z_1| = 1$

$$R(z_1) = \begin{bmatrix} r_0(z_1) & r_1(z_1^{-1}) & r_M(z_1^{-1}) \\ & & r_1(z_1^{-1}) \\ r_M(z_1) & \dots & r_0(z_1 \end{bmatrix} \quad \text{(II.101)}$$

$$r_p(z_1) = \sum_{k=-\infty}^{+\infty} r(k,p) z_1^{-k} \quad \text{(II.102)}$$

II.10.2) Algorithme de levinson- Durbin

Supposons que la séquence d'auto corrélation r (k, l) est analytique et définie positive $(0,0) < (k, l) < (K, L)$ Alors:

- La solution du système d'équations normales peut être obtenue par les équations récursives suivantes :

1) Initialisation

k=1 l=1

$$a_1[1\ 1] = \frac{-r_x[1\ 1]}{r_x[1\ 0]}\ ,\quad \sigma_{1,0}^2 = (1 - a^2[1\ 1]\ ,\quad \sigma_{0,0}^2 = r(0,0)$$

$$z_1 = e^{j\omega_1}\ , z_2 = e^{j\omega_2}\quad \omega_1 = 2\pi f_1\ , \omega_2 = 2\pi f_2$$

2) ***Récurions***

pour k = 2,…….,p , L=2, ……,E

$$a[k\ ,l] = \frac{-1}{\sigma_{k,l-1}^2}[r(k\ ,l) + \sum_{m=0}^{k}\sum_{n=0}^{l-1} a(m\ ,n)r(k-m, l-n)$$

$$A_{k,l}(z_1, z_2) = A_{k,l-1}(z_1, z_2) - a(k\ ,l)z_1^{-k}z_2^{-1}A_{k-1}(z_1^{-l}z_2^{-l})$$

$$A_{0,0}(z_1\ ,z_2) = 1$$

$$\sigma_{k,l}^2 = \sigma_{k,l-1}^2(1 - a^2(k,l)$$

- le filtre $A_{k\ ,l}(z_1,z_2)$ est analytique et est à phase minimale. la structure de l'algorithme est analogue à celle de l'algorithme de levinson monodimensionnel, c'est-à-dire étant donnés un filtre $A_{k,l-1}(z_1, z_2)$ et la variance de l'erreur de prédiction $\sigma_{k,l-1}^2$ vérifiant l'équation normale pour une position donnée (k, l-1), il est possible de calculer $A_{k\ ,l}(z_1,z_2)$ et $\sigma_{k,l}^2$ par une simple récursion. Par analogie avec le cas 1D, les paramètres a (k, l) sont appelés coefficients de réflexion.

La méthode de résolution de l'équation de Yule -Walker par inversion matricielle ou par algorithme de levinson peut conduire à une solution d'un filtre non stable, la méthode d'itérative de Burg est l'algorithme efficace pour la résolution des équations de Yule – walker garanties un filtre stable comme solution.

L'idée de Burg est de calculer directement à partir des données une estimation des coefficients de réflexion $a_k[k,l]$ et ce, sans passer par le calcul des covariances. on déduit si nécessaire par la relation de levinson- Durbin et optimise les coefficients de réflexions par minimisation de la moyenne quadratique des erreurs de prédictions directes e_p^f (f = Forword) , erreur de prédiction rétrograde e_p^b (b = Backword)

Algorithme de Burg garantit que les estimes de coefficients de réflexions sont du module inférieur à 1 (stabilité).

Algorithme de Burg cas 2D

1) Initialisation

$r_x(0,0) = \sigma_e^2$

$e_0^f = x(i,j) \quad i = 0, \ldots\ldots\ldots N_1 - 1 \quad , \quad j = 0, \ldots\ldots\ldots\ldots N_2 - 1$

$e_0^b = x(i,j) \quad i = 0, \ldots\ldots\ldots N_1 - 1 \ , \quad j = 0, \ldots\ldots\ldots\ldots N_2 - 1$

$e_k^b(-1) = 0 \quad , \quad e_k^f(N+i) = 0$

2) Calcul des coefficients de réflexion

K= 1, 2,……..,p $\qquad a_{k+1}[k+1] = -\frac{\sum_{i=1}^{N_1+1} e_k^f(i)(e_k^b(i+1))^H}{\sum_{i=0}^{N_1} e_k^b(i)(e_k^b(i))^H}$

3) Calcule de la récursion de Levinson

K= 2,……,p , l = 2,……,E

$a(k,l) = \frac{1}{\sigma_{k,l-1}^2}[r(k,l) - \sum_{m=0}^{k}\sum_{n=0}^{l-1} a(m,n)r(k-m,l-n)$

$\sigma_{k,l}^2 = \sigma_{k,l-1}^2(1 - a^2(k,l)) \ , \qquad \sigma_{0,0}^2 = r(0,0)$

4) *calcul des érreurs*

$e_{k+1}^f(i) = e_k^f(i) + a_{k+1}[k+1]e_k^b(i-1)$

$e_{k+1}^b(i) = e_k^b(i-1) + J(a_{k+1}[k+1]^* J e_k^f(i)$

$i \in [\,0 \quad N+k\,]$, J : matrice diagonale identité

II.11) conclusion

Dans ce chapitre nous avons donné un aperçu sur quelques concepts mathématiques de bases concernant la modélisation d'image au niveau de gris. Nous avons décrit le modèle AR dans le cas mono et bidimensionnel. Les coefficients du modèle peuvent être estimés par plusieurs méthodes (Levinson- Durbin, Burg, covariance, moindre carrés). Dans le cas 2D ces coefficients sont considérés comme des attributs de texture et dépendent du support du modèle utilisé. Ces supports sont définis par les différents types de causalité et de symétrie qu'on peut déduire de la géométrie du voisinage d'un pixel.

:

III.1 Introduction

Nous allons présenter dans ce chapitre les résultats de la segmentation d'images texturées basée sur les modèles autorégressifs bidimensionnels.

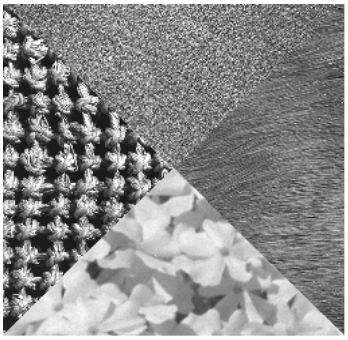
La programmation des différentes étapes de la méthode a été mise en œuvre en utilisant le langage Matlab 7.6. Le matériel utilisé est un PC portable équipé d'un système d'exploitation:

Windows 7 Edition familiale Premium. Les performances sont :

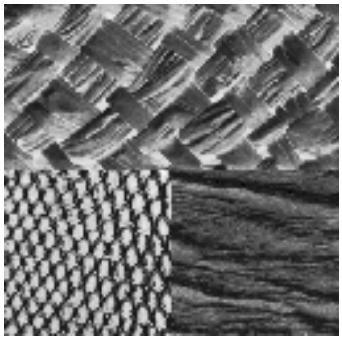
- Un processus : Intel ® Core (TM) i3 CPU M350 @ 2.27 GH_z
- Une mémoire vive de 4,00 G_O.

Afin d'évaluer les performances des techniques développées, les résultats sont illustrés sur deux sortes d'images :

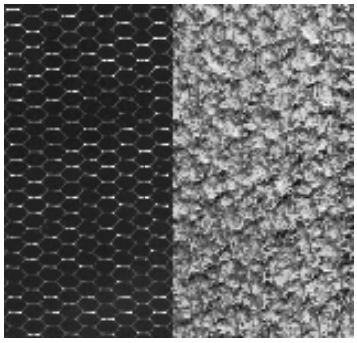
Trois images synthétiques test1, test2 et test3 tirées de l'album de Brodatz de taille 256*256 pixels composées respectivement de 4, 3 et 2 régions texturées codées sur 8 bits.



Test1



Test2



Test3

Deux images réelles test4 et test5 composées de différentes régions uniformes et texturées. L'image test4 est de taille 282*282 et les images test4 et test5 sont de taille 312*312 pixels.

Teste4

Test5

La démarche à suivre pour segmenter une image est la même quelque soit le type de causalité utilisé pour l'extraction des attributs de texture à base des modèles autorégressifs bidimensionnels. Elle est résumée par l'algorithme ci-dessous.

Etape 1 : Initialisation

1- Fixer la taille de la fenêtre de voisinage (w).
2- Fixer le nombre de classes de la texture(Nc).
3- Fixer le type de causalité et la taille du support M du modèle AR

Etape 2 : Estimation des attributs de texture

Calculer les attributs de la texture pour chaque pixel en utilisant la méthode des moindres carrées pour l'estimation des coefficients AR bidimensionnels avec des supports de causalité différents à savoir la causalité quart de plan (CQP), la semi causalité (SC) et la non causalité (NC). Quelque soit le type de causalité, le calcul des attributs de chaque pixel est effectué en tenant compte des pixels appartenant une fenêtre de voisinage de taille $(2W+1)^2$ centrée sur chaque pixel. On note par $t_{ij} = (t_{i1}, t_{i2}, ..., t_{ip})^t$ le vecteur caractéristique du pixel « i », t_{ij} constitue alors l'attribut « j » du pixel « i », avec $j = 1, ..., p$. Ces attributs correspondent aux p coefficients AR. La valeur de P dépend de type de causalité et la taille du support M, il est égal à :

Support NC : $P = (2M+1)^2 - 1$

Support SC : $P = \frac{(2M+1)^2}{2} - 1$

Support QP : $P = \frac{(2M+1)^2}{4} - 1$

Etape 3 : Initialisation des centres de classes

Initialiser au hasard ou manuellement les Nc centres de gravité $[\mu_1, \mu_2, \mu_3, \ldots, \mu_{Nc}]$ correspondant aux Nc classes.

Etape 4 : Affectation

1- affecter chaque pixel « i » à une classe C_k, dont le centre est μ_k.

Un pixel « i » appartient à la classe C_k de centre μ_k si et seulement si $\|t_i - \mu_k\|$ est minimale pour tout $k = 1, \ldots, Nc$.

2- mettre à jour la position du centre de gravité μ_k de la classe C_k.

$$\mu_{kj} = \frac{1}{N_k} \sum_{i \in C_k} t_{ij}$$

où N_k est le nombre de pixels de la classe C_k.

Les résultats de cette méthode dépendent principalement de la taille de la fenêtre de voisinage (W), de l'ordre P du modèle AR donc des attributs de texture correspondant aux coefficients AR, ainsi que le support de causalité. Pour les images synthétiques le nombre de classes Nc est fixé aux nombre de textures correspondantes tandis que pour les images réelles le Nc est fixé à 4 pour l'image test1, 3 pour les images test2 et test4 et 2 pour les images test3 et test 5. Nous allons analyser dans la section suivante les différents résultats de la segmentation en fonction des paramètres W, M et du type de causalité.

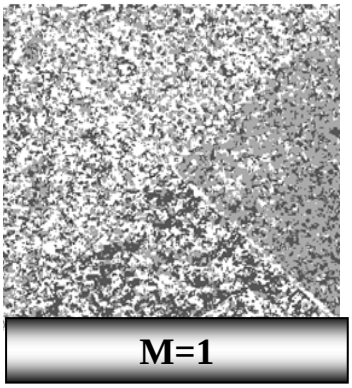
III.2 Présentation des résultats

Afin d'analyser les performances de l'approche proposée, nous avons utilisé l'image test1 pour conduire tous nos tests.

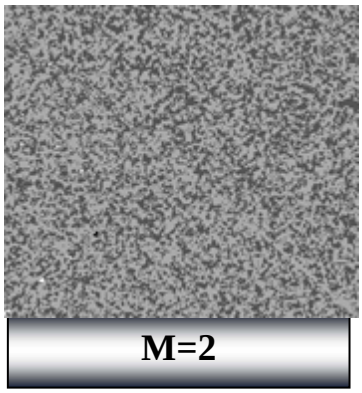
Les figures (III.1) à (III.3) montrent les résultats de la segmentation de l'image synthétique (Test1) à base en fonction du type de causalité et en fonction de la taille de la fenêtre(W) et de l'ordre du modèle ou la taille du support (M).

Les figures (III.4) à (III.6) montrent quelques résultats de la segmentation obtenus sur les autres images tests.

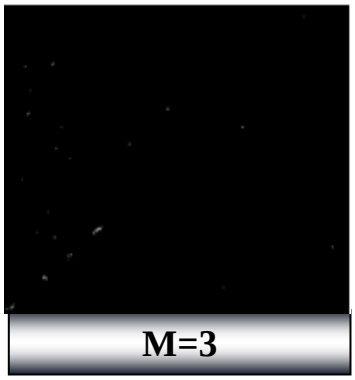
W=1



M=1

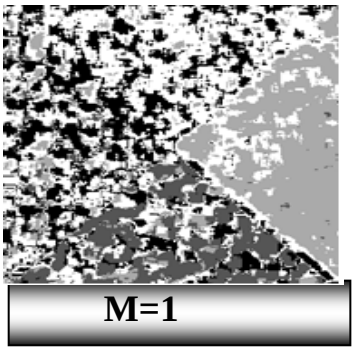


M=2

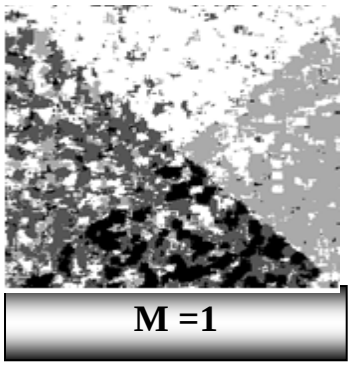


M=3

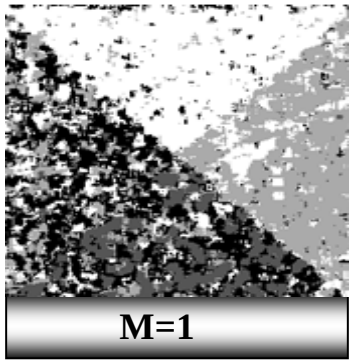
W=3



M=1

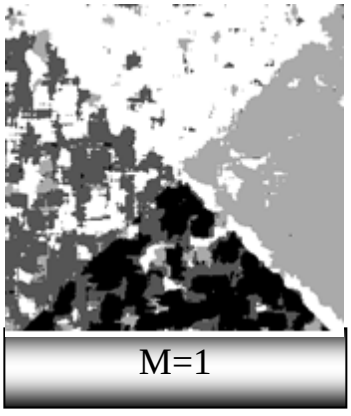


M =1

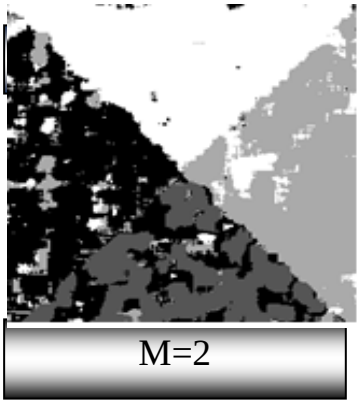


M=1

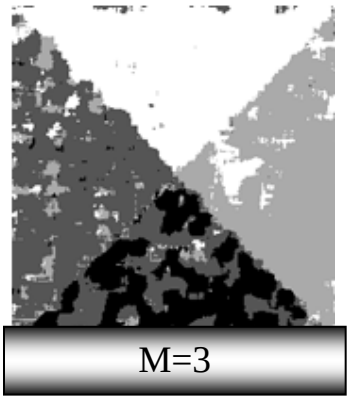
W=5



M=1

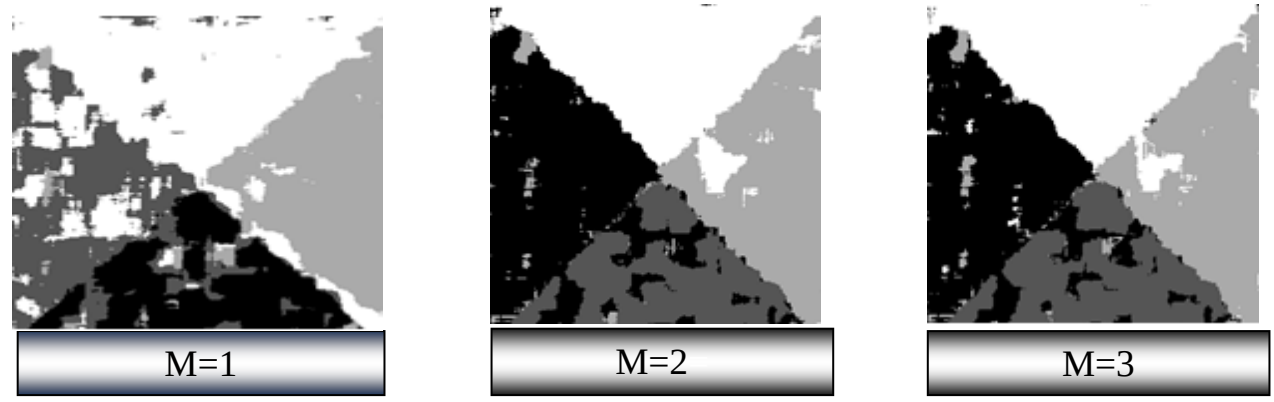


M=2

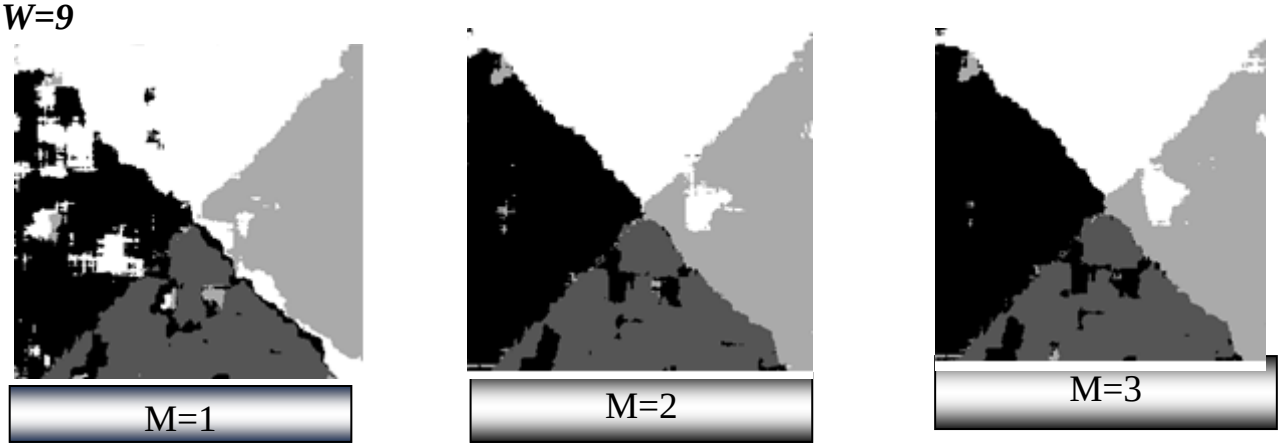


M=3

W=7



W=9



W=11

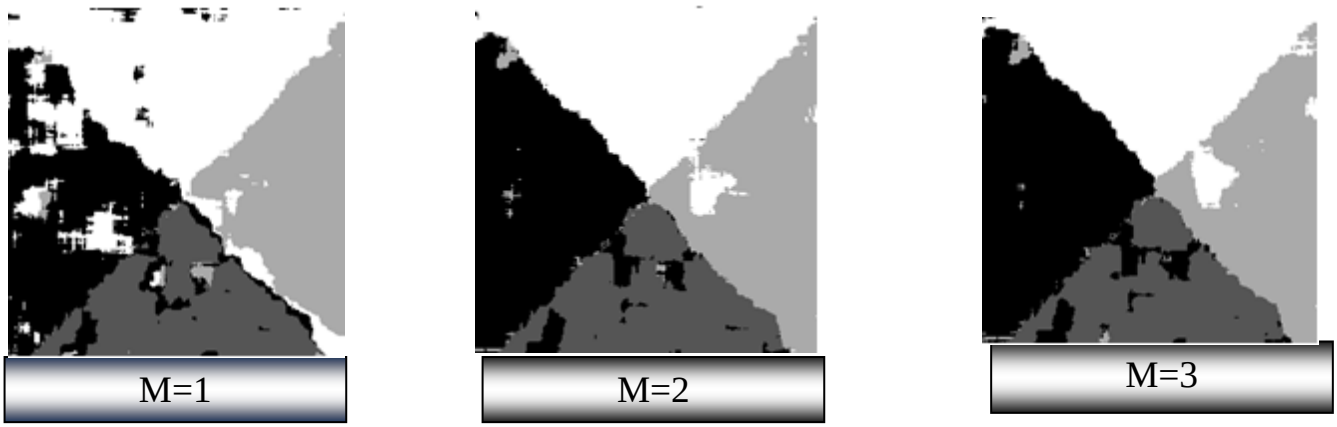
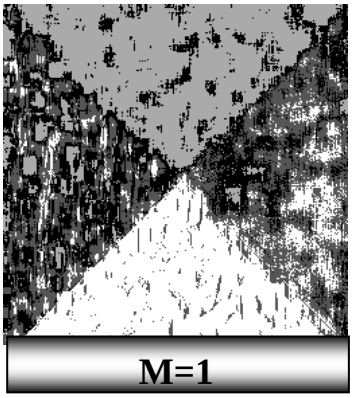
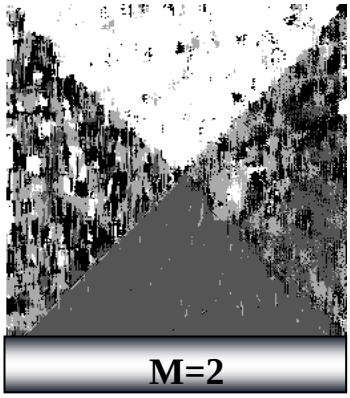
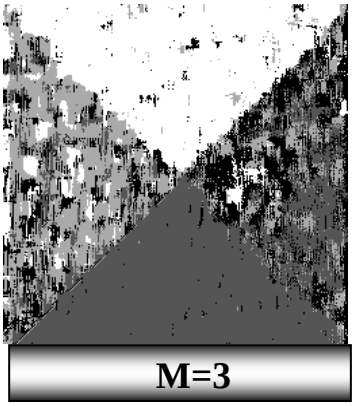


Fig.III.1 : Segmention de l'image Test1 selon la causalité quart de plan

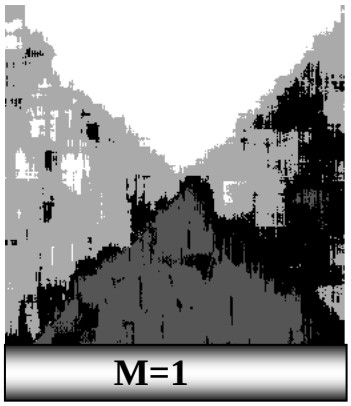
W=1

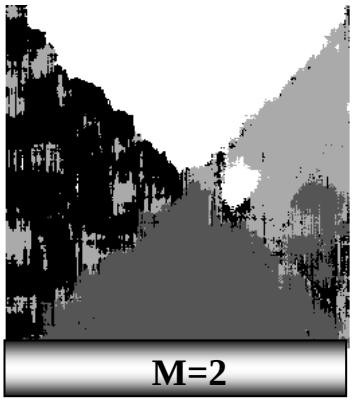


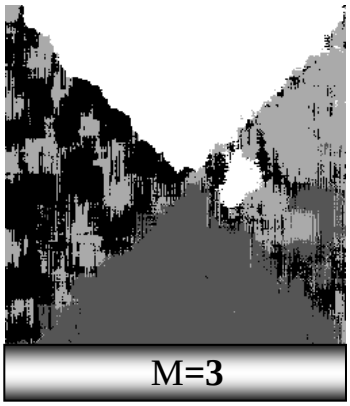




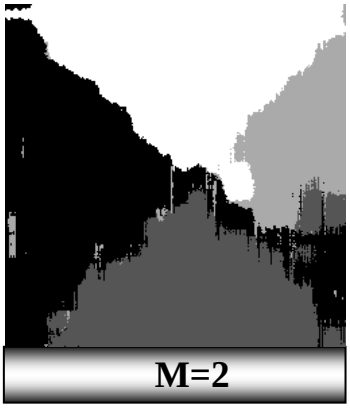
W=3







W=5



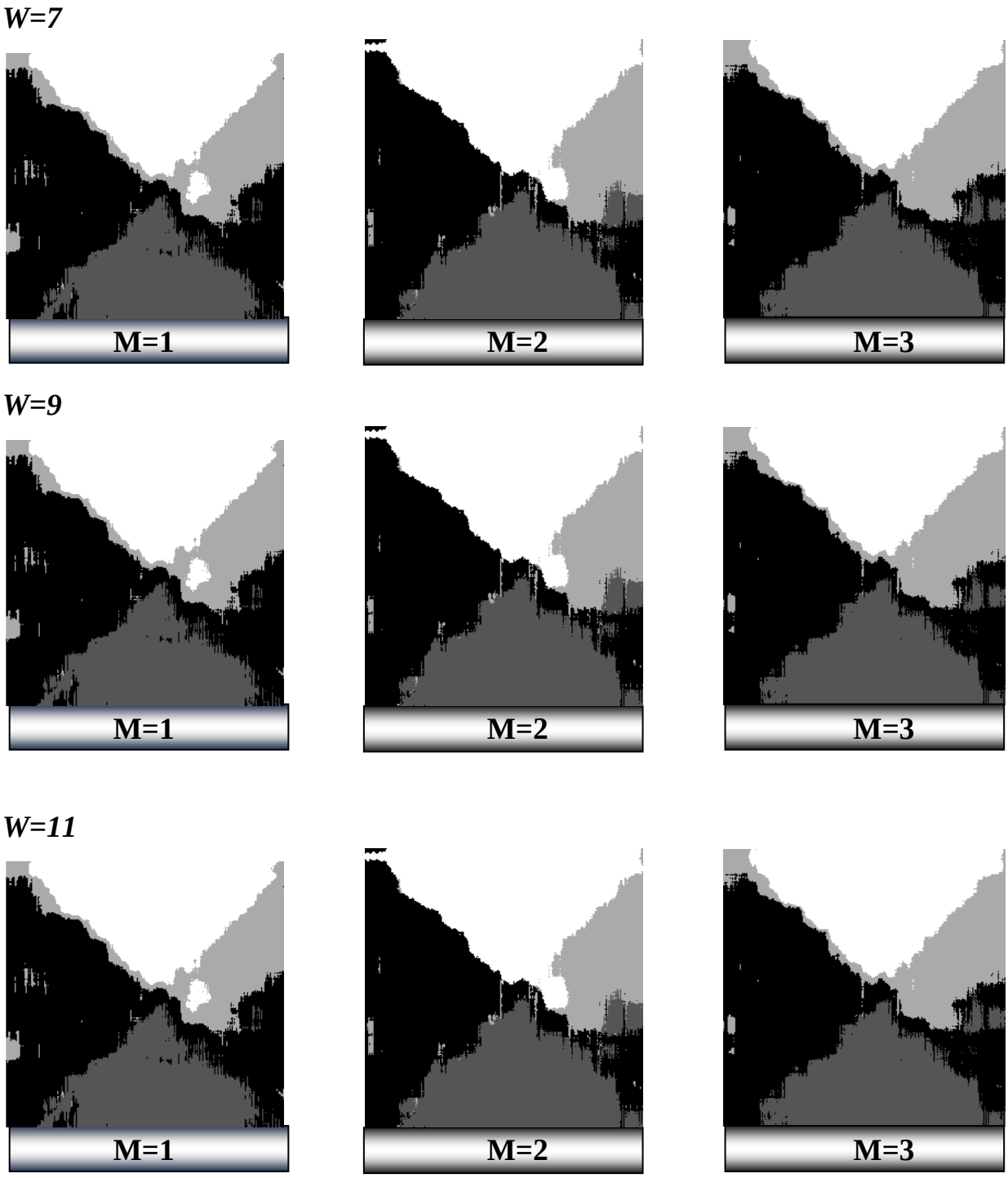
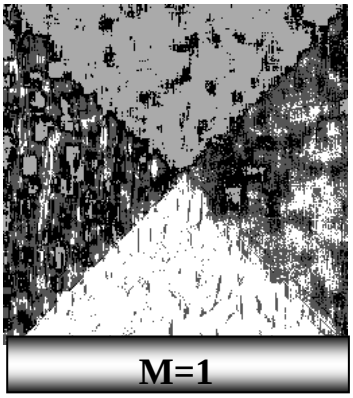
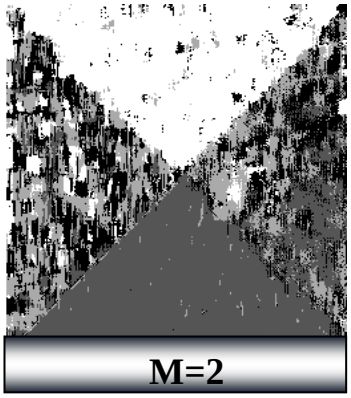
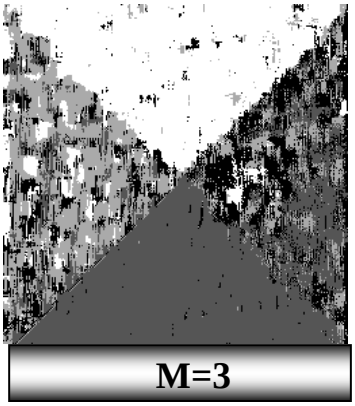


Fig.III.2 : Segmention de l'image Test1 selon la causalité semi causale

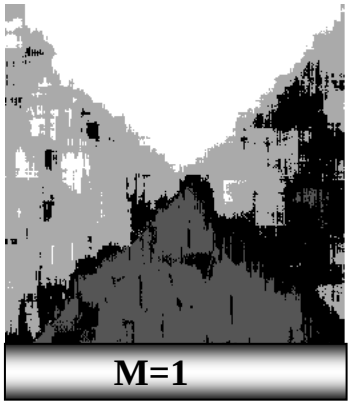
W=1

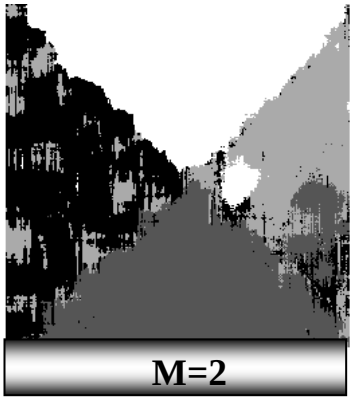


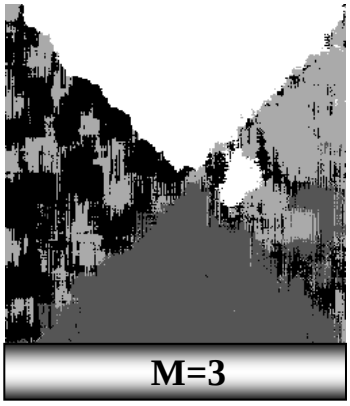




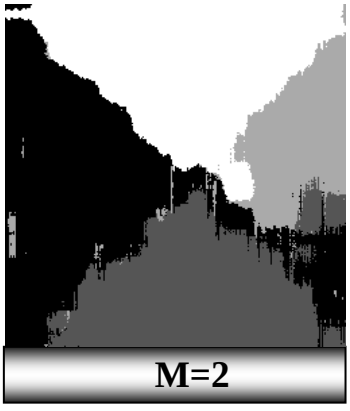
W=3



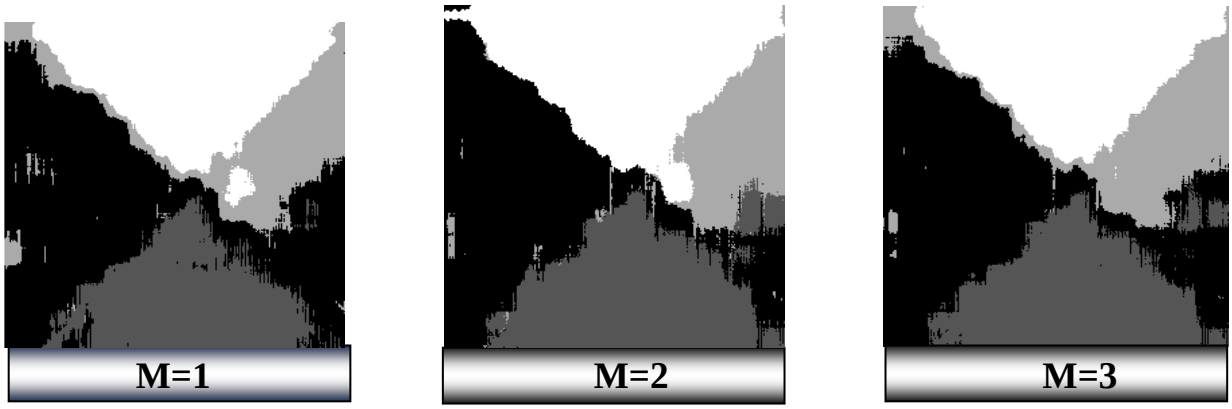




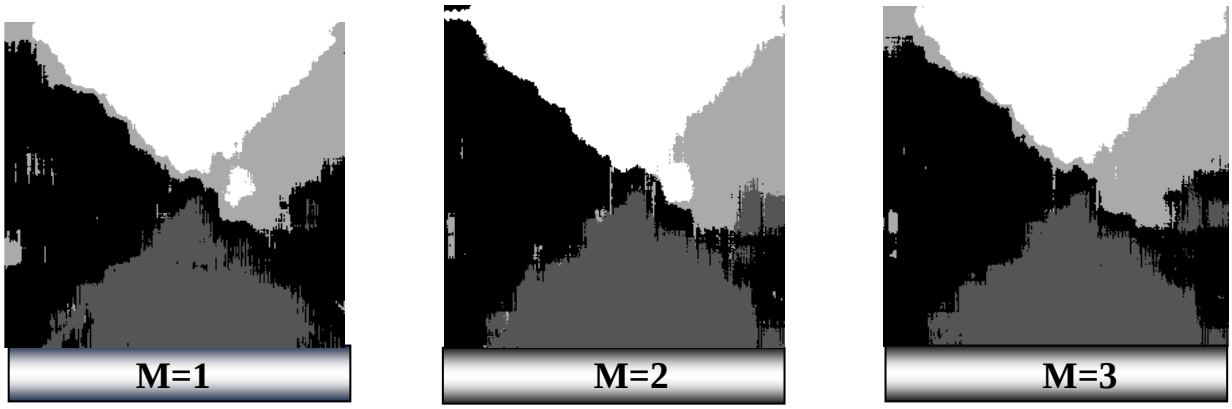
W=5



W=7



W=9



W=11

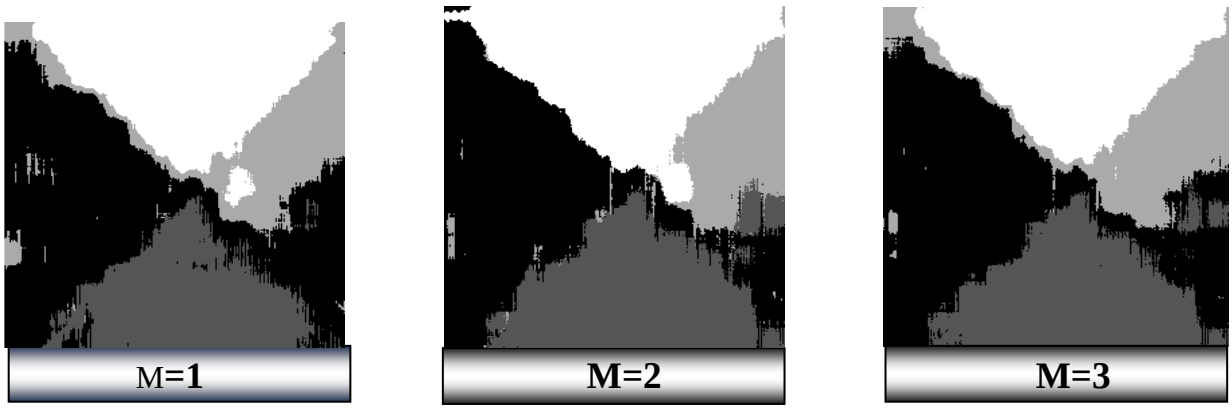


Fig.III.3 : Segmention de l'image Test1 selon la causalité Non causale

Interprétation

D'une manière générale, nous constatons visuellement que la méthode d'estimation par les moindre carrées des coefficients autorégressifs bidimensionels permet de mettre en évidence les zones de même texture lorsque W est grand quelque soit le type de causalité, quelque soit la taille du support M. Cependant ces zones sont mieux délimitées lorsque W est petit.

En comparant les différents types de causalité, nous remarquons que la causalité quart de plan (QP) donne de meilleurs résultats que les autres types de support car les frontières entre les zones de même texture sont bien délimitées et que toutes les 4 textures sont bien détéctées alors que les autres support ont permis de détecter convenablement que 3 zones de textures. Cependant, il très difficile de comparer visuellement les résultats fournis par les supports SC et NC.

On peut également voir que plus le nombre de coefficients AR (P) augmente plus les zones de même texture apparaissent homogènes. Cependant, les meilleurs résultats sont obtenus lorsque M=2.

En ce qui concerne le temps d'éxécution, celui-ci varie selon la taille de la fenêtre et le type et la taille du support. Plus W et M augmentent plus le temps de calcul nécessaire pour l'estimation des attributs de texture est elevé car le nombre de pixels et la taille des systèmes à résoudre devient plus elevés.

Il y a lieu de relever que dans les différents tests que nous avons effectué sur d'autres images, les résultats obtenus ont montré que la performence d'un type de causalité par rapport aux autres dépend de la nature de la texture qui compose l'image selon qu'elle est fine ou grossière, douce ou granulée, ect. Ainsi par exemple, pour l'image test 2, les supports SC et NC donnent de meilleurs résultats par rapport au support QP, alors que pour l'image test 3 tous les supports donnent de bons résultats. La taille du support influe également sur la qualité des résultats. Par exemple, pour l'image test4, la meilleur segmentation est obtenue lorsque M=3 quelque soit le type du support. D'une manière générale, il faut bien choisir la taille du sopport ainsi que la taille de la fenêtre de voisinage. En effet pour des regions de texture fine, une petite fenêtre de voisinage suffit pour donner des resultats plus performants contrairement à des regions de texture grossière.

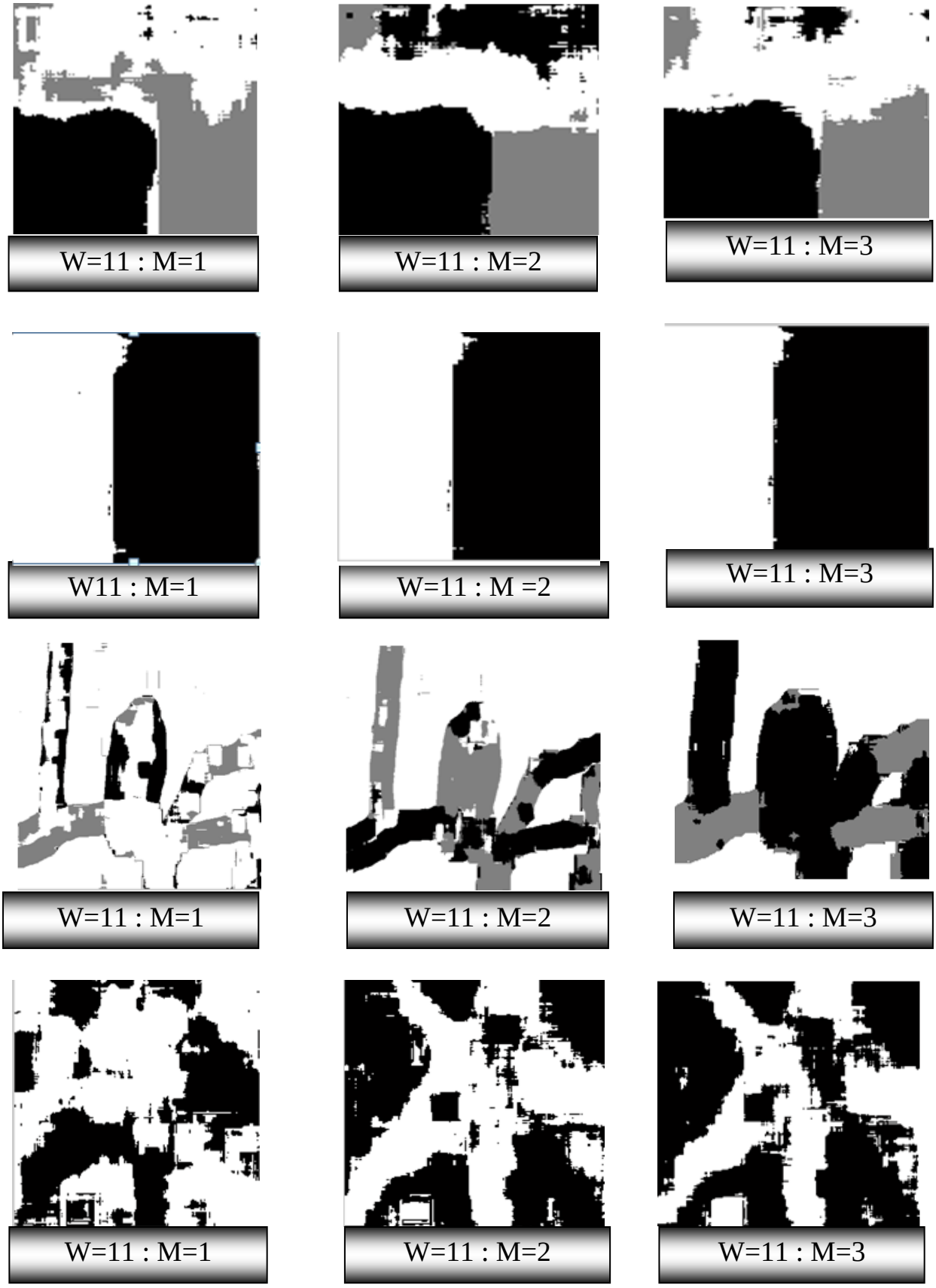


Fig.III.4 : Résultats obtenus de la segmention des autres images tests selon la causalité quart de plan

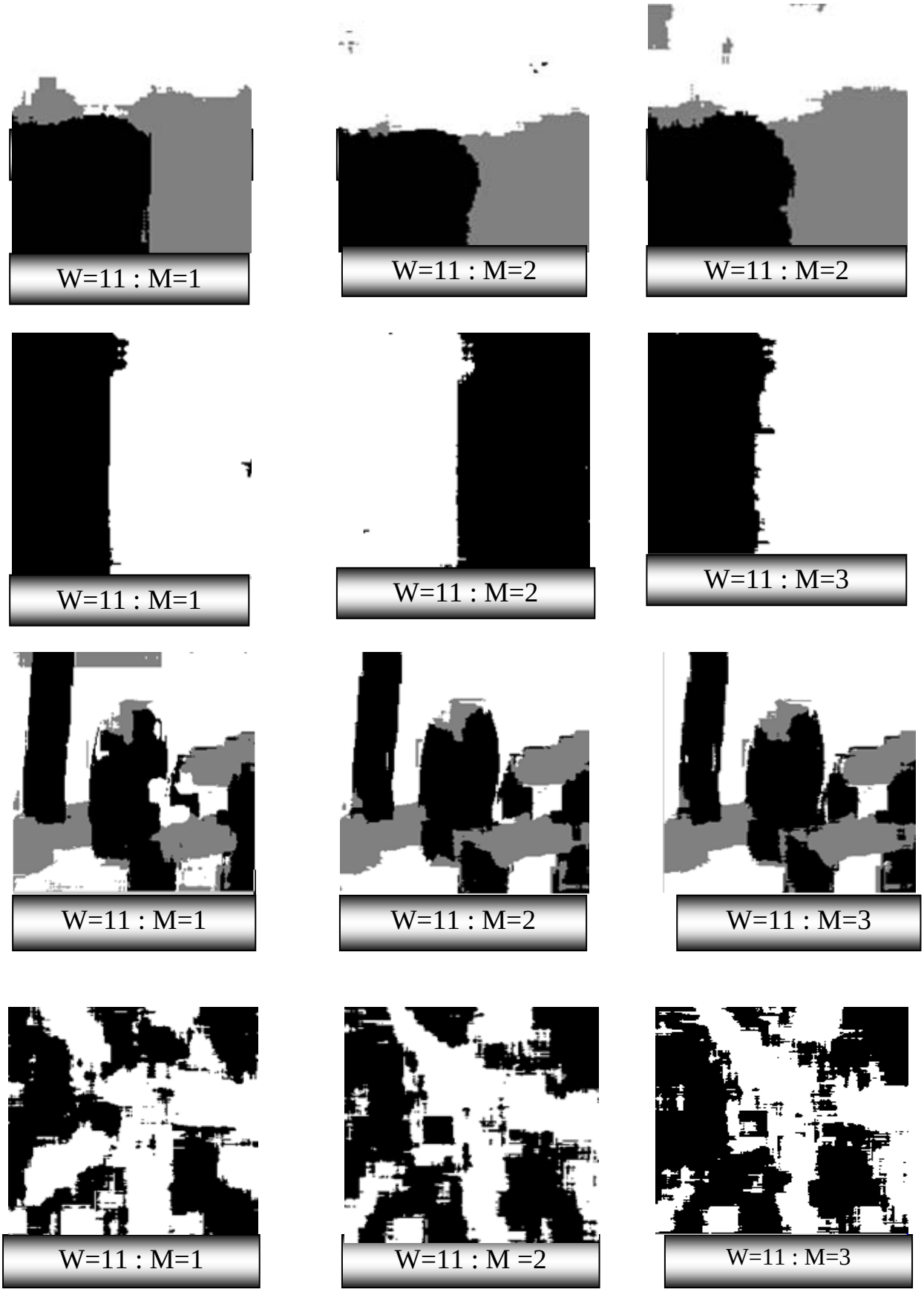


Fig.III.5 : Résultats obtenus de la segmention des autres images tests selon la semi causalité

W=11 : M =1

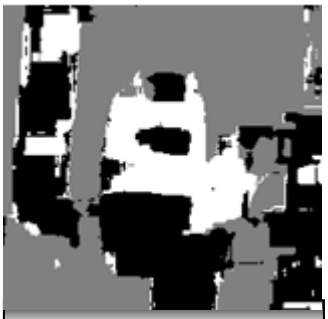
W=11 : M= 2

W=11 : W=11 : M=3

W=11 : M=1

W =11 : M=2

W=11 : M=3



W=11 : M=1

W=11 : M=2

W=11 : M=3

W=11 : M=1

W=11 : M=2

W=11 : M=3

Fig.III.6 : Résultats obtenus de la segmention des autres images tests selon la non causalité

III.3 Conclusion

Nous avons présenté dans ce chapitre, les résultats de segmentation obtenus par les attributs autoresgressifs bidimensionnels estimés par la méthode des moindres carrées. La qualité des résultats dépendent du type de causalité(QP, SC et NC) , de la taille de la fenêtre de voisinage W et de la taille du support M. Le choix de ces paramètres doit être fait selon les textures qui composent l'image. Or cette dernière information est difficile à acquirir.

Conclusion générale

Nous avons présenté dans ce mémoire une méthode de segmentation d'image texturée basé sur la modélisation autorégressive.

Notre méthode consiste à caractériser dans un premier temps chaque pixel par des attributs de texture correspondant aux coefficients autorégressifs en utilisant trois types de causalité, la causalité quart de plan (QP), la semi causalité (SC) et la non causalité (NC). Dans un deuxième temps, les pixels ayant des attributs de textures proches sont regroupés en classes de textures. Les pixels adjacents appartenant à une même classe de texture forment alors dans l'image segmentée une zone ou texture homogène. La classification des pixels est effectuée la méthode classification des K-MEANS.

Une bonne description de l'information texturale nécessite un choix judicieux de plusieurs facteurs. Nous avons montré l'influence de la taille de la fenêtre d'analyse ainsi que le type et la taille du support de causalité sur la qualité de la segmentation.

L'application des modèles AR sur les images texturées ne se limite pas seulement aux trois types de causalité utilisé dans notre travail cependant, elle peut être étendue aux autres types de causalité et à l'estimation des paramètres AR 2D par d'autres méthodes.

Nous souhaitons ainsi que notre travail servira de base pour étudier cette extension et la réalisation d'autres applications dans le domaine de traitement d'image à base des modèles autorégressifs.

Bibliographie

[1] Christophe ROSENBERGER, Mise en Oeuvre d'un système adaptatif de segmentation d'image . L'université de RENNES 1.

[2] M. Unser . Description statique de la texture. PhD thesis , EPLF, Lansanne, 1984.

[3] M. Unser, Texture classification and segmentation using walvet frames. IEEE Transaction on image prossing, 4(11):1549-15-60, November 1995.

[4] T. Chang and C.C Jay Kuo. Texture analysis and classification with treestructured walvet transform.IEEE transaction on image processing ,2(4):429-441, October 1993.

[5] J.F.Haddon and J.F .Boyce. Texture classification of segmented region of flir image using neural networks. In IEEE International Conference on image processing,volume3, page 660-664, AUSTIN, November 1994.

[6] A. Laine and J. Fan . texture classification with walvet packet signatures. IEEE Transactions on pattern Analysis and machine Intelegence 11(15):1186-1191, January 1993.

[7] R.W. Picard. Asociety of models for video and image libraries. IBM system journal,3,1996.

[8] T.Zhang. Issues in texture classification . CS328B, May 1996.

[9] B. Julesz. Experiments in the visual perception of texture. Scientific software, 232:2-11, 1975.

[10] R.M. Haralick. Statistical and structural approaches to texture. In proceedings of the IEEE, volume 69, pages 786-804, may 1979.

[11] J.M. Franco, A.Z.Meiri, and B.Porat. A unified texture model based on a 2-d wold-like decomposition . IEEE Transaction on Singal processing, 41:2665-2678, August 1993.

[12] A. Gagalowicz. Vers un modèle de texture. PhD thesis, Université Pierre et marie CURIE, PARIS VI, May 1983.

[13] thèse Madjid MOGHRANI . segmentation coopérative et adaptative d'images multicomposantes :Application aux image CASI.

[14] M. Idrissa, M. acheray . texture classification using GABOR filters. Patten Recognition, volume 23, pages 1095-1102, 2002.

[15] I.M. Elfodel and R.W. Picard Gibbs randon fields, cooccurrences, and texture modeling. IEEE transaction an pattern analysis and machine intelligence, volume16, 1994.

[16] B. Naegel. Autils pour le traitement et l'analyse d'image, introduction a la morphologie mathématique. Ecole des MINEES a NANCY, 2008.

[17] S. Mavromatis. Analyse de la texture et visualisation Scientifique. Thèse de doctorat en informatique Université de la Méditerranée, 2001.

[18] MR. Turner. Texture discrimination by GABOR fonction biological cybermetics, volume 55, pages 71-82, 1986.

[19] A.K Jain and F. Farrokhnia. An supervised texture segmentation using GABOR filters. pattern. Recognition , volume 24, pages 1167-1186, 1991.

[20] Ouadfel Salim, doctorat, université de Batna contribution à la segmentation d'images basées sur la résolution collective par colonies de fourmis artificielles, 2006.

[21] Thèse K.Hammouche : Méthodes statistique d'analyse de la texture appliquées à la segmentation.

[22] Alain Boucher. IFI. Cours texture : vision par ordinateur.

[23] UNSER, M.and de COULON. F. Analyse des textures et segmentation.

[24] Jeans-Jaques ROUSSELLE. Les contours actifs, une méthode de segmentation. Application à l'imagerie médicale. Juillet 2003 .Université de TOURS.

[25] Catherine ACHARD. Cours traitement d'image 2006.

[26] Henri MAITRE. La détection des contours dans les images.

[27] Sébastien LEFEVRE. Une nouvelle approche pour la classification non supervisée en segmentation d'image (coures). Université de Strasbourg.

[28] Classification par K- means.

Htt://www.aiacess. net:/ y tut liste descriptive .htm #k means

[29] Olivier Besson. Analyse Spectrale paramétrique 3éme année ENSICA.

[30] I.D.LANDAU. Techniques de modélisations récursives pour l'analyse spectrale paramétrique adaptative. Laboratoire d'Automatique de Grenoble (UA 228, CNRS- SAINT-MARTIN-D'HERES.

[30] Mohamed NAJIM Modélisation et identification en traitement du signal Ingénieur ENSERB, Professeur à la faculté des Sciences de Rabat. MASSON, paris Milan Barcelone Mexico 1988.

[31] H.Youlal , M.J.Idrissi, M.Najim. Modélisation paramétrique en traitement d'images.

[32] K. HAMMOUCHE application du modèle atoregressifs à la segmentation d’images texturées (2010).

Le travail présenté dans ce mémoire concerne la méthode de segmentation d'image texturée basé sur la modélisation autorégressive.

Notre méthode consiste à caractériser chaque pixel par des attributs de texture correspondant aux coefficients autorégressifs en utilisant trois types de causalité, la causalité quart de plan (QP), la semi- causalité (SQ) et la non causalité (NC).

Tous les pixels ayant des attributs de textures proches sont regroupés en classe de textures ,la classification des pixels est effectuée par la méthode de K-MEANS.

Une bonne description de l'information texturale est porté sur l'influence de la taille de fenêtre d'analyse ainsi que le type et la taille du support de causalité sur la qualité de la segmentation.

L'application des modèles AR sur les images texturées ne se limite pas seulement aux trois types de causalité utilisé dans notre travail , elle peut être étendue aux autres types de causalité et à l'estimation des paramètres AR 2D par d'autre méthodes .