



N^od'ordre :

**UNIVERSITE*MOHAMED BOUDIAF* DE M'SILA
FACULTE DES SCIENCES ET SCIENCES DE L'INGENIEUR
DEPARTEMENT D'INFORMATIQUE**

MEMOIRE

Présenté pour l'obtention du diplôme de :

Magister

**Spécialité : Informatique
Option : Informatique industrielle**

Par :

LAKEHAL Meftah

Thème :

**INDEXATION DES IMAGES DANS UN
ENVIRONNEMENT DISTRIBUE (JINI)**

Soutenu publiquement le : / / devant le jury composé de :

KHELLADI Abdelkader	Pr. USTHB	Président
BOUDERAH Brahim	M.C. Université de M'sila	Rapporteur
MAAMRA Mohamed Said	C.R. USTHB	Co-Encadreur
ABBAS Moncef	Pr. USTHB	Examineur
MIHOUBI Douadi	M.C. Université de M'sila	Examineur
ASSAS Ouarda	C.C. Université de M'sila	Invitée

Résumé:

L'indexation et la recherche des images par le contenu (CBIR), constituent un domaine de recherche très varié étant donné le développement de l'image numérique.

Du point de vue de l'indexation, nous proposons d'appliquer la transformé en ondelettes communément avec les moments et invariants comme des outils puissants pour le calcul de signatures en se basant sur l'information texture relative des images.

Du point de vue de la recherche, nous utilisons le principe de la recherche par similarité selon lequel les images sont ordonnées à l'aide de la mesure de similarité à la requête. Cette dernière prend la forme d'une image exemple. Dans ce même cadre, nous utilisons la distance géométrique entre les signatures. aussi il y a un survol sur les architectures distribuées, où nous espérons un CBIR distribué.

Mots clés :

Indexation, recherche, ondelettes, similarité, moments et invariants, architectures distribuées.

Remerciement

Je remercie en premier lieu notre Dieu qui m'a facilité la tâche, et m'a donné force et volonté pour bien achever ce travail.

Puis je tiens à remercier les membres du jury :

Abdelkhadar KHALLADI, professeur à l'Université de sciences et technologies Houari Boumediene d'Alger (USTHB), directeur générale du CIRIST, pour l'honneur qu'il m'accorde en acceptant de présider le jury de ce mémoire.

Brahim BOUDERAH, Maître de conférence à l'Université M^{ed} BOUDIAF de M'sila pour avoir accepté d'être rapporteur de ce mémoire. Je lui exprime ma gratitude pour m'avoir encouragé, soutenu et pour avoir été disponible tout au long de ce travail.

Mohamed Said MAAMRA, Chargé de recherche à l'USTHB, co-encadreur de ce mémoire. Ses critiques constructives et ses encouragements répétés m'ont été très précieux. Je le remercie encore de m'avoir proposé ce sujet.

Moncef ABBAS, professeur à l'USTHB, Douadi MIHOUBI, maître de conférence à l'université M^{ed} BOUDIAF de M'sila, et Ouarda ASSAS, Chargée de cours à l'université M^{ed} BOUDIAF de M'sila, d'avoir accepté d'évaluer ce travail.

Je tiens aussi à remercier tous ceux qui m'ont entouré dans ce travail :

Ma famille, mes amis et mes collègues, et surtout Abdelkarim GHALLAB pour m'avoir aidé et encouragé au cours de ce travail, dans les bons et moins bons moments.

L'ensemble des enseignants, étudiants et personnel du département d'informatique et de l'université M^{ed} BOUDIAF de M'sila, ainsi, Rossafi MAHDJOUBI chef du département, Abdelkader GHASMI chef du conseil scientifique, Fares HEIMER, Hamza BACHIRI et Khaled qui, par leurs conseils et leurs encouragements, ont contribué directement ou indirectement à l'accomplissement de ce mémoire.

*A mes parents,
A mes frères,
A mes amis.*

Table des matières

Introduction:1

Chapitre I : La recherche d'informations visuelles3

1. Introduction	3
2. Méthodologie des systèmes d'indexation et de recherche.....	4
2.1. La recherche des images	5
2.2. Indexation des images	6
3. Descripteurs du contenu de l'image.....	7
3.1 Indexation et recherche par la couleur.....	8
3.1.1. Les espaces de couleur.....	8
3.1.2. Histogramme.....	9
3.1.3. Information spatiale	10
3.1.4. Les moments de couleur	10
3.1.5. Corrélogramme de couleurs.....	11
3.1.6. Vecteur de cohérence couleur	12
3.2 Indexation et recherche par la forme	12
3.2.1. Segmentation et détection de points d'intérêts.....	13
3.3. L'indexation et la recherche par la texture	13
3.3.1 Caractéristiques de <i>Tamura</i>	14
3.3.2. Caractéristiques de Wold	16
3.3.3. Modèle Auto-Regressif Simultané (SAR)	16
3.3.4. La transformée en ondelettes	17
4. Conclusion	18

Chapitre II : Les ondelettes19

1. Introduction	19
2. De l'analyse de Fourier à l'analyse en ondelette	20
2.1 Analyse de Fourier	20
2.2. Transformée de Fourier à fenêtre glissante	20
2.3 Les ondelette	23
2.3.1. Transformée en ondelettes discrète.....	27

3. banc de filtres	28
3.1 Sous-échantillonnage	29
3.2 Sur-échantillonnage.....	30
4. Analyse multirésolution.....	32
4.1 Espace d'approximation.....	33
4.2 Espace de détails	34
5. Conclusion	37
Chapitre III : Extraction de l'index	38
1. Introduction	38
2. Les ondelettes multidimensionnelles	39
3. Les ondelettes séparables versus les ondelettes non séparables :.....	39
4. Echantillonnage en quinconce	40
5. La transformée en quinconce.....	42
5.1. Implémentation dans le domaine de Fourier	44
6. La transformée en Lifting	49
6.1. La décomposition	49
6.2. La reconstruction.....	51
7. Application de la transformée en lifting pour l'analyse d'images	52
7.1. Filtre de Neville	53
8. Moments et invariants	55
9. La mesure de similarité.....	57
9.1. Distance géométrique.....	57
9.2. Distance de Mahalanobis	58
10. Expérimentation et résultats.....	59
11. Conclusion	66
Chapitre IV : CBIR Distribué	66
1. Introduction	66
2. CBIR distribué	67
3. L'architecture Jini.....	68
3.1. Fonctionnement du Jini	70
3.1.1. Service lookup	70
3.1.2. Protocole de découverte.....	70
3.1.3. Processus d'adhésion	73
3.1.4. L'indépendance de protocole	73
3.1.5. Les groupes	73
4. L'architecture RMI	74

4.1. Les objets sérialisés.....	75
4.2. Le code mobile.....	75
4.3. Les interfaces et serveurs RMI.....	76
5. L'architecture CORBA.....	77
5.1. ORB (Object Request Broker)	78
5.2. Les Protocoles CORBA	78
5.3. Langage IDL	78
6. Comparaison entre JINI, CORBA, RMI.....	79
7. Conclusion.....	81
Conclusion	82
Bibliographie :.....	84

Notations :

Notation générale	Interprétation
(i, j)	Pixel
$a = 2^j, j \in \mathbb{Z}$	a : échelle dyadique, j : un niveau, $\frac{1}{a} = 2^{-j}$: la résolution
ψ	Ondelette
\hat{f}, X	Transformée de Fourier de la fonction f ou x
x, s	Signal ou image
$g = f * h$	Convolution de f et h .
$L^2(\mathbb{R}), L^2(\mathbb{Z})$	L'ensemble des fonctions d'énergie finie.
$\langle s s' \rangle = \int_{\mathbb{R}} s(x) \overline{s'(x)} dx$	Produit scalaire des fonctions s et s' .
$\delta(n)$	Impulsion de Dirac $(\delta(n) = \begin{cases} 1 & \text{si } n = 0 \\ 0 & \text{ailleurs} \end{cases})$

Introduction:

Avec l'expansion du multimédia, une nouvelle problème est apparue : exploiter les quantités énormes et croissantes de documents numériques et particulièrement images numériques disponibles. Pour illustrer cette montée en puissance, selon [LAW 99], 180 millions d'images (indexable) soit environ de 3 téraoctets, étaient réparties sur 800 millions de pages disponibles sur Internet.

Les images sont en générale compressées puis archivées dans des bases de données, accessibles par les réseaux de télécommunication. Afin de pouvoir à la réutilisation de ces données, il est donc nécessairement de trouver des outils efficaces de manipulation. Au début, des systèmes basés sur l'annotation textuelle (mots clés associés aux images) ont été proposés; cependant le mot clé est une source d'information externe manuellement attachée aux documents. Le procédé d'indexation par mots clés est donc long et coûteux puisque manuel, et délicat de sursoit car la pertinence de termes attribués aux images conditionne la qualité des recherches futures. En d'autre part, il apparaît qu'une liste réduite de mots clés ne peut couvrir complètement la sémantique souvent riche par l'image. L'approche basé sur le contenu, visant à extraire directement l'information nécessaire à sa caractérisation, apparaît comme une alternative à l'approche textuelle.

L'émergence de la recherche d'images par le contenu remonte au début des années 90. depuis cette date, nous avons vu apparaître plusieurs systèmes comme QBIC chez IBM [FLI 95] ou bien SurfImage [NAS 98] à l'INRIA.

L'approche basé sur la recherche par le contenu, vise à utiliser les caractéristiques visuelles de l'image qui sont la couleur, la texture et la forme pour extraire un vecteur multidimensionnel pour indexer une image. La couleur peut être représenté par un histogramme et cela peut constituer la perte d'information spatiale qu'elle porte l'image et d'autre part pose le problème de discrimination des histogrammes , dans l'autre coté la forme est obtenu par la segmentation de l'image et cela est impratique à cause de deux raisons : la première, les artefacts qui peuvent être introduits suite à cette opération, et surtout pour les image à haute fréquence, la deuxième, le coût élevé de la segmentation. La texture nous a apparu comme une caractéristique la plus stable par apport les deux autres.

Dans ce travail, nous appliquons la transformé en ondelettes pour analyser la texture pour trouver l'index d'une image. Dans l'autre coté nous donnons un survole sur les plateformes distribuées sur lesquels peut être implanté un système de recherche d'images par le contenu.

Ce mémoire est focalisé autour quatre chapitres. Dans le premier, on présente l'architecture d'un système de recherche d'images par le contenu, ainsi que les différentes approches utilisées dans l'indexation à l'issue des différentes constituants primaires d'une image, dans le deuxième chapitre, il y a une introduction sur les ondelettes, ainsi que son avantage par apport à l'analyse de Fourier, et son liaison pratique avec l'analyse d'images, dans le troisième chapitre, on expose une comparaison applicative entre deux algorithmes d'analyse de la texture en vue de l'indexation, et on présente deux méthode pour la mésure de similarité, dans le quatrième chapitre, on donne une description sur quelques architectures distribués où on espérera un système de recherche d'images par le contenu distribué.

Chapitre I :

La recherche d'informations visuelles

1. Introduction :

Récemment, la recherche d'images par le contenu (*CBIR :Content Based Image Retrieval*) est un champ de recherche très étudiée à cause de la profusion d'images numériques ainsi que la richesse sémantique de l'image. Les méthodes traditionnelles (basées sur le texte) de la recherche des images se basent sur l'annotation textuelle et manuelle de l'image. Il y a deux inconvénients. Premièrement, le coût élevé de l'annotation manuelle des milliers d'images. Deuxièmement, la richesse sémantique d'une image est difficile à décrire et des gens différents peuvent décrire la même image de différents points de vue (facteur de subjectivité) [FOU 02] [OGI]. Pour franchir ces inconvénients, la recherche des images par le contenu (*CBIR*) c'est une nouvelle approche qui permet la recherche des images directement et automatiquement basée sur les caractéristiques visuelles du contenu de l'image comme la couleur, la texture, la forme [FUH 04] [PAS 96].

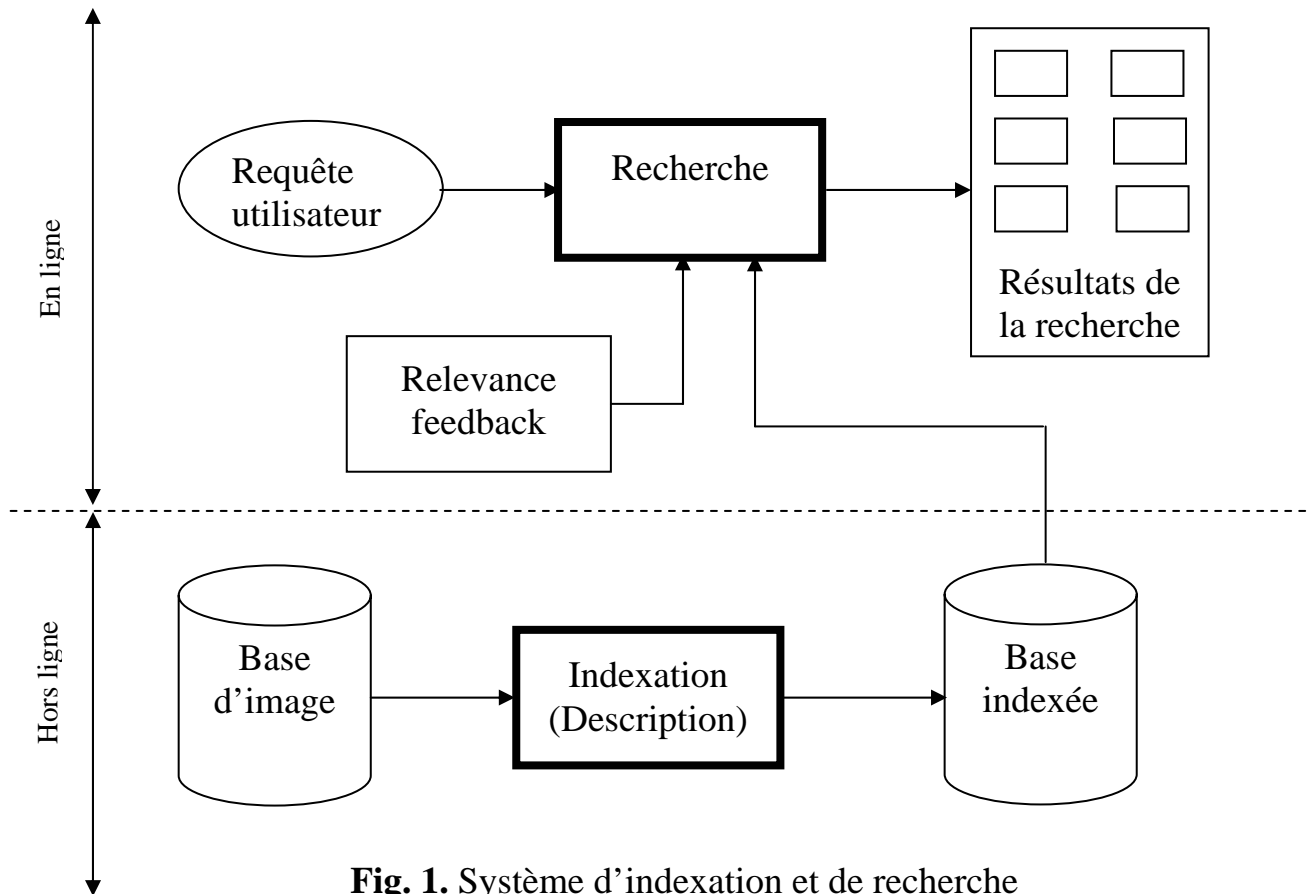


Fig. 1. Système d'indexation et de recherche d'images par le contenu

2. Méthodologie des systèmes d'indexation et de recherche:

Deux aspects indissociables coexistent dans notre problème, l'indexation et la recherche. Le premier concerne le mode de représentation informatique des images et le second concerne l'utilisation de cette représentation dans un but de recherche. L'architecture d'un système de recherche d'images, présentée en figure 1, se décompose en deux phases de traitement : une phase hors-ligne pour l'indexation ou le calcul des indexes des images de la base et une phase en-ligne de recherche.

2.1. La recherche des images :

Une première classification des méthodes de recherche d'images peut être proposée en considérant le but visé par l'utilisateur. Cette classification, introduite par Cox et al. dans [COX 00] et reprise par Smeulders et al. [SME 00] distingue trois grandes catégories :

- la recherche associative (search by association),
- la recherche de cible (target search),
- la recherche de catégorie (category search).

En premier lieu, la recherche associative est un type de recherche au but vague procédant par exploration de la base d'images. L'utilisateur ne possède pas dans ce cas d'exemple précis de ce qu'il recherche (il peut d'ailleurs ne pas savoir ce qu'il recherche), c'est l'exploration qui permet, au moyen d'une interaction système-utilisateur forte, de définir et d'affiner le but de la recherche.

En second lieu et au contraire de l'approche précédente, la recherche de cible a un but clairement défini, à savoir retrouver une image particulière, précise que l'utilisateur a en tête, par exemple la photographie d'un objet donné. Cette approche fait appel à un processus interactif permettant de converger vers l'image cherchée.

En troisième et dernier lieu, la recherche de catégorie a pour objectif de retourner une, plusieurs ou bien toutes les images appartenant à une catégorie donnée. C'est l'utilisateur qui fixe les limites des catégories en interagissant avec le système.

Une seconde classification des techniques de recherche d'images permet de distinguer deux grandes méthodes :

- la recherche par similarité.
- la modélisation de la pertinence (relevance feedback).

Nous appelons recherche par similarité toute recherche initiée par un exemple (image, croquis, région dans une image,..) fourni par l'utilisateur au système. La recherche se base sur une mesure de similarité calculée entre les images cibles et la requête. En conséquence, les images résultats retournées par le système sont proches de l'exemple, autrement dit, similaires à la requête au sens de la mesure adoptée. Cette approche est donc fortement dépendante de la requête, qui peut être vue comme un point d'ancrage (dans l'espace de recherche) pour le calcul de similarité, basé sur une fonction propre au système.

La recherche par modélisation de la pertinence n'est généralement pas initiée par un exemple. Une valeur traduisant l'adéquation à la requête, généralement une probabilité, est affectée aux images. La densité de probabilité construite pour l'ensemble de la base est initialisée uniformément, puis mise à jour sur le principe du bouclage de pertinence. Les modes qui émergent de la densité de probabilité, au fil des itérations, correspondent théoriquement aux images recherchées.

2.2. Indexation des images :

Nous identifions trois types de modèles d'indexation : le modèle booléen, le modèle vectoriel et le modèle probabiliste [ROU 99].

- Pour le modèle booléen, tout d'abord, les images sont caractérisées par une liste de descripteurs et la requête est une formule logique qui combine à la fois des descripteurs exemples et des opérateurs logiques du type ET, OU, NON, etc. En réponse, le système effectue une classification en deux classes correspondant d'une part aux images qui satisfont la requête et d'autre part à celles qui ne la satisfont pas.

- Dans le modèle vectoriel, l'image requête et les images cibles, c'est-à-dire les images de la base, sont représentées par un vecteur dans un espace d'attributs. Ce vecteur correspond à la concaténation des pondérations sur des descripteurs basiques. Une fonction de similarité entre vecteurs est utilisée pour classer les images en fonction de leur adéquation à la requête. Ce modèle est le choix retenu dans ce mémoire.
- Dans le modèle probabiliste, une probabilité de pertinence de l'image, en réponse à la requête, est attribuée à chacun des descripteurs. Sous hypothèse d'indépendance des descripteurs, il est possible de calculer la probabilité que l'image réponde à la requête de l'utilisateur comme le produit des probabilités précédentes. Ce type de modèle fait appel à une interaction utilisateur forte par l'intermédiaire, par exemple, de jugements de pertinence émis par l'utilisateur sur les propositions de résultats faites par le système.

3. Descripteurs du contenu de l'image :

Le but de l'indexation est de fournir une représentation image permettant des recherches efficaces. Il ne s'agit pas de coder toute l'information portée par l'image mais de se concentrer sur l'information qui permet de traduire efficacement une similarité proche des besoins exprimés par un utilisateur. Sans dissocier l'extraction des attributs de la structuration de l'information sous forme de signature (index) et de l'utilisation de cette représentation pour la recherche, une des clés de l'indexation efficace est l'identification de caractéristiques primaires en accord avec le type et le but des recherches visées par le système.

3.1 Indexation et recherche par la couleur :

La couleur est une des attributs les plus employés en indexation d'images. Le premier point concerne le choix de l'espace colorimétrique. La littérature dense sur cet aspect montre qu'il n'y a pas d'espace couleur idéal pour l'indexation. Si certains se contentent de travailler dans l'espace RGB, d'autres font le choix de projeter les images dans des espaces colorimétriques alternatifs caractérisés par des propriétés différentes.

3.1.1. Les espaces de couleur :

Tout pixel d'une image peut être représenté comme un point dans un espace couleur 3D. Généralement les espaces de couleurs utilisés dans la recherche des images sont : RGB, CIE L*u*v*, CIE L*a*b*, HSV, et l'espace de couleur opposant. Il n'y a pas d'accord qu'il y a un meilleur. Néanmoins, une des caractéristiques désirée d'un espace de couleur approprié est leur uniformité. L'uniformité consiste que s'il y a deux couleurs sont plus similaires dans l'espace de couleurs alors sont aussi plus similaires par les utilisateurs. En d'autres termes, la proximité mesurée entre les couleurs doit être relié directement à la similarité psychologique entre eux.

L'espace RGB est l'un des espaces les plus utilisé dans l'affichage des images. Il est composé de trois couleurs rouge, vert et bleu. Ces composants sont appelés les primaires additifs puisque l'espace RGB est le produit de leurs mélanges. A l'opposé du RGB, L'espace CMY est utilisé principalement dans l'impression. Les trois couleurs composantes sont le cyan, le magenta et le jaune. Ces trois composantes sont appelées les primaires soustractifs puisque l'espace CMY est produit via l'absorption de la lumière. Les deux espaces RGB et CMY sont orientés matériels et perceptuellement non-uniforme.

Les espaces CIE $L^*u^*v^*$, CIE $L^*a^*b^*$ sont orientés utilisateurs et perceptuellement uniforme. Il sont constitués de la composante luminance (L) et deux composantes chromatiques a et b ou u et v. CIE $L^*a^*b^*$ a été conçu pour être utiliser avec les mixtures colorants soustractifs, tandis que CIE $L^*u^*v^*$ a été conçu pour être utiliser avec les mixtures colorants additifs.

L'espace HSV est largement utilisé dans l'informatique graphique et est une méthode plus intuitive pour caractériser la couleur. Les trois couleurs composantes sont la teinte, la saturation et le degré de luminosité.

L'espace de couleur opposant utilise les axes de couleur (R-G, 2B-R-G, R+G+B). L'avantage de cette représentation consiste à isoler l'information de la luminosité dans le troisième axe.

3.1.2. Histogramme :

L'histogramme constitue une approximation de la densité de probabilité associée à l'image, vue comme une variable aléatoire. Cette approche statistique est proche de la recherche textuelle pour laquelle les documents sont couramment caractérisés par la fréquence d'apparition de mots-clés.

En 1991, Swain et Ballard [SWA 91] sont les premiers à utiliser l'histogramme pour l'indexation par la couleur. L'histogramme de couleur est utilisé comme le résumé visuel, ou signature de l'image donc constitue l'index de L'image. La raison du succès de l'histogramme résulte de sa simplicité de calcul et d'utilisation, assortie d'une relative robustesse à la rotation, aux changements d'échelle et aux occlusions. En revanche L'histogramme présente deux inconvénients. En premier lieu, cette représentation conduit à la perte de l'information spatiale, caractéristique importante afin de correctement représenter le contenu de l'image. En second lieu, l'utilisation de l'histogramme pose le problème de la quantification des espaces de caractéristiques. En effet, la

taille et l'efficacité de l'index, composantes essentielles en indexation, varient en fonction de la palette d'attributs. Ce dernier aspect est, par ailleurs, à rattacher au choix de la mesure de similarité.

3.1.3. Information spatiale :

L'inconvénient majeur de l'indexation par histogramme est la perte de l'information spatiale. En laissant pour le moment de côté les méthodes basées sur la segmentation de l'image, nous pouvons distinguer deux grandes approches pour la prise en compte de l'information spatiale lors de l'indexation : l'approche relative et l'approche absolue. La première vise à introduire une notion de cohérence spatiale entre les caractéristiques primaires calculées pour chaque primitive image, il s'agit de représenter les liens relatifs entre primitives. La seconde approche tend plutôt à coder une information spatiale absolue : la position du pixel (indice de ligne et de colonne) dans l'image est alors la donnée de base.

3.1.4. Les moments de couleur :

Les moments de couleur ont été utilisés avec succès dans plusieurs systèmes de recherche d'images (comme QBIC), notamment lorsque le contenu de l'image contient juste un objet. Le premier ordre (moyen), le deuxième ordre (variance) et le troisième ordre (déviation) de moments de couleur constituent un moyen efficace dans la représentation des distributions des couleurs des images. Mathématiquement les moments de couleur sont définis comme :

$$\begin{aligned}\mu_i &= \frac{1}{N} \sum_{j=1}^N f_{ij} \\ \sigma_i &= \left(\frac{1}{N} \sum_{j=1}^N (f_{ij} - \mu_i)^2 \right)^{\frac{1}{2}} \\ s_i &= \left(\frac{1}{N} \sum_{j=1}^N (f_{ij} - \mu_i)^3 \right)^{\frac{1}{3}}\end{aligned}$$

où f_{ij} est la valeur de i -th couleur composante du pixel j de l'image, et N le nombre des pixels dans l'image. Puisque seulement 9 (trois moments pour chaque trois couleurs composantes) nombres sont utilisés pour représenter le contenu en couleur pour chaque image, la méthode de moments de couleur c'est une représentation plus compact. A cause de cette compacité cette méthode peut diminuer la discrimination entre les images.

3.1.5. Corrélogramme de couleurs :

Le corrélogramme de couleur est proposé pour caractériser non seulement la distribution de couleur des pixels, mais aussi la corrélation spatiale entre les pairs de couleurs. Le premier et le deuxième dimension représentent les couleurs de chaque paire de pixels et le troisième dimension représente leur distance spatiale. Le corrélogramme de couleur est un tableau indexé par des pairs de couleurs, ou le k -eme entré de (i, j) spécifie la probabilité de trouver un pixel de couleur j à une distance k d'un pixel de couleur i dans une image. Soit I l'ensemble de pixels d'une image et $I_{c(i)}$ l'ensemble de pixels dont la couleur est $c(i)$. Alors le corrélogramme de couleur est défini par :

$$\gamma_{i,j}^{(k)} = \Pr_{p_1 \in I_{c(i)}, p_2 \in I} [p_2 \in I_{c(j)}, |p_1 - p_2| = k]$$

Où $i, j \in \{1, 2, \dots, N\}, k \in \{1, 2, \dots, d\}$, et $|p_1 - p_2|$ est la distance entre les pixels p_1 et p_2 . Si on considère toutes les combinaisons de paires de couleurs alors la taille du corrélogramme de couleurs est considérable ($O(N^2d)$), partant une version simplifiée appelée autocorrélogramme est fréquemment utilisée. L'autocorrélogramme de couleurs capture seulement la corrélation spatiale entre les couleurs identiques et alors diminue la dimension à $O(Nd)$.

En comparant avec l'histogramme de couleurs et CCV, l'autocorrélogramme de couleurs prouve leur puissance dans la recherche

d'image mais l'inconvénient réside dans le coût de calcul à cause leur haute dimension.

3.1.6. Vecteur de cohérence couleur :

Le vecteur de cohérence couleur CCV est proposé par [PAS 96]. Les pixels sont tout d'abord regroupés en deux classes cohérent/incohérent suivant qu'ils appartiennent ou non à une région de couleur homogène. Soit α_i dénote le nombre de pixels cohérents dans la i -th couleur et β_i dénote le nombre de pixels incohérents dans une image. Alors le CCV de l'image est défini comme le vecteur $\langle (\alpha_1, \beta_1), (\alpha_2, \beta_2), \dots, (\alpha_N, \beta_N) \rangle$. Noter que $\langle \alpha_1 + \beta_1, \alpha_2 + \beta_2, \dots, \alpha_N + \beta_N \rangle$ est l'histogramme couleur de l'image. Donc deux histogrammes correspondant à une accumulation sur chacune des deux classes servent à indexer les images. L'avantage de cette approche est l'ajout de l'information spatiale à l'histogramme et cela à partir leur raffinement, cependant, elle présente l'inconvénient d'amplifier la sensibilité aux conditions d'illumination, contrairement au corrélogramme.

3.2 Indexation et recherche par la forme :

Les attributs de forme sont complémentaires de la description couleur. Nous distinguons deux catégories de descripteurs de formes : les descripteurs basés régions et les descripteurs basés frontières (contours).

Les premiers font classiquement référence aux moments invariants [DER 99] [HU 62] et sont utilisés pour caractériser l'intégralité de la forme d'une région. Ces attributs sont robustes aux transformations géométriques comme la translation, la rotation et le changement d'échelle. La seconde approche faite classiquement référence aux descripteurs de Fourier [PER 77] [RUI 96] et porte sur une caractérisation des contours de la forme. Les attributs de forme sont utilisés pour caractériser les objets dans les images. La plupart des travaux

laissent de côté l'aspect segmentation, souvent nécessaire à l'extraction de ces objets. Or, la difficulté du problème est réelle, notamment dans le cadre de l'indexation, lorsqu'il s'agit de segmenter automatiquement l'intégralité d'une base d'images.

3.2.1. Segmentation et détection de points d'intérêts :

La segmentation des images est une question centrale en traitement de l'image [COC 95]. En indexation, de nombreux auteurs se sont focalisés sur cet aspect et des algorithmes de segmentation originaux ont été proposés [BOU 00] [PAU 99] [WIN 99] ainsi que des techniques complètes pour l'indexation et la recherche [CAR 99]. La difficulté particulière qui se pose pour ce type de travaux vient de la nécessité de segmenter automatiquement un grand nombre d'images (les images de la base). Or, cette contrainte est encore trop forte pour que l'on puisse se satisfaire des résultats obtenus par les techniques actuelles de segmentation. Les approches faisant appel à la segmentation sont principalement développées dans le cadre de l'indexation partielle des images même si parfois il s'agit d'une étape intermédiaire pour le calcul de signatures (indexes) globales.

La focalisation sur des zones particulières de l'image peut également se faire autour de points d'intérêt. Ces pixels sont définis comme concentrant l'information de l'image et correspondent à des pixels de fortes courbures [HAR 88][ITT 98][SEB 00]. La caractérisation locale se fait par le calcul d'invariants différentiels sur un voisinage autour des points d'intérêt [SCH 95] [TUY 99].

3.3. L'indexation et la recherche par la texture :

La texture est une autre caractéristique importante des images. En fait, les méthodes de représentation de la texture peuvent être classées en deux catégories : structurelles et statistiques. Les méthodes structurelles, comportant

opérateur morphologique et graphe d'adjacence, décrit la texture par l'identification des primitives structurales et leurs rôles de placement. Il vaut mieux avec les textures régulières. Les méthodes statistiques, comportant les matrices de co-occurrence, caractéristique de *Tamura*, décomposition hautaine, champ aléatoire de *Markov* et les techniques de filtrage multirésolution comme la transformation de *Gabor* et en ondelettes, caractérisent la texture par la distribution des niveaux de gris de l'image. Dans cette section nous introduisons quelques représentations de la texture qui sont utilisées dans les systèmes de recherches d'images (CBIR).

3.3.1 Caractéristiques de *Tamura* :

Les caractéristiques de *Tamura* [TAM 78], comportant la grossièreté (*coarseness*), le contraste, et la Directivité sont conçues conformément avec les études psychologiques pour la perception de la texture par l'être humain. Ces caractéristiques sont données comme suivant :

a. La grossièreté (*coarseness*) :

La grossièreté est la mesure de la granularité de la texture. Pour calculer la grossièreté, premièrement les moyens mobiles $A_k(x, y)$ sont calculés en utilisant $2^k \times 2^k$ ($k = 0, 1, \dots, 5$) la taille des fenêtres à chaque pixel (x, y) i.e.,

$$A_k(x, y) = \frac{1}{2^{2k}} \sum_{i=x-2^{k-1}}^{x+2^{k-1}-1} \sum_{j=y-2^{k-1}}^{y+2^{k-1}-1} g(i, j)$$

Où $g(i, j)$ est le niveau de gris à (i, j) .

Alors, les différences entre moyens mobiles non chevauchants dans les directions horizontale et verticale de chaque pixel sont données par :

$$E_{k,h}(x, y) = |A_k(x + 2^{k-1}, y) - A_k(x - 2^{k-1}, y)|$$

$$E_{k,v}(x, y) = |A_k(x, y + 2^{k-1}) - A_k(x, y - 2^{k-1})|$$

Donc, la valeur de k qui maximise E dans n'importe quelle direction sera utilisée d'être la meilleure taille pour chaque pixel i.e.,

$$S_{best}(x, y) = 2^k$$

La grossièreté (coarseness) est mesurée par le moyennement du S_{best} sur la totalité de l'image, i.e.,

$$F_{crs} = \frac{1}{m \times n} \sum_{i=1}^m \sum_{j=1}^n S_{best}(i, j).$$

b. Le contraste :

Le contraste est donné par la formule :

$$F_{con} = \frac{\sigma}{\alpha_4^{1/4}}$$

Où, $\alpha_4 = \mu_4 / \sigma^4$, μ_4 est le quatrième moment du moyen, et σ^2 est la variance.

c. La directivité :

Pour calculer la directivité, l'image est convoluée par deux 3x3 matrices

i.e. :

$$\begin{matrix} -1 & 0 & 1 & 1 & 1 & 1 \\ -1 & 0 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & -1 & -1 & -1 \end{matrix}$$

Et le vecteur gradient est calculé pour chaque

pixel.

La magnitude et l'angle de cet vecteur sont définis par :

$$|\Delta G| = (|\Delta_H| + |\Delta_V|) / 2$$

$$\theta = \tan^{-1}(\Delta_V / \Delta_H) + \pi/2$$

Où Δ_H et Δ_V sont les différences horizontale et verticale de la convolution.

Un histogramme de θ peut être construit, en calculant le nombre de pixels pour chaque direction. Cet histogramme exhibe les hautes directions de l'image.

3.3.2. Caractéristiques de Wold :

La décomposition de Wold [PIC 96][FRA 93] montre une autre approche pour décrire les textures en termes des propriétés perceptuelles. Les trois composant de Wold, harmonique, évanescence et indéterministique correspondent à la périodicité, directivité et le caractère aléatoire de la texture respectivement. Les textures périodiques sont caractérisées par leurs fort composant harmonique, les textures d'haut directivité sont caractérisées par leurs fort composant d'évanescence tandis que les textures moins structurées sont caractérisées par leurs fort composant indéterministique.

Soit un champ aléatoire homogène et régulière $\{y(m,n), (m,n) \in Z^2\}$. La décomposition 2D de Wold permet de décomposer le champ en trois composants orthogonaux :

$$y(m,n) = u(m,n) + d(m,n) = u(m,n) + h(m,n) + e(m,n)$$

Où, $u(m,n)$ est le composant indéterministique, et $d(m,n)$ est le composant déterministique lequel peut être décomposé en composant harmonique $h(m,n)$ et composant d'évanescence $e(m,n)$. Les trois composants orthogonaux peuvent être obtenus par le recours à un système d'équation linéaire.

3.3.3. Modèle Auto-Regressif Simultané (SAR) :

Le modèle SAR est une instance des modèles MRF (champ aléatoire de Markov) est l'une des technique utilisée dans l'analyse des textures. En comparaison avec d'autre MRF le modèle SAR utilise moins de paramètres.

Dans le modèle SAR, l'intensité d'un pixel est considérée comme une variable aléatoire. L'intensité $g(x, y)$ à un pixel (x, y) peut être estimée comme une combinaison linéaire des valeurs $g(x', y')$ des pixels voisins et un terme additif $\varepsilon(x, y)$ c'est-à-dire :

$$g(x, y) = \mu + \sum_{(x', y') \in D} \theta(x', y') g(x', y') + \varepsilon(x, y)$$

Où : μ représente le moyen dans l'image entière, D représente le voisinage du pixel (x, y) , $\theta(x', y')$ désigne le poids de chaque pixel (x', y') du voisinage D , $\varepsilon(x, y)$ est une variable aléatoire gaussienne de moyen 0 et de variance σ^2 . Les paramètres θ et σ sont utilisés pour mesurer la texture. Le paramètre σ détermine la granularité de la texture tandis que le paramètre θ indique la directivité de la texture. Il existe plusieurs techniques pour estimer les paramètres du modèle SAR parmi lesquelles on cite LES (Least Square Error), MLE (Maximum Likelihood estimation),...

3.3.4. La transformée en ondelettes :

La transformée en ondelettes (WT) consiste à décomposer un signal en utilisant une famille de fonctions $\psi_{m,n}(x)$ obtenus par la translation et la dilatation d'une fonction mère $\psi(x)$ s'appelle ondelette. Cette famille de fonctions constitue la base de projection du signal.

Sur le plan de calcul numérique l'analyse multirésolution permet un calcul aisé et rapide de la transformée en ondelettes, en introduisant une fonction supplémentaire dite d'échelle, notée φ . La décomposition d'un signal 2D (image numérique) est présentée sous la forme d'un filtrage par bancs de filtres à n canaux. Dans le cas d'analyse multirésolution séparable $n = 4$ et pour le cas d'analyse multirésolution non séparable $n = 2$.

Le schéma d'analyse multirésolution ainsi construit est à rapprocher du modèle du système visuel humain construit par les psychophysiciens. En effet elle existe une similitude entre ces deux systèmes, mais aussi des différences importantes en particulier la non séparabilité en horizontale et verticale du système visuel humain, mais dans notre implémentation on utilisera la transformée non séparable, ce dernier peut éviter relativement ces différences.

4. Conclusion :

Dans ce chapitre, nous avons introduit quelques techniques fondamentales de l'indexation et la recherche d'images par le contenu. Les caractéristiques visuelles généraux de l'image sont les plus utilisées pour cet effet, comme la couleur, la forme, et la texture. La couleur est généralement représentée par l'histogramme, cet dernier est utilisé comme le résumé visuel, ou signature. Tellement sa simplicité de calcul et d'utilisation présente l'inconvénient majeur que cette représentation conduit à la perte d'information spatiale, caractéristique importante afin de correctement représenter le contenu de l'image. La forme est obtenue généralement par la segmentation des images. Malgré le succès des techniques basées sur la forme dans d'autres applications, la difficulté du problème est réelle, notamment dans le cadre de l'indexation, lorsqu'il s'agit de segmenter automatiquement toute l'intégralité d'une base d'images. La texture peut être représentée par les caractéristiques de Tamura, caractéristique de Wold, modèle auto-régressif simultané (SAR),... avec ces techniques la texture peut être analysée à une seule échelle d'une façon et d'une autre façon ces techniques nous donnent une représentation plus compacte, à cause de cette compacité ces techniques peuvent diminuer la discrimination entre les images. Nous voyons que les ondelettes sont plus avantageuses pour une analyse multi-échelle avec une efficacité de calcul en utilisant les bancs de filtres.

Chapitre II :

Les Ondelettes

1. Introduction :

Un CBIR est constitué de deux phases primordiales : l'indexation et la recherche. L'indexation est basée sur l'analyse des images pour obtenir des indexes pour les images, tandis que la recherche est basée sur la mesure de proximité entre ces indexes. Alors, l'analyse des images constitue le noyau d'une opération d'indexation.

Dans ce chapitre nous abordons les ondelettes communément avec l'analyse multirésolution comme un outil puissant dans l'analyse d'images, ainsi que son avantage par apport à l'analyse de Fourier. Pour les calculs numériques de la transformée en ondelette nous faisons appel au banc des filtres.

2. De l'analyse de Fourier à l'analyse en ondelette :

2.1 Analyse de Fourier :

Une image est considérée comme un signal à deux dimensions. La plus célèbre et la plus ancienne des outils pour étudier un signal est l'analyse de Fourier. Lors de cette analyse le signal est décomposé sur un ensemble de fonctions de base qui sont le cosinus, le sinus, ou l'exponentielle imaginaire.

Très tôt dans l'histoire de l'analyse de signaux, il est apparu que la décomposition obtenue n'était pas toujours la plus satisfaisante tellement son immense succès. Ainsi que l'analyse de Fourier permet de connaître les différentes fréquences excitées dans un signal, c'est-à-dire son spectre, mais ne permet pas de savoir à quels instants ces fréquences ont été émises. Cette analyse donne une information globale et non locale, car les fonctions d'analyse utilisées sont des sinusoïdes qui oscillent indéfiniment sans s'amortir, ainsi des perturbations sur l'axe des temps influencent chaque point sur l'axe fréquentiel (Fig. 1), ce qui signifie que les moments où interviennent ces perturbations ne sont pas connus.

2.2. Transformée de Fourier à fenêtre glissante:

Gabor dans les années 1940 [GAB 46] découvre la première forme de représentation temps-fréquence. Sa technique consiste à découper le signal en différentes plages ou fenêtres de longueur fixée (Fig. 3). Chaque plage est alors étudiée séparément des autres par l'analyse traditionnelle de Fourier. Une telle méthode est appelée transformée de Fourier à fenêtre glissante ou 'Short Time Fourier transform' (STFT). L'idée consiste à tronquer le signal en ne le considérant que sur un intervalle fini $[-\tau, \tau]$. Ceci revient à multiplier le signal $s(t)$ par une fonction $\phi(t)$ qui est égale à zéros en dehors de l'intervalle d'étude.

Le produit :

$$s(t)\phi(t-b) = s_b(t) \quad (1)$$

contient alors les information de $s(t)$ autour de $t = b$.

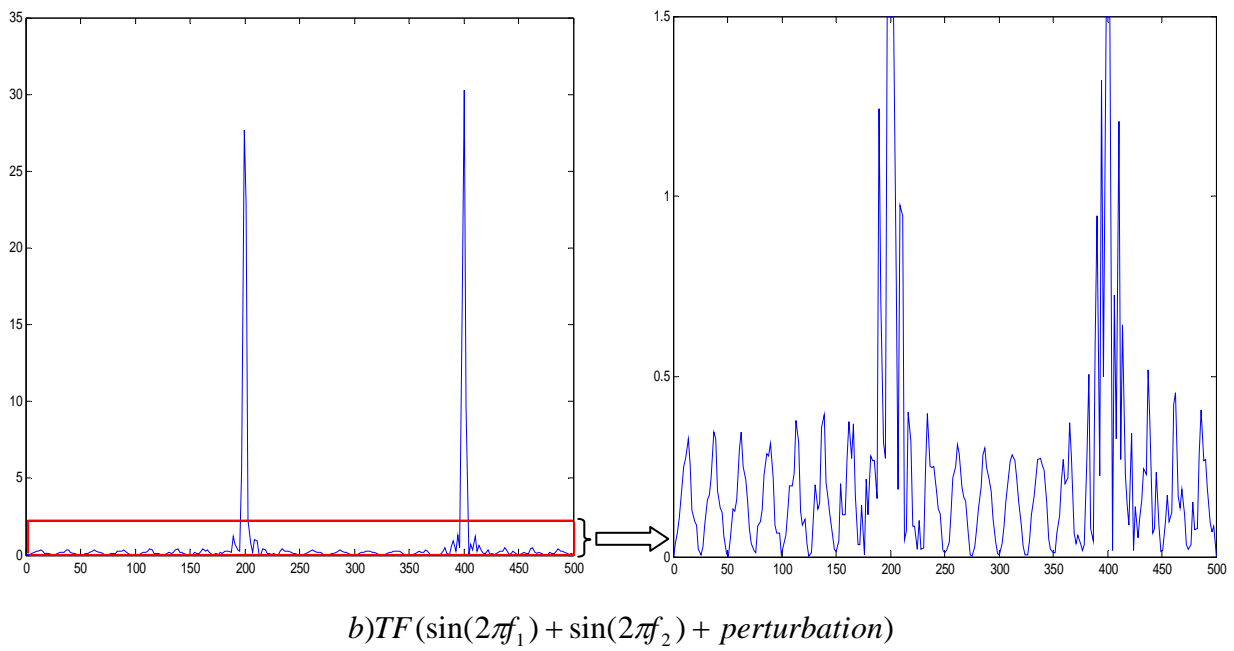
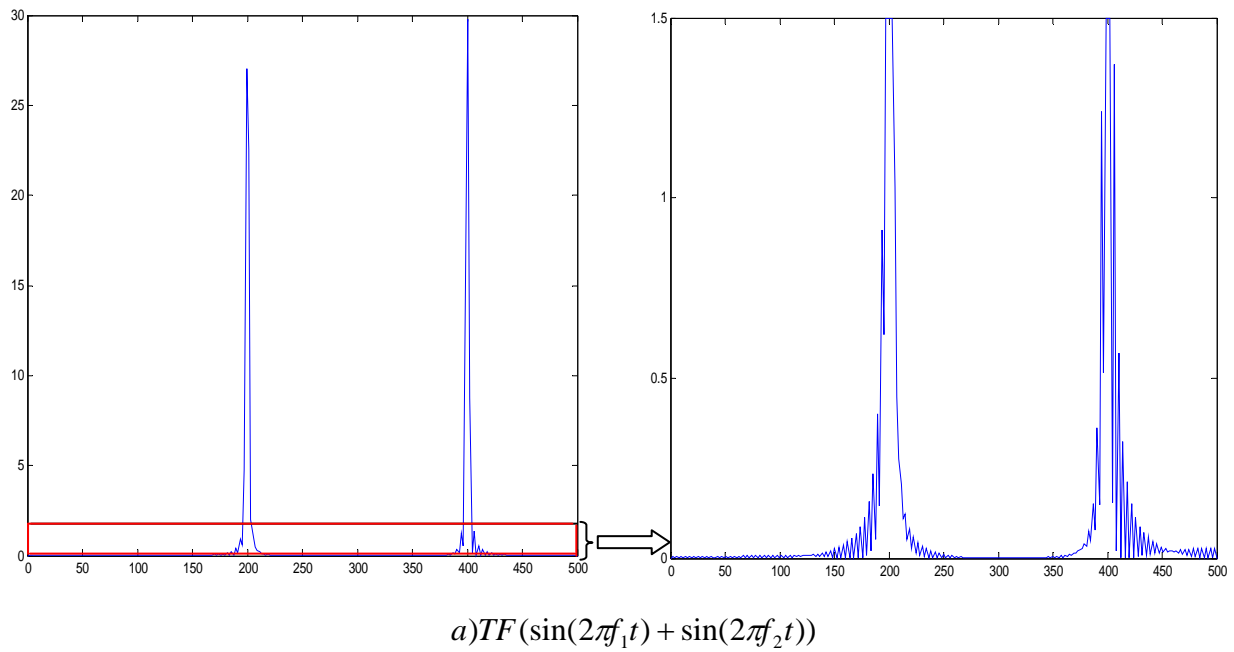


Fig.1 : Exemple de transformée de Fourier sur un signal non perturbé (a) avec $f_1 = 200\text{Hz}$ et $f_2 = 400\text{Hz}$ puis perturbée (b) aux instants $t_1 = 790\text{ms}$ et $t_2 = 1190\text{ms}$.

En particulier si ($\phi(t) = 1$ si $t \in [-\tau, \tau]$ et 0 d'ailleurs), voir Figure 2, alors :

$$s_b(t) = \begin{cases} s(t) & t \in [b-\tau, b+\tau] \\ 0 & \text{d'ailleurs} \end{cases} \quad (2)$$

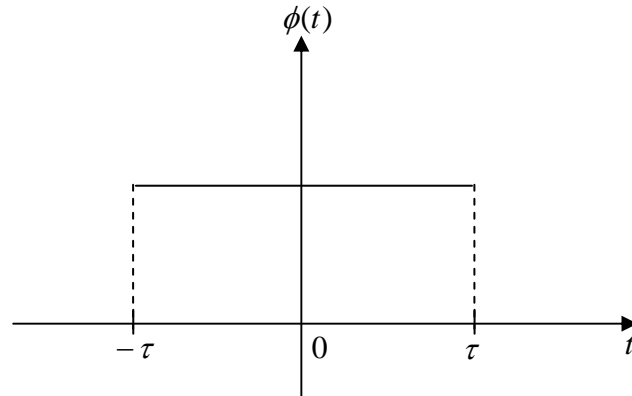


Fig.2 : La fonction $\phi(t)$

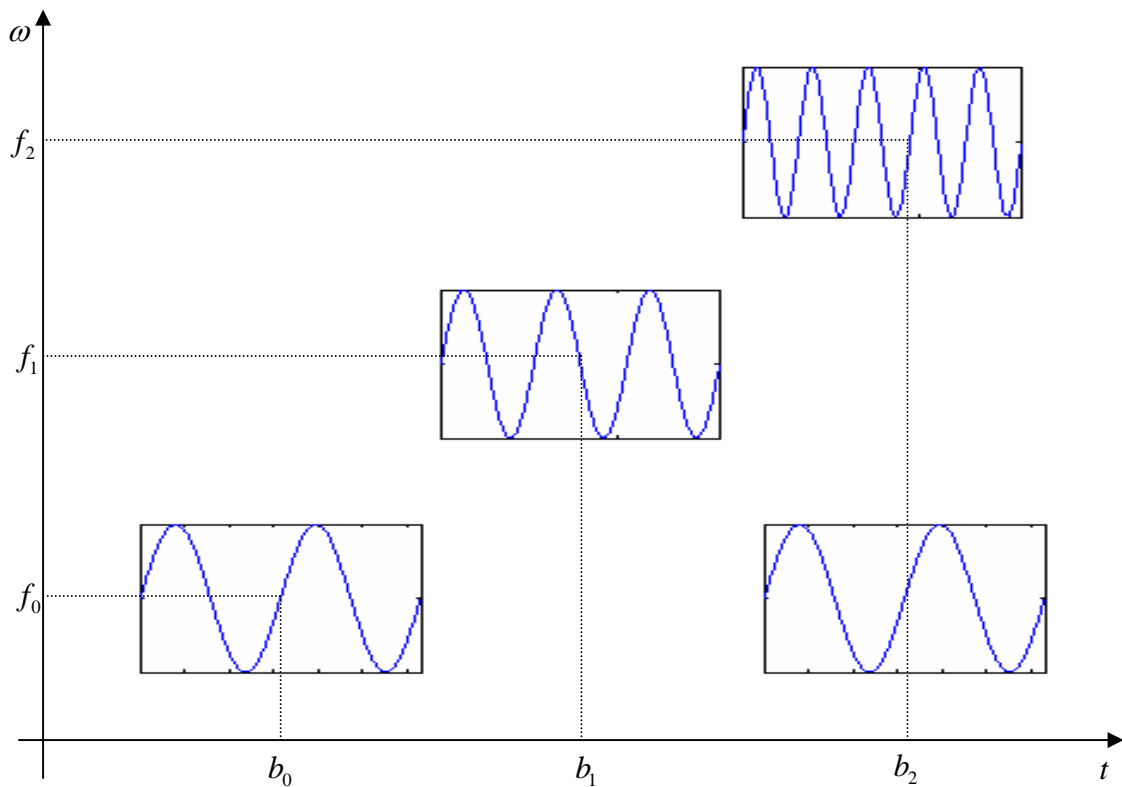


Fig. 3 : Fenêtre temps fréquence de la STFT

En changeant le paramètre b , la fenêtre se déplace le long de l'axe des temps. Ainsi le but est d'obtenir une fonction de fenêtrage correctement définie à la fois en temps et en fréquence. Un compromis entre résolution temporelle et résolution fréquentielle, donné par le principe d'incertitude de Heisenberg doit être envisagé.

Dans la STFT les informations temps-fréquences restent limitées par une incertitude, sur la localisation à la fois en temps et en fréquence. Cette méthode présente un inconvénient majeur d'avoir une fenêtre de longueur fixe. Il n'est donc pas possible d'analyser simultanément des phénomènes dont les échelles de temps sont différentes. De plus, le calcul de la STFT doit s'effectuer pour chaque changement de taille de la fenêtre d'où une charge de calcul importante.

2.3 Les ondelettes :

Les ondelettes, famille de fonctions déduites d'une même fonction, appelée ondelette mère, par opérations de translations et dilatations. En effet, cet outil permet la représentation de fonctions de L^2 , dans une base bien localisée en espace et en fréquence, offrant les avantages de l'analyse de Fourier et s'affranchissant des inconvénients du manque de localisation de cette dernière.

A partir d'une fonction de base ψ , appelée ondelette mère, on construit une famille de fonctions :

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) \quad a > 0, b \in \mathbb{R} \quad (3)$$

où le paramètre a est le facteur d'échelle (dilatation), et b est le paramètre de translation.

On définit alors les coefficients de la transformée en ondelettes d'un signal $s(t)$, comme étant les produits scalaires :

$$C_s(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{+\infty} s(t) \overline{\psi\left(\frac{t-b}{a}\right)} dt \quad (4)$$

Ces coefficients mesurent, en un certain sens, les fluctuations du signal $s(t)$ autour du point $t=b$, à l'échelle fournie par a . D'après l'équation (3), en diminuant a , le support de $\psi_{a,b}$ réduit en temps et donc couvre une plage en fréquence plus grande et vice versa (Fig. 4).

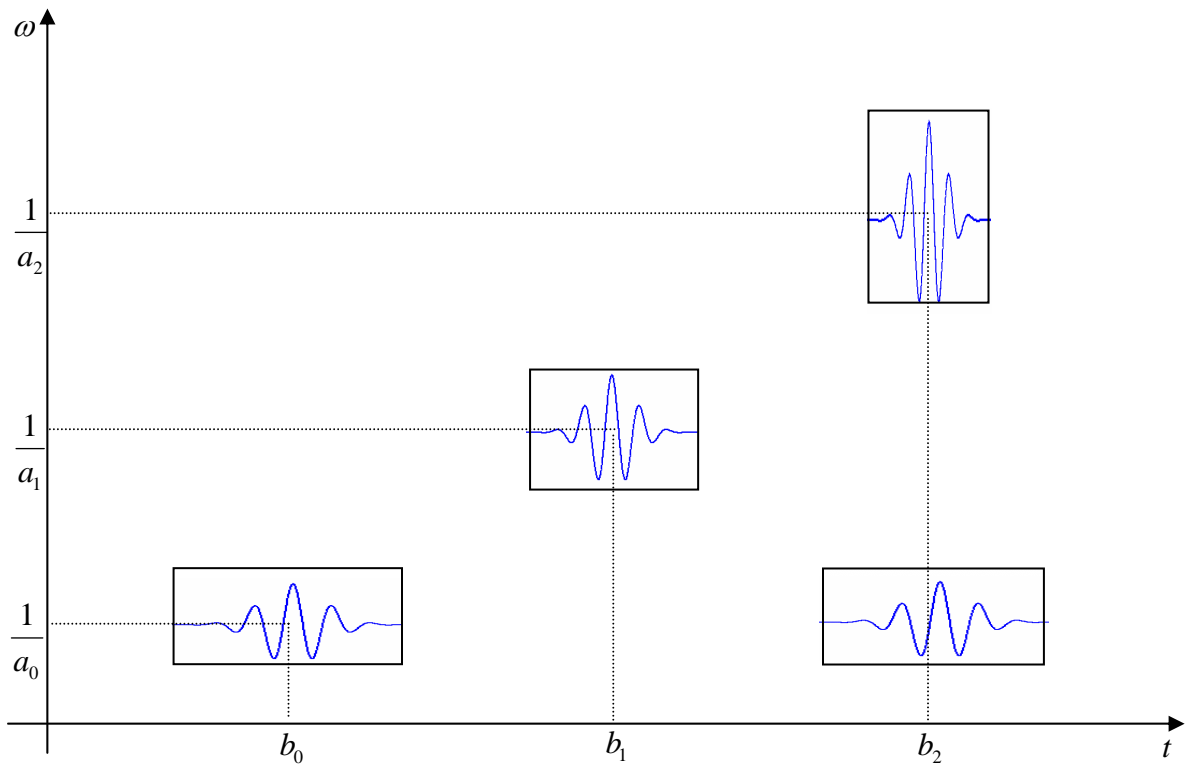


Fig.4 :Fenêtrage temps-fréquence pour la transformée en ondelette.

Pour un facteur d'échelle assez grand, la représentation des coefficients d'ondelettes en fonction de b , donne une représentation de la forme générale de la fonction. Par contre un facteur d'échelle faible correspond à une représentation des singularités. Cette propriété de microscope est très utile pour l'étude de la régularité d'une fonction. L'analyse en ondelette n'est pas unique, elle dépend de l'ondelette mère utilisée.

L'ondelette mère $\psi(t)$ devra avoir une bonne localisation, donc à l'infini, une convergence rapide vers 0 (Fig. 5), et devra être oscillante. On demande que l'intégrale de $\psi(t)$ soit nulle et qu'il en soit de même pour les m premiers moments de ψ . Cela s'écrit :

$$\int_{-\infty}^{+\infty} t^k \psi(t) dt = 0, \text{ pour } 0 \leq k \leq m \quad (5)$$

$$\int_{-\infty}^{+\infty} \psi(t) dt = \hat{\psi}(0) = 0 \quad (6)$$

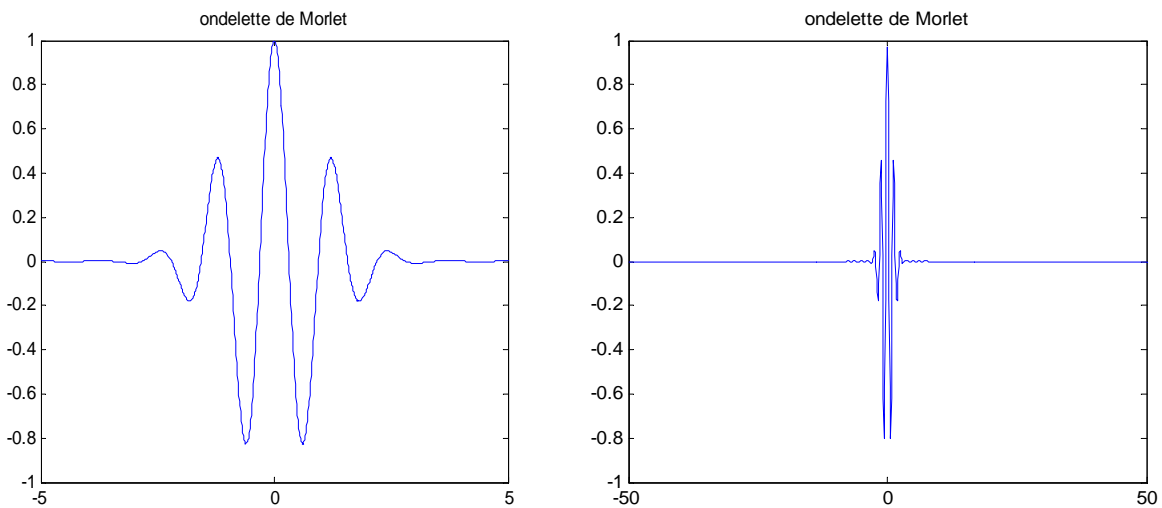


Fig. 5 : La bonne localisation de l'ondelette

La transformée en ondelette est inversible, ainsi le signal $s(t)$ peut être reconstruit après double intégration suivant le facteur d'échelle a et le paramètre de translation b :

$$s(t) = \frac{1}{C_\psi} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \frac{1}{a^2} C_s(a,b) \psi_{a,b}(t) da db \quad (7)$$

Cette possibilité reste théorique car le calcul n'est possible que numériquement. Le coefficient C_ψ est une constante qui dépend du choix de l'ondelette et est donnée par :

$$C_{\psi} = \int_{-\infty}^{+\infty} \frac{|\hat{\psi}(\omega)|^2}{|\omega|} d\omega < +\infty. \quad (8)$$

La transformée en ondelettes a les propriétés suivantes :

➤ **Linéarité :**

La transformée en ondelettes est linéaire, c'est-à-dire :

$$C_{\alpha s_1 + \beta s_2}(a, b) = \alpha C_{s_1}(a, b) + \beta C_{s_2}(a, b) \quad \alpha, \beta \in \mathbb{R}. \quad (9)$$

➤ **Translation :**

Une des propriétés importante de la transformée en ondelettes est l'invariance en translation :

$$C_{\tau_{b_0} s}(a, b) = C_s(a, b - b_0) \quad (10)$$

où $\tau_{b_0} s$ désigne la translation de s par le vecteur b_0 , c'est-à-dire $(\tau_{b_0} s)(t) = s(t - b_0)$.

➤ **Dilatation :**

La transformée en ondelettes a également une propriété de dilatation :

$$C_{\delta_k s}(a, b) = C_s(ka, kb) \quad , \quad \text{avec} \quad \delta_k s(t) = s(kt) \quad (11)$$

Sur la figure 6 nous illustrons un résultat de la transformée en ondelette sur un signal non perturbé (figure 6.a) puis perturbé aux instants $t_1 = 70$ et $t_2 = 400$ (figure 6.b). Nous constatons aisément à la différence de l'analyse de Fourier à quels instants ont lieu les deux perturbations.

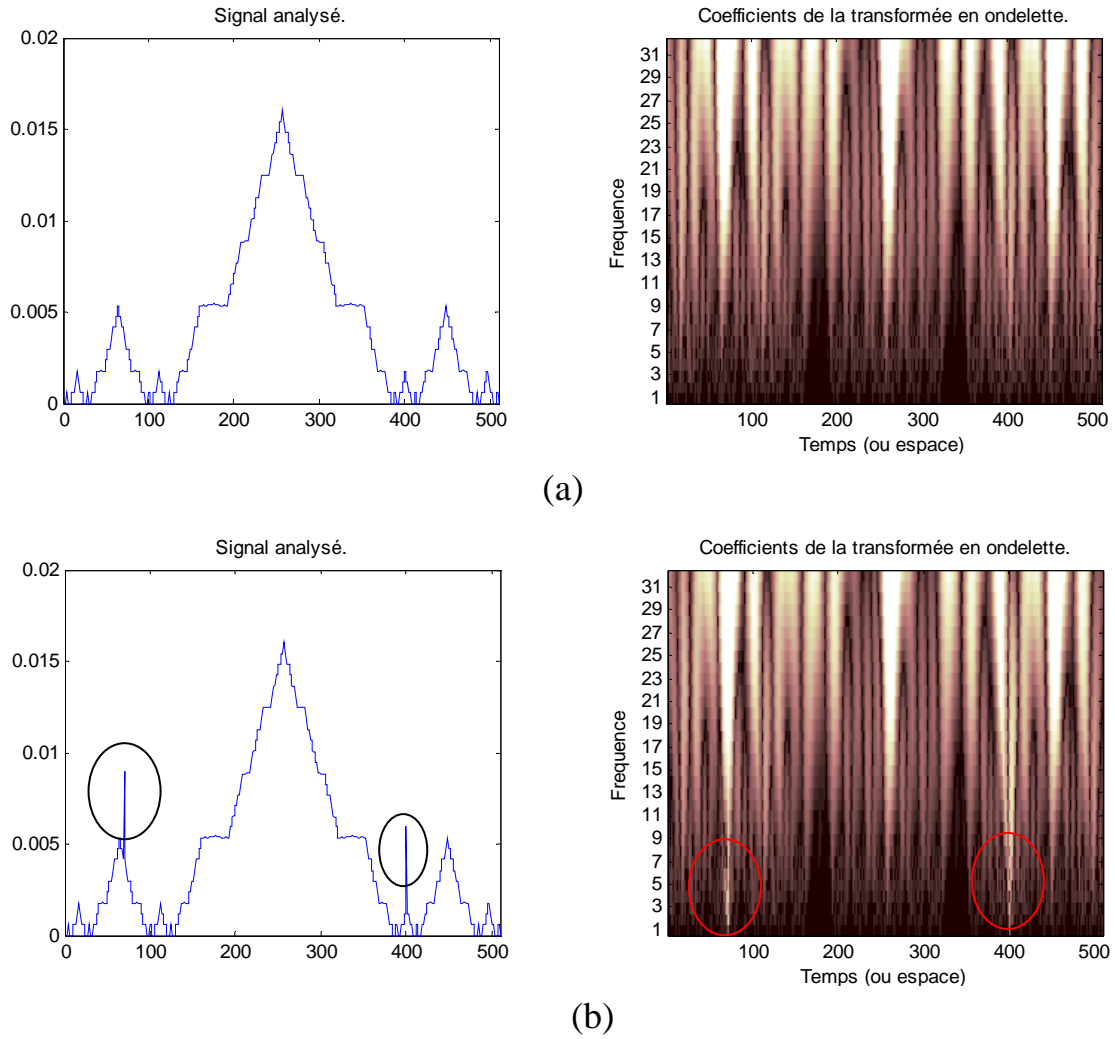


Fig. 6 : Exemple de la transformée en ondelette sur (a) un signal non perturbé puis (b) perturbé aux instants $t_1 = 70$ et $t_2 = 400$

2.3.1. Transformée en ondelettes discrète :

La formule (4) dépend des réels a et b , on peut décider une infinité de valeurs pour ces deux paramètres. On peut les faire varier continûment : on parle alors de transformée continue. De par cette aspect, la transformée en ondelette telle quelle est définie redondante, c'est-à-dire que l'on obtient plus de coefficients d'ondelettes qu'il n'en est nécessaire pour décrire le signal de manière exhaustive.

Les valeurs discrètes du facteur d'échelle a et du paramètre de translation b seront considérées sous la forme : $a = 2^{-j}$ et $b = k2^{-j}$ où $k, j \in Z$. Avec ces valeurs de a et b , l'équation (4) devient :

$$C_s(2^{-j}, k2^{-j}) = 2^{\frac{j}{2}} \int_{-\infty}^{+\infty} s(t) \psi(2^j t - k) dt \quad (12)$$

Si la fonction $s(t)$ est discrétisée, l'équation (12) s'écrit alors :

$$C_s(2^{-j}, k2^{-j}) = 2^{\frac{j}{2}} \sum_n s(n) \psi(2^j n - k) \quad (13).$$

3. banc de filtres :

Banc de filtres est un ensemble de filtres passe-bas, passe-bands et passe-hautes utilisé pour la décomposition et la composition spectrale des signaux. Il joue un rôle important dans l'analyse des images. La figure 7 illustre un banc de filtres à M -canaux. Le signal entrant est décomposé en M composants signal en appliquant M différents filtres d'analyse. Alors chaque composant signal représente une information du signal entrant.

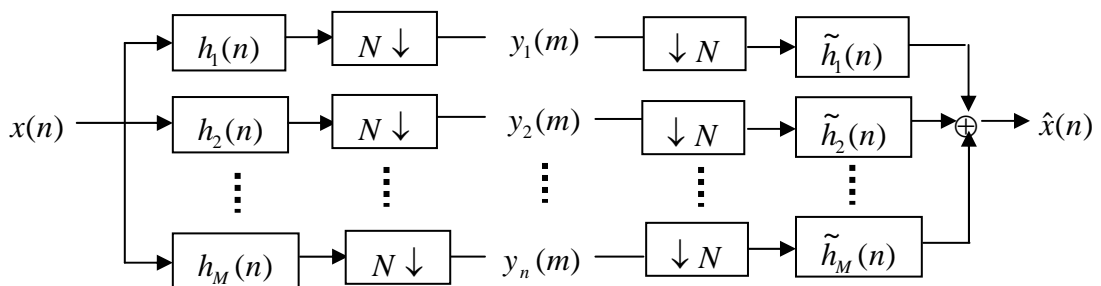


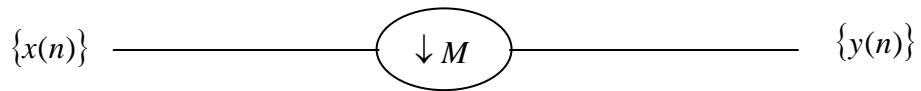
Fig. 7 : Banc de filtres a M canaux

3.1 Sous-échantillonnage :

Un sous-échantillonnage par M garde seulement un échantillon sur M d'un signal. Dans le domaine temporel (spatial), un sous-échantillonnage de M points d'une séquence d'entrée $\{x(n)\}$ s'écrit :

$$y(n) = x(nM) \quad (14)$$

Le schéma suivant décrit le système d'un sous-échantillonnage par M :



Dans la suite seul un sous-échantillonnage par $M = 2$ sera utilisé. En terme de matrice, l'équation (14) s'écrit :

$$[y] = [DEC_{\downarrow 2}] [x] \quad (15)$$

ou

$$\begin{bmatrix} \cdot \\ \cdot \\ y(-2) \\ y(-1) \\ y(0) \\ y(1) \\ y(2) \\ y(3) \\ \cdot \\ \cdot \\ \cdot \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \cdot & \cdot \\ 0 & 0 & 1 & 0 & 0 & \cdot & \cdot \\ \cdot & \cdot & 0 & 0 & 1 & 0 & 0 & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & 0 & 0 & 1 & 0 & 0 & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 0 & 0 & 1 & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cdot \\ \cdot \\ x(-2) \\ x(-1) \\ x(0) \\ x(1) \\ x(2) \\ x(3) \\ x(4) \\ \cdot \\ \cdot \end{bmatrix} \quad (16)$$

La matrice de sous-échantillonnage est orthogonale, $[DEC_{\downarrow 2}]^{-1} = [DEC_{\downarrow 2}]^t$ en conséquence, le sous-échantillonnage est une transformation orthogonale.

Dans le domaine de Fourier la version de (15) est donnée par :

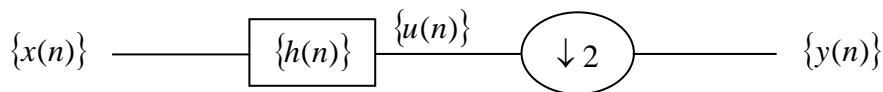
$$[DEC_{\downarrow 2}]x \leftrightarrow \frac{1}{2} \left[X(z^{\frac{1}{2}}) + X(-z^{\frac{1}{2}}) \right]. \quad (17)$$

Convolution puis sous-échantillonnage :

Mathématiquement, l'expression d'un tel opérateur est :

$$y(n) = \{h(n) * x(n)\}_{\downarrow 2} \quad (18)$$

Le schéma suivant représente le processus de convolution suivi d'un sous-échantillonnage :



On obtient aussi :

$$u(n) = \sum_k x(k)h(n-k) \quad (19)$$

Puis après sous-échantillonnage par deux :

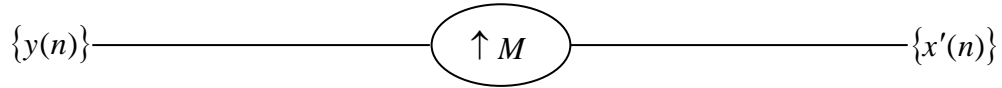
$$y(n) = u(2n) = \sum_k x(k)h(2n-k) \quad (19)$$

3.2 Sur-échantillonnage :

Le sur-échantillonnage de données signifie l'ajout de points entre ces données pour augmenter l'échantillonnage. Supposons que $\{y(n)\}$ soit l'entrée d'un sur-échantillonneur. L'augmentation du nombre d'échantillons par M , implique une insertion de $M - 1$ zéros entre les points de la séquence $\{y(n)\}$:

$$x'(n) = \begin{cases} y\left(\frac{n}{M}\right) & \text{pour } n = kM \\ 0 & \text{d'ailleurs} \end{cases} \quad (20)$$

Le schéma suivant décrit le système d'un sur-échantillonnage par M :



Dans la suite seul un sur-échantillonnage par $M = 2$ sera utilisé. En terme de matrice, l'équation (20) s'écrit :

$$[x'] = [INT_{\uparrow 2}] [y] \quad (20)$$

ou

$$\begin{bmatrix} \cdot \\ \cdot \\ x'(-2) \\ x'(-1) \\ x'(0) \\ x'(1) \\ x'(2) \\ x'(3) \\ \cdot \\ \cdot \\ \cdot \end{bmatrix} = \begin{bmatrix} \cdot & 0 & 1 & 0 & \cdot \\ \cdot & 0 & 0 & 0 & \cdot \\ \cdot & 0 & 1 & 0 & \cdot \\ \cdot & 0 & 0 & 0 & \cdot \\ \cdot & 0 & 1 & 0 & \cdot \\ \cdot & 0 & 0 & 0 & \cdot \\ \cdot & 0 & 1 & 0 & \cdot \\ \cdot & 0 & 0 & 0 & \cdot \\ \cdot & 0 & 1 & 0 & \cdot \\ \cdot & 0 & 0 & 0 & \cdot \\ \cdot & 0 & 1 & 0 & \cdot \end{bmatrix} \begin{bmatrix} \cdot \\ \cdot \\ y(-2) \\ y(-1) \\ y(0) \\ y(1) \\ y(2) \\ y(3) \\ y(4) \\ \cdot \\ \cdot \end{bmatrix} \quad (21)$$

Dans le domaine de Fourier la version de (20) est donnée par :

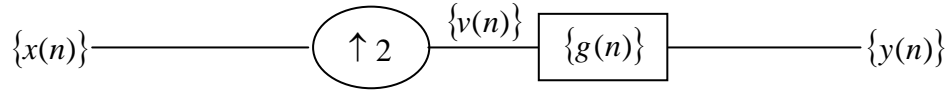
$$[INT_{\downarrow 2}] [y] \leftrightarrow Y(z^2).$$

Sur-échantillonnage puis convolution :

Mathématiquement, l'expression d'un tel opérateur est :

$$y(n) = \{g(n) * [x(n)]_{\uparrow 2}\} \quad (22)$$

Le schéma suivant représente le processus d'un sur-échantillonnage suivi d'une convolution :



On obtient ainsi :

$$y(n) = \sum_k v(k)g(n-k) \quad (23)$$

Puis après sur-échantillonnage par deux, avec $v(k) = x(k/2)$ pour les k pairs :

$$\begin{aligned} y(n) &= \sum_{k:\text{pair}} x(k/2)g(n-k) \\ &= \sum_p x(p)g(n-2p) \end{aligned} \quad (24)$$

4. Analyse multirésolution :

L'analyse multirésolution a été introduite par [MAL 89]. Ainsi l'analyse multirésolution (MRA) est utilisée communément avec les ondelette dans l'analyse des signaux (images). La théorie mathématique de l'analyse multirésolution permet d'analyser à la fois des niveaux grossiers (peu riches en détails ou basses fréquences) et des niveaux fins (riches en détails ou hautes fréquences) d'un signal, c'est-à-dire la décomposition du signal en gros approximation et détail (différence) qui sont mutuellement orthogonal. Cette orthogonalité assure la non redondance dans la représentation du signal d'origine. Le signal approximé obtenu est décomposé ultérieurement en gros approximation et détail et ainsi de suit.

4.1 Espace d'approximation :

La MRA de l'espace $L^2(\mathbb{R})$ des fonctions continues d'une variable réelle et de carré intégrable est basée sur des sous-espace V_j , emboîtées les un dans les autres, tels que le passage de l'un à l'autre soit le résultat d'un changement dyadique de facteur d'échelle ce qui correspond à un zoom.

$$f(x) \in V_j \Leftrightarrow f(2x) \in V_{j+1} \quad (25)$$

On a de même, de façon plus générale les propriétés suivantes :

$$\dots \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \dots \quad (26)$$

$$\bigcup_{j \in \mathbb{Z}} V_j = L^2(\mathbb{R}) \quad (27)$$

$$\bigcap_{j \in \mathbb{Z}} V_j = \{0\} \quad (28)$$

Il existe une fonction ϕ dit d'échelle tels que $\{\phi(t-k)\}_{k \in \mathbb{Z}}$ constitue une base de V_0 . La fonction d'échelle a aussi comme propriété de pouvoir engendrer une base orthonormée de V_j par translation et dilatation :

$\phi_{j,n}(t) = \frac{1}{\sqrt{2^{-j}}} \phi(2^j t - n), j, n \in \mathbb{Z}$. Les fonctions $\phi_{j,n}$ constituent les fonctions de la

base de l'espace V_j , qui est orthonormé si $\int_{-\infty}^{+\infty} \phi(t) \phi^*(t-n) dt = \delta(n)$, avec $n \in \mathbb{Z}$.

On pourra projeter n'importe quelle fonction f de $L^2(\mathbb{R})$ sur la base ainsi créée.

En utilisant la notation $\langle f|g \rangle$ pour designer le produit scalaire $\int_{-\infty}^{+\infty} f(t) g^*(t) dt$, et

$P = \sum |\phi_{j,n}\rangle \langle \phi_{j,n}|$ le projecteur sur la base de $\phi_{j,n}$, on a :

$$P|f\rangle = \sum_n |\phi_{j,n}\rangle \langle \phi_{j,n} | f \rangle = \sum_n \langle \phi_{j,n} | f \rangle |\phi_{j,n}\rangle \quad (29)$$

On pose $a_n^j = \langle f | \phi_{j,n} \rangle$ l'approximation à la résolution j définie par la suite discrète des nombres (réels ou complexes) a_n^j .

4.2 Espace de détails :

L'espace de détail W_j (Fig. 8) est l'espace tel que :

$$V_{j+1} = V_j \oplus W_j \quad (30)$$

C'est l'espace contenant la différence d'information entre une approximation à la résolution j et une approximation à la résolution $j+1$. Comme W_{j+1} est orthogonal à V_{j+1} , alors W_{j+1} est orthogonal à W_j .

$$\forall j, k \neq j, \text{ Alors } W_j \perp W_k \quad (31)$$

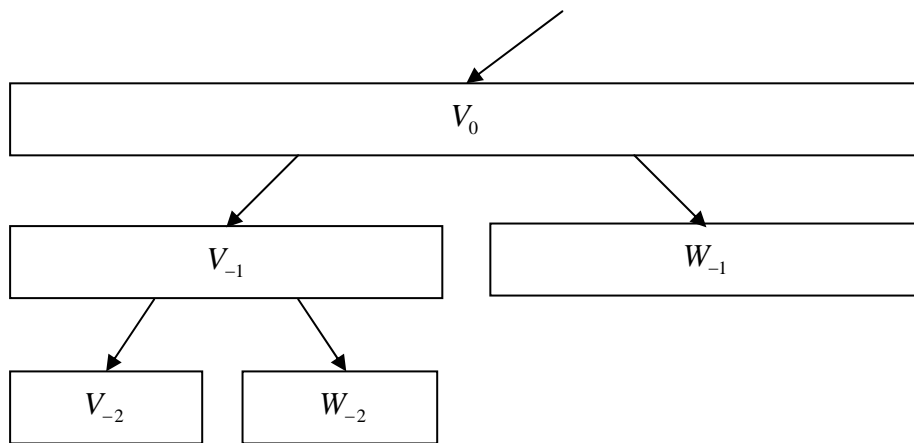


Fig. 8 : Schéma de l'analyse multirésolution

Les fonctions ondelettes, par dilatation et translation, engendrent une base orthonormée des W_j .

On va maintenant montrer comment les projections sur les différents sous-espaces vont finalement nous permettre de trouver les coefficients des filtres de l'implémentation en banc de filtres. Le point clé est la décomposition de $a_n^j = \langle f | \phi_{j,n} \rangle$ en fonction de $a_n^{j+1} = \langle f | \phi_{j+1,n} \rangle$.

Par construction $\phi(t)$ est une fonction de V_0 , comme $V_0 \subset V_1$ on peut décomposer $\phi(t)$ sur la base de V_1 .

Donc il existe $h[n]$ suite numérique avec $n \in Z$ telle que :

$$\phi(t) = \sum h[n] \phi_{1,n}(t) \text{ Avec } \phi_{1,n}(t) = 2^{\frac{1}{2}} \phi(2t - n) \text{ et } h[n] = \langle \phi | \phi_{1,n} \rangle \quad (32)$$

$h[n]$ sera considérée comme étant la réponse impulsionnelle d'un filtre

numérique H . Comme $\phi(t) = \sum_n h[n] 2^{\frac{1}{2}} \phi(2t - n)$ alors :

$$\phi_{j,n} = 2^{\frac{-j}{2}} \sum_k h[k] 2^{\frac{1}{2}} \phi(2(2^{-j}t - n) - k), \text{ d'où } \phi_{j,n} = \sum_k h[k] \phi_{j+1,k+2n}.$$

On peut calculer les coefficients $a_n^j = \langle f | \phi_{j,n} \rangle$ de l'approximation à la résolution j :

$$a_n^j = \sum_k h[k] \langle f | \phi_{j+1,k+2n} \rangle \quad (33)$$

En posant $l = k + 2n$, il vient :

$$a_n^j = \sum_l h[l - 2n] \langle f | \phi_{j+1,l} \rangle \quad (34)$$

Si maintenant on note $\tilde{h}[n] = h[-n]$ la séquence retournée (filtre symétrique) de H ,

$$a_n^j = \sum_l \tilde{h}[2n - l] \langle f | \phi_{j+1,l} \rangle \quad (35)$$

On obtient finalement l'équation réursive suivante :

$$a_n^j = \sum_l \tilde{h}[2n-l]a_n^{j+1} . \quad (36)$$

On considère a_n^j comme une séquence numérique indexée par n , alors le calcul précédent peut être vu comme un produit de convolution entre \tilde{h} et a^{j+1} évalué pour un indice sur deux, ou encore comme le filtrage de la séquence a^{j+1} par le filtre de réponse impulsionnelle \tilde{h} suivi par un sous-échantillonnage de rapport 2.

Un schéma analogue est bâti à partir de la décomposition de l'ondelette W_0 sur la base de V_{-1} :

$$\psi = \sum_n g[n]\phi_{1,n} \quad (37)$$

$$\psi(t) = \sum_n g[n]2^{\frac{1}{2}}\phi(2t-n) \quad (38)$$

Ce qui conduit à l'équation de construction de g :

$$g[n] = \langle \phi | \psi_{1,n} \rangle \quad (39)$$

$g[n]$ est également considérée comme étant la réponse impulsionnelle d'un filtre numérique. Un calcul semblable à celui précédemment effectuée pour les coefficients d'approximation a_n^j permet d'aboutir à l'expression des coefficients de détails en posant préalablement $\tilde{g}[n] = g[-n]$:

$$d_n^j = \sum_l \tilde{g}[2n-l]a_n^{j+1} \quad (40)$$

Le filtre $g[n]$ est lié à $h[n]$ est peut être construit à partir de ce dernier. La relation qui lie les deux filtres est de la forme :

$$\hat{g}(\omega) = -\exp(-i\omega)\hat{h}^*(\omega + \pi) \quad (41).$$

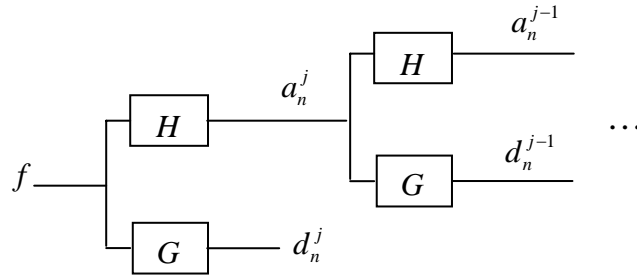


Fig. 9 : La relation entre la transformée en ondelette et les bancs de filtres

5. Conclusion :

Dans ce chapitre nous avons vu l'analyse de Fourier comme un outil plus ancien pour étudier un signal, ainsi son inconvénient par rapport la transformée de Fourier à fenêtre glissante (STFT), en fait, l'analyse de Fourier manque de localisation spatio-fréquentielle, par contre, dans la STFT le signal est découpé en différentes plages de longueur fixe pour offrir une information spatio-fréquentielle, mais la longueur fixe de la fenêtre de la fenêtre reste la source d'incertitude dans la localisation spatio-fréquentielle. Pour éviter ce problème la transformée en ondelette permet d'offrir une fenêtre de longueur variable et cela pour éviter l'incertitude et détecter les singularités du signal. Nous avons vu aussi que l'analyse multirésolution constitue une liaison entre la transformée en ondelette et les bancs de filtres pour faire les calculs numériques.

Chapitre III :

Extraction de l'index

1. Introduction :

Dans ce chapitre nous présentons deux techniques d'analyse d'images en vue de l'opération d'indexation : la transformée en quinconce [FEI 05] en se basant sur les bancs de filtres pour les manipulations numériques et la transformée en lifting [SWE 95][ZEE 02] en se basant sur le schéma en lifting. Nous proposons aussi à appliquer la technique de moments et invariants pour extraire les composants de l'index à partir des coefficients obtenus à l'issue du processus d'analyse d'images. A propos de la mesure de similarité, nous proposons à utiliser les distances géométriques, et enfin nous terminons par la description des résultats.

2. Les ondelettes multidimensionnelles :

Pour analyser une image de taille $N \times N$ en utilisant les ondelettes, on réfère à la base 2D. La décomposition en ondelettes à deux dimensions s'effectue de deux manières : la décomposition séparable et la décomposition non séparable. Dans le cas séparable la décomposition s'effectue d'abord en décomposant chaque ligne de la matrice en utilisant la décomposition en ondelettes à une dimension. Deux matrices rectangulaires de taille $N \times N/2$ sont alors obtenues. Ces deux matrices sont ensuite décomposées suivant leurs colonnes pour obtenir quatre matrices de taille $N/2 \times N/2$. Cette procédure peut être répétée jusqu'à une résolution finie. Dans le cas non séparable la décomposition s'effectue directement en deux dimensions sur l'intégralité de l'image.

3. Les ondelettes séparables versus les ondelettes non séparables :

Les ondelettes séparables sont les plus utilisées couramment dans l'analyse des images. Il y a deux causes principales. Le premier, est que la théorie des ondelettes a été développée dans 1D, ce qui est facilement extrapolable dans 2D par le produit tensoriel des fonctions de base. Le deuxième réside dans l'économie de calcul, ainsi que la transformée séparable peut être implémentée par un traitement 1D successif des lignes et des colonnes de l'image. Néanmoins, l'inconvénient, est que la transformée séparable privilège les directions verticale, horizontale et diagonale et néglige les autres directions de l'image et cela, constitue une contradiction avec la nature de l'image dans laquelle l'information s'évolue dans toutes les directions.

Par contre les ondelettes non-séparables sont convenables à utiliser pour analyser toute l'intégralité de l'image et non pas les lignes puis les colonnes. Son inconvénient, est qu'elles demandent spatialement plus de calculs, mais, cet inconvénient est évitable si on fait les calculs dans le domaine fréquentiel

en utilisant la transformée de Fourier rapide (FFT). Ainsi que pour la réduction multirésolutionnelle on utilise l'échantillonnage en quinconce. Il y a deux causes. Le premier, est que l'échantillonnage en quinconce est presque isotrope, surtout en utilisant la transformée de McClellan [MCC 73] pour trouver le cas 2D de l'ondelette. Le deuxième, en comparant avec les ondelettes séparables il y a une seule ondelette et la réduction multirésolutionnelle est plus progressive, $\sqrt{2}$ au lieu 2.

4. Echantillonnage en quinconce :

Dans ce paragraphe nous donnons les primitives de l'échantillonnage en quinconce (Fig. 1).

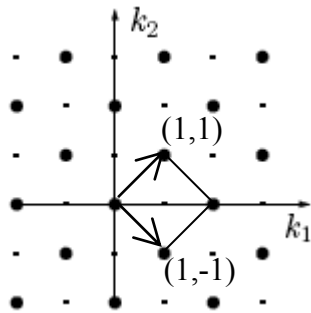


Fig. 1 : l'échantillonnage en quinconce dans le domaine spatial.

Soit $x[\vec{k}]$, $\vec{k} = (k_1, k_2) \in Z^2$ une image numérique. La transformée de Fourier 2D de $x[\vec{k}]$ est donnée par : $X(\vec{\omega}) = \sum_{\vec{k} \in Z^2} x[\vec{k}] e^{-j\langle \vec{\omega} | \vec{k} \rangle}$ avec $\vec{\omega} = (\omega_1, \omega_2)$. Alors son sous-échantillonnage en quinconce est donnée par :

$$[x]_{\downarrow D}[\vec{k}] = x[D\vec{k}] \quad , \quad \text{avec} \quad D = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \quad (1)$$

Nous remarquons que $|\det D| = 2$, ce qui indique la réduction de la taille de l'image est de 2 (non séparable) au lieu 4 dans le cas séparable. Dans le domaine de Fourier (1.1) s'écrit alors :

$$[X]_{\downarrow D}[\vec{\omega}] = \frac{1}{2} [X(D^{-T}\vec{\omega}) + X(D^{-T}\vec{\omega} + \vec{\pi})], \text{ avec } \vec{\pi} = (\pi, \pi).$$

Le sur-échantillonnage est défini par :

$$[x]_{\uparrow D}[\vec{k}] = \begin{cases} x[D^{-1}\vec{k}] & \text{si } k_1 + k_2 \text{ pair} \\ 0 & \text{d'ailleurs} \end{cases},$$

son version dans le domaine de Fourier est donnée par :

$$[X]_{\uparrow D}[\vec{\omega}] = X(D^T\vec{\omega}). \quad (2)$$

d'après (1) et (2), on peut déduire une formule d'un sous-échantillonnage suivi d'un sur-échantillonnage comme le suivant :

$$[x]_{\downarrow D \uparrow D}[\vec{k}] = \begin{cases} x[\vec{k}] & \text{si } k_1 + k_2 \text{ pair} \\ 0 & \text{d'ailleurs} \end{cases},$$

ce qui va donner dans le domaine de Fourier :

$$[X]_{\downarrow D \uparrow D}[\vec{\omega}] = \frac{1}{2} [X(\vec{\omega}) + X(\vec{\omega} + \vec{\pi})] \quad (3)$$

Tant que l'échantillonnage en quinconce réduit la taille de l'image par un facteur de 2, alors le banc de filtres associé a deux canaux (Fig. 2). Le filtre passe-bas \tilde{H} réduit la résolution par un facteur de $\sqrt{2}$. Les coefficients de l'ondelette correspondent à la sortie du filtre passe-haut \tilde{G} .

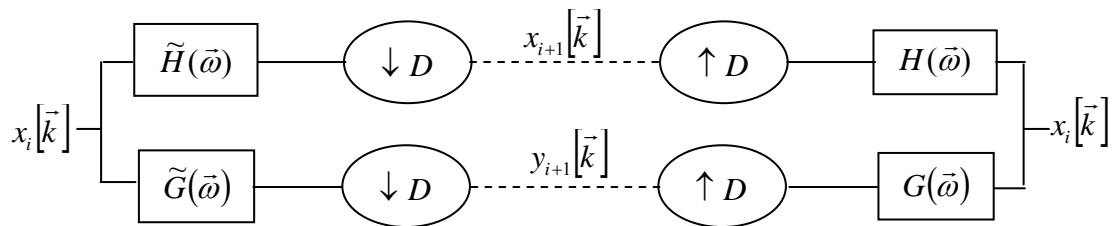


Fig. 2 : La reconstruction parfaite du banc de filtre avec l'échantillonnage en quinconce.

En appliquant la formule (3) au diagramme de la Fig.2, on peut conclure la condition de la reconstruction parfaite :

$$\begin{cases} \tilde{H}(\bar{\omega})H(\bar{\omega}) + \tilde{G}(\bar{\omega})G(\bar{\omega}) = 2 \\ \tilde{H}(\bar{\omega} + \bar{\pi})H(\bar{\omega}) + \tilde{G}(\bar{\omega} + \bar{\pi})G(\bar{\omega}) = 0 \end{cases} \quad (4)$$

où H et G (respectivement \tilde{H} et \tilde{G}) sont les fonctions de transfert des filtres de synthèse (respectivement d'analyse). Ces filtre correspondent au bases de décomposition et reconstruction des ondelettes. La reconstruction parfaite il nous indique qu'il n'y a pas une perte d'informations au cours des itérations d'analyse. Dans notre cas l'analyse sera orthogonal c'est-à-dire les filtres d'analyse sont les même de synthèse.

5. La transformée en quinconce:

Dans notre implémentation, on a choisi une nouvelle famille de filtres orthogonaux [FEI 05] qui ont été d'abord construits en une seule dimension sous la forme :

$$\begin{aligned} H_{\alpha}(z) &= \frac{\sqrt{2}(z + 2 + z^{-1})^{\frac{\alpha}{2}}}{\sqrt{(z + 2 + z^{-1})^{\alpha} + (-z + 2 - z^{-1})^{\alpha}}} \\ &= \frac{\sqrt{2}(2 + 2\cos\omega)^{\frac{\alpha}{2}}}{\sqrt{(2 + 2\cos\omega)^{\alpha} + (2 - 2\cos\omega)^{\alpha}}} \end{aligned} \quad (5)$$

Pour extrapoler en 2D, on applique la transformée de diamond McClellan qui remplace $\cos\omega$ par $(\cos\omega_1 + \cos\omega_2)/2$. Ainsi, le filtre fractionnaire en quinconce devient :

$$H_{\alpha}(\omega_1, \omega_2) = \frac{\sqrt{2}(2 + \cos\omega_1 + \cos\omega_2)^{\frac{\alpha}{2}}}{\sqrt{(2 + \cos\omega_1 + \cos\omega_2)^{\alpha} + (2 - \cos\omega_1 - \cos\omega_2)^{\alpha}}} \quad (6)$$

Le filtre de la transformée orthogonale en ondelettes $G_\alpha(\omega_1, \omega_2)$ est obtenu par modulation c'est-à-dire :

$$G_\alpha(\omega_1, \omega_2) = e^{j\omega_1} H_\alpha(-\omega_1 - \pi, -\omega_2 - \pi). \quad (7)$$

La fonction d'échelle correspondante $\varphi_\alpha(\vec{x})$ est définie par la résolution de l'équation :

$$\varphi_\alpha(\vec{x}) = \sqrt{2} \sum_{\vec{k} \in \mathbb{Z}^2} h_\alpha[\vec{k}] \varphi_\alpha(D\vec{x} - \vec{k})$$

où $h_\alpha[\vec{k}]$ est la version spatial du filtre $H_\alpha[\vec{\omega}]$.

L'ondelette en quinconce sous-jacent est donnée par :

$$\psi_\alpha(\vec{x}) = \sqrt{2} \sum_{\vec{k} \in \mathbb{Z}^2} g_\alpha[\vec{k}] \varphi_\alpha(D\vec{x} - \vec{k})$$

où $g_\alpha[\vec{k}]$ est la version spatial du filtre $G_\alpha[\vec{\omega}]$.

L'avantage de cette approche, est que les filtres pour les petites valeurs de α sont presque isotropes (Fig. 3), c'est-à-dire, les coefficients de la transformée en ondelettes obtenus n'ont aucune orientation préférentielle, et cela reflète la qualité de l'analyse.

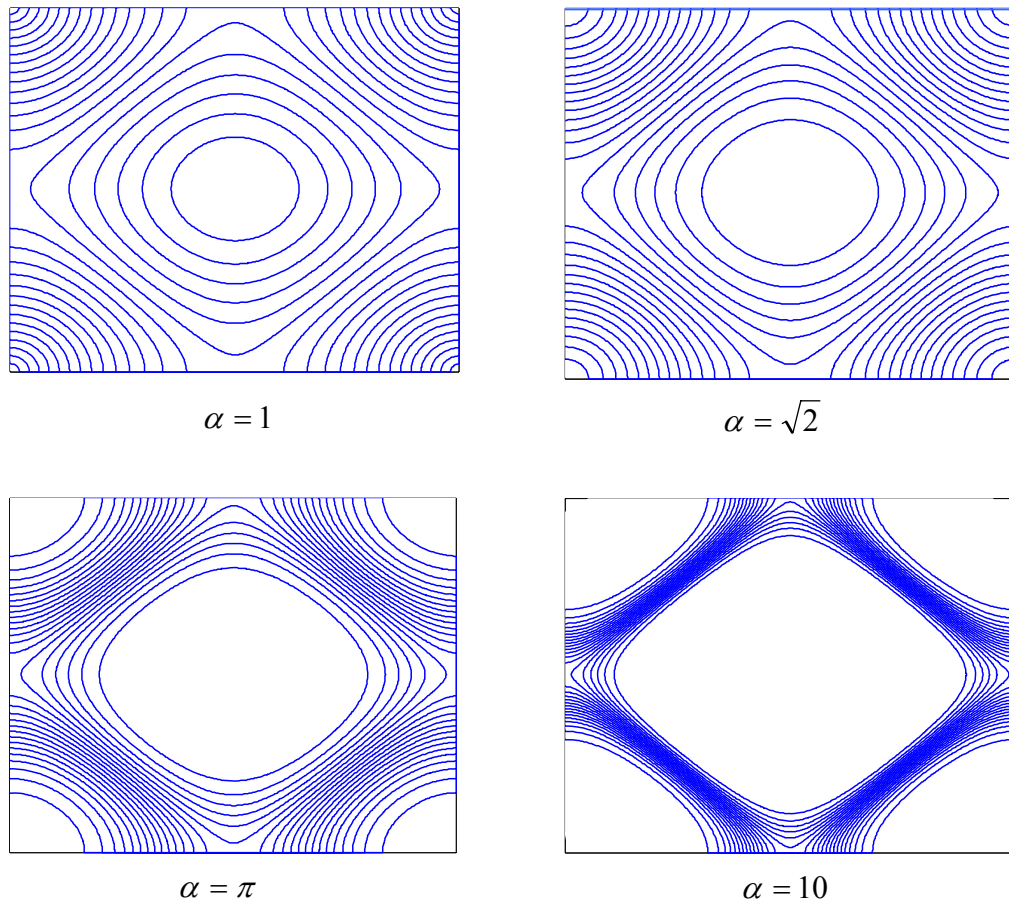


Fig. 3 : L'isotropie du filtre $H_\alpha(\omega_1, \omega_2)$ pour différentes valeurs de l'ordre α

5.1. Implémentation dans le domaine de Fourier :

Pour notre implémentation, nous avons utilisé l'algorithme de Feilner [FEI 05] illustrée sur la figure 1.3. Cet algorithme est implémenté dans le domaine de Fourier pour gagner le temps de calcul, ainsi que la convolution par la réponse impulsionnelle des filtres dans le domaine spatial devient une simple multiplication avec la réponse fréquentielle de ces dernier dans le domaine fréquentiel. Le passage du domaine spatial au domaine fréquentiel et vice versa s'effectue par la transformée de Fourier rapide (FFT) dont la complexité de

calculé est de $O(\frac{1}{2}N \log_2 N)$ au lieu $O(N^2)$ pour la transformée de Fourier normale [ALM 96]

D'après le schéma block de la figure 4, la première étape consiste à calculer la FFT d'une image entrante $x[\vec{k}]$, supposons que cette image est de taille $N \times N$, et de calculer la réponse fréquentielle des filtres d'analyse correspondants $\tilde{H}[\vec{n}]$ et $\tilde{G}[\vec{n}]$ en utilisant (6) et (7). La FFT de l'image $x[\vec{k}]$ est donnée par :

$$X_i[\vec{n}] = \sum_{\vec{k}} x_i[\vec{k}] e^{-j \frac{2\pi \langle \vec{k} | \vec{n} \rangle}{N}}, \quad \vec{n} = \begin{pmatrix} n_1 \\ n_2 \end{pmatrix} \text{ telle que } n_1, n_2 = 0 \dots N-1.$$

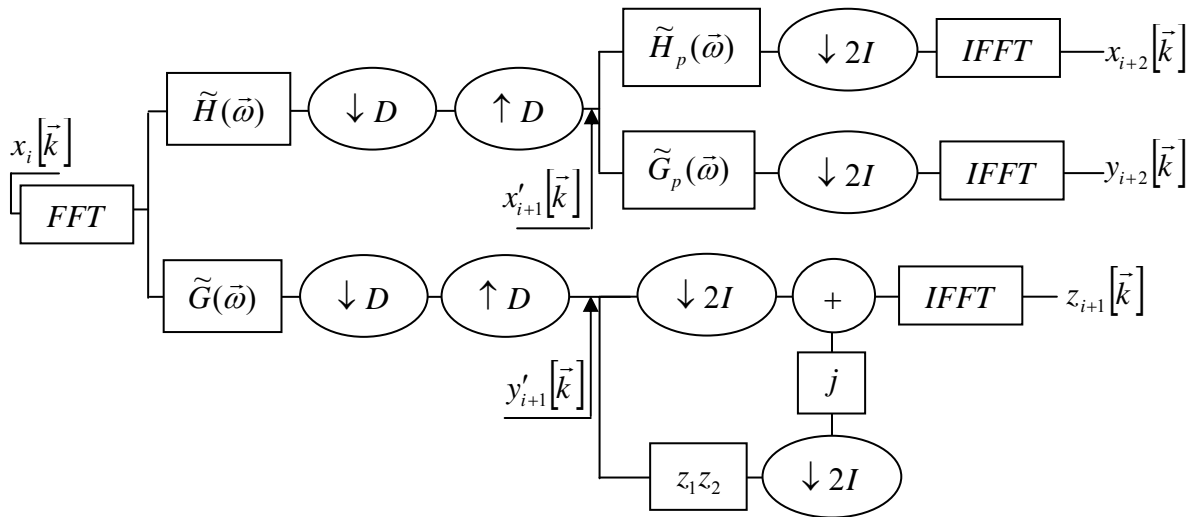


Fig. 4 : Schéma illustratif de l'algorithme de transformée en quinconce.

Pour les itérations impaires la FFT des coefficients de la transformée en quinconce est donnée par :

$$X'_{i+1}[\vec{n}] = \sum_{\vec{k}} x'_{i+1}[\vec{k}] e^{-j \frac{2\pi \langle \vec{k} | \vec{n} \rangle}{N}}, \quad n_1, n_2 \in [0, N-1]^2. \quad (8)$$

$$Y'_{i+1}[\vec{n}] = \sum_{\vec{k}} y'_{i+1}[\vec{k}] e^{-j \frac{2\pi \langle \vec{k} | \vec{n} \rangle}{N}}$$

Le sous-échantillonnage suivi par le sur-échantillonnage, permet d'introduire des zéros dans le domaine spatial et de garder la taille de l'image. Mais il introduit des redondances dans le domaine fréquentiel. Seulement le demi des coefficients qui sera calculé. Cela va donner :

$$\begin{aligned} Y'_{i+1}[m_1, n_2] &= \frac{1}{2} \left(\tilde{G}[m_1, n_2] X_i[m_1, n_2] + \tilde{G} \left[m_1 + \left(\frac{N}{2} \right), n_2 + \left(\frac{N}{2} \right) \right] X_i \left[m_1 + \left(\frac{N}{2} \right), n_2 + \left(\frac{N}{2} \right) \right] \right) \\ X'_{i+1}[m_1, n_2] &= \frac{1}{2} \left(\tilde{H}[m_1, n_2] X_i[m_1, n_2] + \tilde{H} \left[m_1 + \left(\frac{N}{2} \right), n_2 + \left(\frac{N}{2} \right) \right] X_i \left[m_1 + \left(\frac{N}{2} \right), n_2 + \left(\frac{N}{2} \right) \right] \right) \end{aligned} \quad (9)$$

où, $m_1 \in \left[0, \frac{N}{2} - 1 \right], n_2 \in [0, N - 1]$.

La variable $z_{i+1}[\vec{k}]$ est utilisée comme un artifice pour gagner le temps de calcul dans le calcul de la transformée de Fourier inverse *IFFT*. Ainsi que l'image se sépare en deux sous images, l'une contient les lignes paires et l'autre contient les lignes impaires. Les images paire et impaire constituent les parties réel et imaginaire successivement d'une variable $Z[\vec{k}]$ qui est l'objet de la *IFFT* comme suivant :

$$\begin{aligned} Z[\vec{m}] &= Y'_{i+1}[\vec{m}] + Y'_{i+1} \left[\vec{m} + \left(0, \frac{N}{2} \right) \right] + j(Y'_{i+1}[\vec{m}] - Y'_{i+1} \left[\vec{m} + \left(0, \frac{N}{2} \right) \right]) e^{j \frac{2\pi(m_1+m_2)}{N}} \\ z_{i+1}[\vec{k}] &= y_{i+1, \text{pair}}[\vec{k}] + jy_{i+1, \text{impair}}[\vec{k}] \end{aligned} \quad (10)$$

où $z_{i+1}[\vec{k}]$ constitue la *IFFT* de la variable $Z_{i+1}[\vec{m}]$. Le coefficient $y_{i+1}[\vec{k}]$ est obtenue par : $y_{i+1}[k_1, 2k_2 + 1] = \text{Re}\{z_{i+1}[\vec{k}]\}$, et $y_{i+1}[k_1, 2k_2] = \text{Im}\{z_{i+1}[\vec{k}]\}$.

Pour les itérations paires, les filtres $\tilde{G}_p[\vec{m}]$ et $\tilde{H}_p[\vec{m}]$ sont obtenus par l'échantillonnage par D de la version précédente des filtres impairs $\tilde{G}[\vec{n}]$ et $\tilde{H}[\vec{n}]$, cela va donner :

$$\tilde{H}_p[\vec{m}] = \tilde{H}[D\vec{m} \bmod(N, N)] \text{ et } \tilde{G}_p[\vec{m}] = \tilde{G}[D\vec{m} \bmod(N, N)]$$

La transformée de Fourier de la sortie des itérations paires est donnée par :

$$Y_{i+2}[\vec{m}] = \sum_{\vec{k}} y_{i+2}[\vec{k}] e^{-j \frac{2\pi \langle \vec{k} | \vec{m} \rangle}{\frac{N}{2}}}, \quad X_{i+2}[\vec{m}] = \sum_{\vec{k}} x_{i+2}[\vec{k}] e^{-j \frac{2\pi \langle \vec{k} | \vec{m} \rangle}{\frac{N}{2}}} \quad \text{Pour } \vec{m} = \begin{pmatrix} m_1 \\ m_2 \end{pmatrix} \text{ et}$$

$$m_1, m_2 = 0, \dots, \frac{N}{2} - 1.$$

et les opérations de filtrage sont données par :

$$\begin{aligned} X_{i+2}[\vec{m}] &= \frac{1}{2} \left(\tilde{H}_p[\vec{m}] X'_{i+1}[\vec{m}] + \tilde{H}_p \left[\vec{m} + \left(0, \frac{N}{2}\right) \right] X'_{i+1} \left[\vec{m} + \left(0, \frac{N}{2}\right) \right] \right) \\ Y_{i+2}[\vec{m}] &= \frac{1}{2} \left(\tilde{G}_p[\vec{m}] X'_{i+1}[\vec{m}] + \tilde{G}_p \left[\vec{m} + \left(0, \frac{N}{2}\right) \right] X'_{i+1} \left[\vec{m} + \left(0, \frac{N}{2}\right) \right] \right) \end{aligned} \quad (11)$$

L'algorithme itère jusqu'à une résolution plus fine. A la fin nous obtenons $y_1[\vec{k}], y_2[\vec{k}], \dots, y_J[\vec{k}]$ comme coefficients de la transformée en quinconce. Pour l'obtention des coefficients pour les itérations paires, nous appliquons les mêmes artifices appliqués pour les itérations impaires.

La figure 5 illustre quatre exemples représentant les coefficients de détail de la transformée en quinconce avec $\alpha = 1$, $\alpha = \sqrt{2}$, $\alpha = \pi$ et $\alpha = 10$.



Image à analyser 256x256



$\alpha = 1$



$\alpha = \sqrt{2}$



$\alpha = \pi$



$\alpha = 10$

Fig. 5 : Résultats de la transformée en quinconce avec 6 itérations pour différents ordres α

6. La transformée en Lifting :

Le schéma en lifting [SWE 95][ZEE 02] c'est un algorithme qui permet de calculer la transformée en ondelette d'une manière efficace, et sans recours à la transformée de Fourier. Cet algorithme est caractérisé par son efficacité dans l'espace mémoire utilisé pour les coefficients de la transformée. Le schéma en lifting est constitué de deux phases l'une pour l'analyse ou la décomposition et l'autre pour la synthèse ou la reconstruction, et chacune des phases est constituée de deux étapes de prédiction et de mise a jour.

6.1 La décomposition :

La phase de décomposition est constituée de trois étapes : division (split), prédiction (predict) et mise a jour (update) voir figure 6.

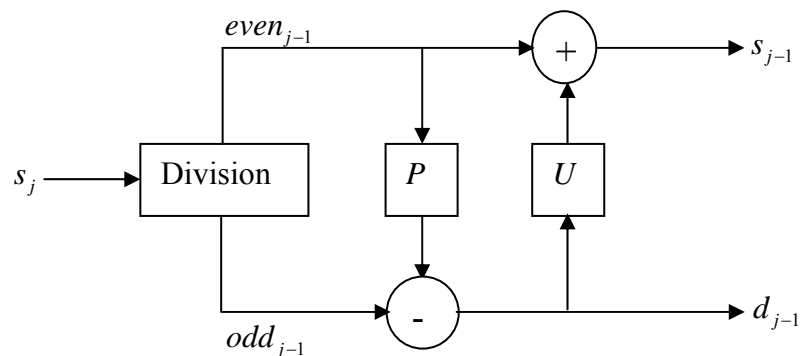


Figure 6: La phase de décomposition

- **Division (split) :** dans cette étape le signal est subdivisé en deux sous ensembles. Dans cette première définition nous considérons un sous ensemble contient les échantillons paires s_{2l} et l'autre contient les échantillons impaires s_{2l+1} .

Où on peut définir un opérateur *Split* tel que :

$$(even_{j-1}, odd_{j-1}) = Split(s_j)$$

- **Prédiction (predict)** : le but de cette étape est de prédire les échantillons impairs s_{2j+1} à partir des échantillons pairs s_{2j} . Soit le détail $d_{j-1,l}$ qui est la différence entre les échantillons impairs et ses prédictions :

$$d_{j-1,l} = s_{j,2l+1} - s_{j,2l}$$

Où on peut définir un opérateur P tel que :

$$d_{j-1,l} = odd_{j-1} - P(even_{j-1}).$$

- **Mise a jour (Update)** : l'un des propriétés du signal s_{j-1} et qu'il a la même valeur moyenne que le signal original, c'est-à-dire la quantité :

$$S = 2^{-j} \sum_{l=0}^{2^j-1} s_{j,l}$$

est indépendante du j . Alors cette étape permet de garantir cette valeur moyenne, en d'autre terme:

$$s_{j-1,l} = s_{j,2l} + d_{j-1,l}/2.$$

Où on peut définir un opérateur U sous la forme :

$$s_{j-1} = even_{j-1} + U(d_{j-1}).$$

Tous ces calculs peuvent être calculés dans la même place : les échantillons paires se replacent par le signal du prochaine itération et les échantillons impaires se replacent par les coefficients de détails. L'algorithme de décomposition peut être donné par :

$$\begin{aligned} (odd_{j-1}, even_{j-1}) &\leftarrow Split(s_j); \\ odd_{j-1} &\leftarrow odd_{j-1} - P(even_{j-1}); \\ even_{j-1} &\leftarrow even_{j-1} + U(odd_{j-1}); \end{aligned}$$

6.2. La reconstruction :

La phase de reconstruction constituée aussi de trois étapes : prédiction inverse (undo predict) , mise à jours inverse (undo update) et fusion (merge) voir Fig.7.

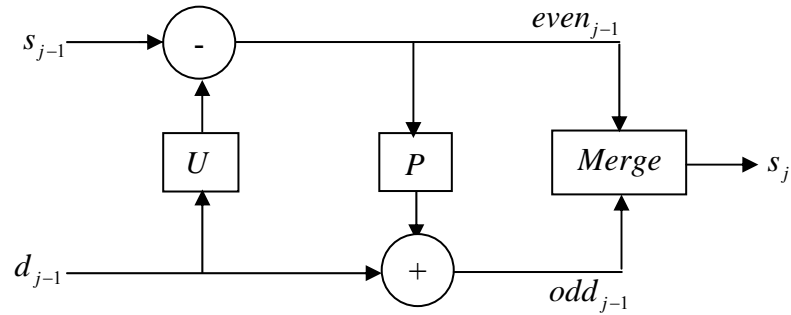


Figure 7 : La phase de reconstruction

- **Mise a jours inverse (undo update) :** Si nous avons d_{j-1} et s_{j-1} , on peut récupérer les échantillons pairs par la soustraction des informations de mise à jour :

$$even_{j-1} = s_{j-1} - U(d_{j-1}).$$

- **Prédiction inverse (undo predict) :** si nous avons $even_{j-1}$ et d_{j-1} , on peut récupérer les échantillons impairs par l'ajout des informations de prédiction :

$$odd_{j-1} = d_{j-1} + P(even_{j-1}).$$

- **Fusion (merge) :** maintenant, nous avons les échantillons pairs et impairs, nous pouvons reconstruire le signal original en définissant l'opérateur de fusion *Merge* :

$$s_j = Merge(even_{j-1}, odd_{j-1}).$$

L'algorithme de reconstruction peut être donnée par :

$$\begin{aligned} \text{even}_{j-1} &\leftarrow \text{even}_{j-1} - U(\text{odd}_{j-1}); \\ \text{odd}_{j-1} &\leftarrow \text{odd}_{j-1} + P(\text{even}_{j-1}); \\ s_j &\leftarrow \text{Merge}(\text{odd}_{j-1}, \text{even}_{j-1}); \end{aligned}$$

7. Application de la transformée en lifting pour l'analyse de textures:

Nous considérons la texture d'une image comme un signal a deux dimensions. Nous subdivisons ce signal en deux sous ensembles définis en grilles en quinconce, voir figure 8. Cette division s'appelle aussi la division en rouge et noir. Les pixels en spots rouges sont utilisés pour prédire les pixels en spots noirs, tandis que la mise a jours des spots rouges sont effectués par l'utilisation des coefficients de détails des spots noirs.

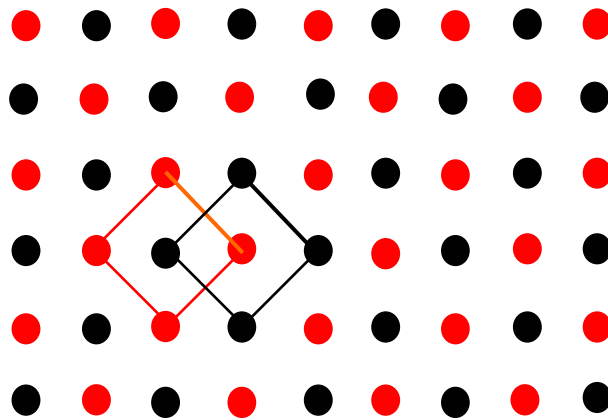


Figure 8 : Une grille rectangulaire composée de deux grilles en quinconce.

Exemple :

Un exemple de la transformée en lifting avec filtres de prédiction et de mise a jours d'ordre 2 est donné par :

$$\begin{aligned} (Px)(i, j) &= [x(i-1, j) + x(i, j-1) + x(i+1, j) + x(i, j+1)]/4, i \bmod 2 \neq j \bmod 2, \\ (Ux)(i, j) &= [x(i-1, j) + x(i, j-1) + x(i+1, j) + x(i, j+1)]/8, i \bmod 2 = j \bmod 2, \end{aligned}$$

L'algorithme en utilisant la grille en quinconce est aussi appelé la transformée en ondelette rouge et noir voir [GUY 97].

7.1. Filtre de Neville :

En général l'opérateur de prédiction P peut être s'écrit comme :

$$(Px)(i, j) = \sum_{(n,m) \in S_{\tilde{N}}} a_{\tilde{N}}(n, m)x(i+n, j+m), i \bmod 2 \neq j \bmod 2$$

où $S_{\tilde{N}} = \{(n, m) \in \mathbb{Z}^2, (n+m) \bmod 2 = 1\}$ $a_{\tilde{N}}(s), s \in S_{\tilde{N}}$ c'est un ensemble de coefficients dans R .

et L'opérateur de mise a jour en général peut être s'écrit comme :

$$(Ux)(i, j) = \sum_{(n,m) \in S_N} a_N(n, m)x(i+n, j+m)/2, i \bmod 2 = j \bmod 2,$$

Les coefficients $a_N(s)$ peuvent avoir les mêmes valeurs pour plusieurs éléments de S_N , alors ces derniers peuvent être classés en sous ensemble selon l'ordre N c'est-à-dire :

$$\begin{aligned} V_1 &= \{(\pm 1, 0), (0, \pm 1)\}, \\ V_2 &= \{(\pm 1, \pm 2), (\pm 2, \pm 1)\}, \\ V_3 &= \{(\pm 3, 0), (0, \pm 3)\}, \\ V_4 &= \{(\pm 2, \pm 3), (\pm 3, \pm 2)\}, \\ V_5 &= \{(\pm 1, \pm 4), (\pm 4, \pm 1)\}, \\ V_6 &= \{(\pm 5, 0), (0, \pm 5)\}, \\ V_7 &= \{(\pm 3, \pm 4), (\pm 4, \pm 3)\}. \end{aligned}$$

Les coefficients de filtre de Neville [KOV 00] $a_N(s), s \in V_k$, pour différents ordres N (2 à 8) sont illustrées dans la table 1. Pour illustrer le filtre de Neville, prenons $N = 4$ voir table 2, ici les valeur 1 et 2 correspondent aux valeurs de coefficients du filtre donnés par V_1 et V_2 respectivement. Dans le cas $N = 2$

montre tout simplement le cas de la transformée rouge et noir décrite précédemment voir figure 8.

N	V_1	V_2	V_3	V_4	V_5	V_6	V_7
2	1/4	0	0	0	0	0	0
4	10/32	-10/32	0	0	0	0	0
6	$87/2^8$	$-27/2^9$	2^{-8}	$3/2^9$	0	0	0
8	$5825/2^{14}$	$-2235/2^{15}$	$625/2^{16}$	$425/2^{15}$	$-75/2^{16}$	$9/2^{16}$	$-5/2^{12}$

Table1. Les coefficients du filtre de Neville.

	2		2	
2		1		2
	1	●	1	
2		1		2
	2		2	

		2		2		
2		1		1		2
			○			
2		1		1		2
		2		2		

Table2 : Filtre de Neville d'ordre 4 : rectangulaire (à gauche) et en quinconce (à droite)

La figure 9 illustre un exemple représentant un coefficient de détail de la transformée en lifting à l'itération 2 et en utilisant le filtre de Neville d'ordre 2.



(a)



(b)

Fig. 9 : Résultat de la transformée en lifting (a) l'image à analyser, (b) coefficient de détail à l'itération 2 en utilisant le filtre de Neville d'ordre 2

8. Moments et invariants :

Nous utilisons les invariants basés sur les moments d'ordre trois [HU 62]. A l'origine ces invariants ont été utilisés dans la reconnaissance des formes pour connaître la forme géométrique de différents objets. Les moments 2D m_{pq} d'ordre $p+q$ de la fonction $f(x,y)$ ($f(x,y)$ représente soit la texture d'une image,

ou dans notre cas représente les coefficients de détail de la transformée en ondelette) sont définis par :

$$m_{pq} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} x^p y^q f(x, y) dx dy, \quad (11)$$

les moments centrés μ_{pq} sont calculés par rapport au centre de gravité (x_g, y_g) de $f(x, y)$. Ils sont calculés par :

$$\mu_{pq} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} (x - x_g)^p (y - y_g)^q f(x, y) dx dy, \quad (12)$$

où $x_g = \frac{m_{10}}{m_{00}}$ et $y_g = \frac{m_{01}}{m_{00}}$, le dénominateur m_{00} représente la surface déterminée par la fonction $f(x, y)$.

Calculons le centre de gravité (x'_g, y'_g) de $g(x, y) = f(x - a, y - b)$, cela va donner $(x'_g, y'_g) = (x_g - a, y_g - b)$. A partir de (12) nous trouvons $\mu_{pq}(f) = \mu_{pq}(g)$, c'est-à-dire les moments centrés sont invariants par translation. Hu [HU 62] a trouvé les sept invariants d'ordre $N = p + q \leq 3$ donnés par :

$$\begin{aligned} I_1 &= \mu_{20} + \mu_{02} \\ I_2 &= (\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2, \\ I_3 &= (\mu_{30} - 3\mu_{12})^2 + (3\mu_{21} - \mu_{03})^2, \\ I_4 &= (\mu_{30} + \mu_{12})^2 + (\mu_{21} + \mu_{03})^2, \\ I_5 &= (\mu_{30} - 3\mu_{12})(\mu_{30} + \mu_{12})((\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2) + \\ &\quad (3\mu_{21} - \mu_{03})(\mu_{21} + \mu_{03})(3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2), \\ I_6 &= (\mu_{20} - \mu_{02})((\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2) + 4\mu_{11}(\mu_{30} + \mu_{12})(\mu_{21} + \mu_{03}), \\ I_7 &= (3\mu_{21} - \mu_{03})(\mu_{30} + \mu_{12})((\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2) - \\ &\quad (\mu_{30} - 3\mu_{12})(\mu_{21} + \mu_{03})(3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2). \end{aligned}$$

Dans notre travail, ces invariants sont extraits à partir des coefficients de détail de la transformée en ondelette. Le vecteur $I = (I_1, I_2, I_3, I_4, I_5, I_6, I_7)$ constitue une

portion du vecteur index finale. chaque itération de la transformée en ondelette elle dégagera un coefficient de détail sur lequel s'effectue le calcul des invariants ci-dessus.

9. La mesure de similarité :

La recherche au sein de la base d'images se base sur l'utilisation des index pré-calculés. Nous nous concentrons sur la recherche par similarité : l'utilisateur fait une requête sous la forme d'une image et les images de la base sont ordonnées en fonction d'un lois de mesure de similarité à cette requête. Considérons deux images Im_1 (image requête) et Im_2 (image cible) indexées par des vecteurs de R^N : $I_1 = \{I_1(i), 1 \leq i \leq N\}$ et $I_2 = \{I_2(i), 1 \leq i \leq N\}$. Le calcul de similarité entre Im_1 et Im_2 passe par une mesure de proximité entre les index I_1 et I_2 . Dans ce contexte, toute mesure entre vecteurs de R^N est utilisable, dans notre travail nous utilisons deux distances pour la recherche d'images par le contenu.

9.1. Distance géométrique :

Le premier type de mesure de similarité correspond aux distances géométriques entre vecteur. Dans ce cas, on parle de distances car ces mesures ont la propriété de respecter les axiomes des espaces métriques. Un espace métrique E se définit par Maurice Fréchet (1878-1973) comme un ensemble non vide doté d'une application d , appelée distance, de $E \times E$ dans R^+ vérifiant les axiomes suivants :

$$\forall x, y, z \in E$$

1. $d(x, y) = 0 \Leftrightarrow x = y$ (*Identité*)

2. $d(x, y) = d(y, x)$ (*Symétrie*)

3. $d(x, y) + d(y, z) \geq d(x, z)$ (*Inégalité triangulaire*)

Les métriques de Minkowski (ou normes L_p) sont les distances géométriques les plus courantes. Leur forme générale est la suivante :

$$d_{Mink}(I_1, I_2) = \left[\sum_{i=1}^N (I_1(i) - I_2(i))^p \right]^{\frac{1}{p}}$$

Deux types de métriques de Minkowski se retrouvent dans la littérature :

- Norme L_1 : $d_{L_1}(I_1, I_2) = \sum_{i=1}^N |I_1(i) - I_2(i)|$.
- Norme L_2 (distance euclidienne) : $d_{L_2}(I_1, I_2) = \left[\sum_{i=1}^N (I_1(i) - I_2(i))^2 \right]^{\frac{1}{2}}$, cette dernière qu'on l'adopte dans ce travail.

9.2. Distance de Mahalanobis :

La distance de Mahalanobis est défini par :

$$d_m(I_1, I_2) = \sqrt{(I_1 - I_2)^T C^{-1} (I_1 - I_2)}$$

Où C est la matrice de covariance défini par :

$$c_{ij} = \sum_{k=1}^m (f_{ki} - \mu_i)(f_{kj} - \mu_j) / (n-1), \quad i, j \in \{1, \dots, n\}$$

Où $F = \{f_{ij}\}$ contient tous les index de la base d'images (Fig. 10). Chaque ligne de F correspond a un index d'une image.

L'avantage de la distance de mahalanobis est qu'elle est invariante à tout changement non singulier avec une matrice $A \in R^n \times R^n$, c'est-à-dire, si nous substituons tout vecteur ligne x de F par Ax , alors la distance d_m reste non changée.

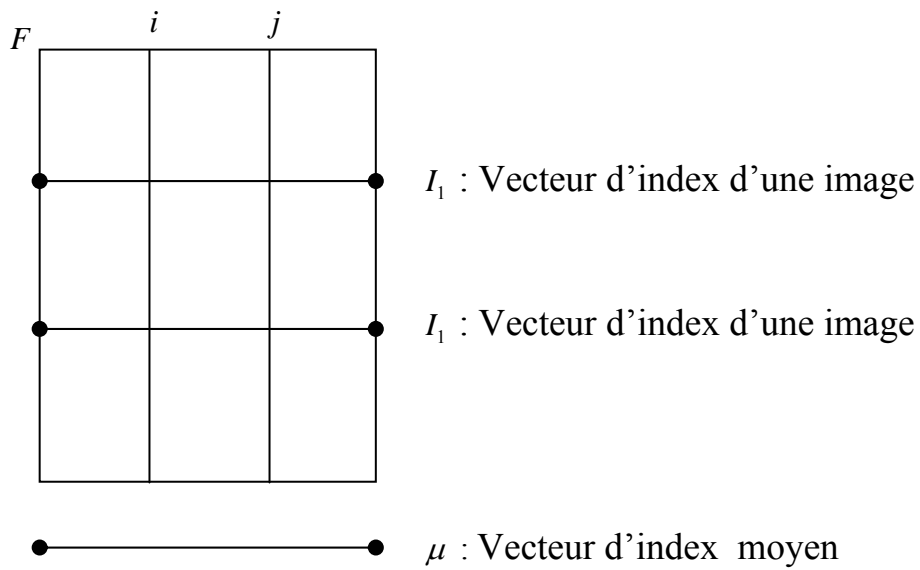


Fig. 10 : Matrice d'index de la base d'images

10. Expérimentation et résultats :

Dans notre travail nous avons choisi le Matlab comme outil de programmation, à cause de son efficacité de calcul, ainsi que Matlab se base sur le langage C pour son génération de code. Le Matlab a aussi l'avantage de générer l'interface utilisateur sans aucune complexité et ainsi permet de faciliter l'interaction avec le programme. A cause de sa simplicité syntaxique le Matlab augmente la productivité du programmeur.

Pour les testes nous avons choisi une base d'images généraliste contient 435 images de taille 256x256 de différents sujets. La recherche d'images par la mesure de similarité s'effectue par la comparaison des vecteurs index de taille 42. Sur les figures 11 et 12, on présente quelques résultats de recherche en prenant comme requête une image de couche de soleil. Dans les deux figures les images résultantes correspondent aux 20 images les plus similaires à la requête.

A partir des figure 11 et 12, nous constatons que l'ensemble d'images du résultat de recherche constituée de deux sous ensembles celle qui contient les

images pertinentes et l'autre contient les images non pertinentes vis-à-vis la sémantique ou la subjectivité. Citons d'abord qu'un système d'indexation d'images ne connaît pas la sémantique de l'image, et surtout dans le cas où la base d'images contient des images hétérogènes, mais ils se basent sur des caractéristiques de bas niveau (dans notre cas la texture se définit comme la distribution des niveaux de gris), alors le système il capte ou bien mesure la régularité entre les images ou en d'autres termes les images qui ont la même régularité sont considérées similaires par le système, mais il ne peut pas mesurer la sémantique qu'elle contient l'image ou la subjectivité de l'utilisateur. Le système il peut approcher à la subjectivité de l'utilisateur ou à la sémantique qu'elle contient l'image par ce qu'on appelle le bouclage de pertinence qui est l'objet d'une future recherche.

Sur la figure 15 nous constatons que les résultats de recherche sont presque les mêmes pour chaque méthode et pour différents filtres. Ici il y a deux facteurs qui peuvent influencer le résultat de recherche qui sont le type de la mesure de similarité choisi et l'ordre du filtre utilisé pour l'analyse de l'image. Pour l'approche de la transformée en quinconce (voir figure 15.b et 15.d) nous constatons que les ordres 1.41 et 3.14 correspondent aux meilleures isotropies du filtre ont donné relativement les meilleurs résultats. Pour l'approche de la transformée en lifting (voir figure 15.a et 15.c) nous pouvons dire que les meilleurs résultats sont donnés en utilisant les filtres de Neville d'ordre 6 et 8 et cela peut être interprété par le bon voisinage de ces filtres.

Sur les figures 13 et 14, nous illustrons les résultats de la recherche pour des requêtes médicales. Nous constatons que les résultats sont merveilleux pour les deux approches, et cela peut être interprété par la spécificité du domaine.

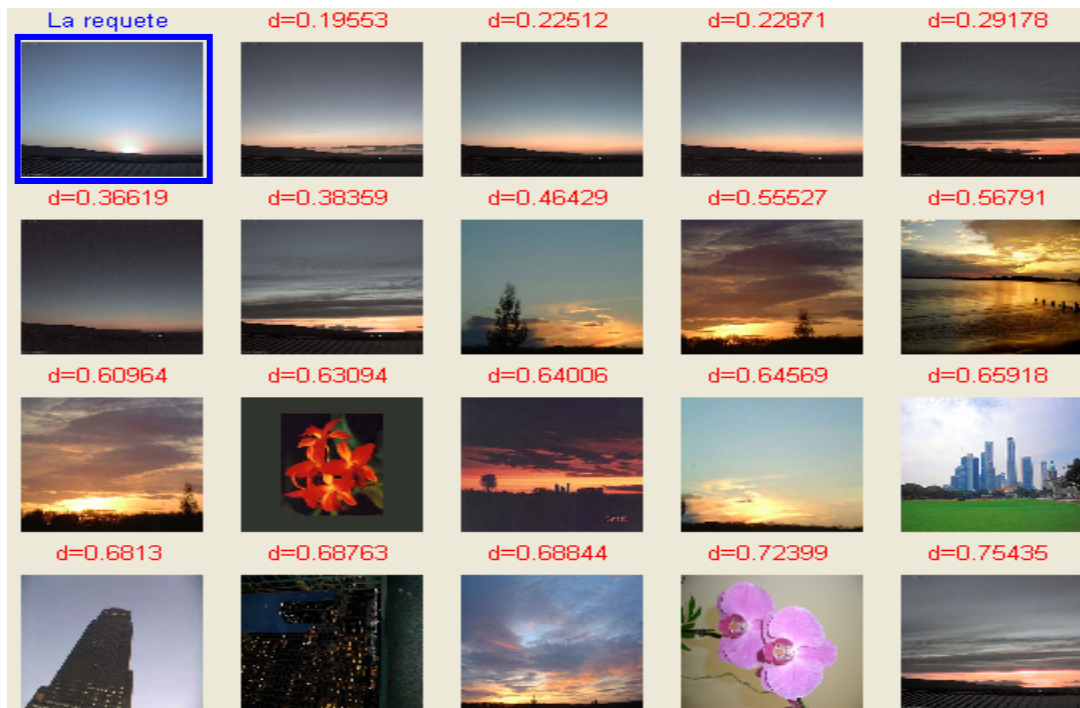


Fig. 11 : Résultats de recherche en utilisant le schéma en lifting avec filtre de Neville d'ordre 2 et la distance de mahalanobis.

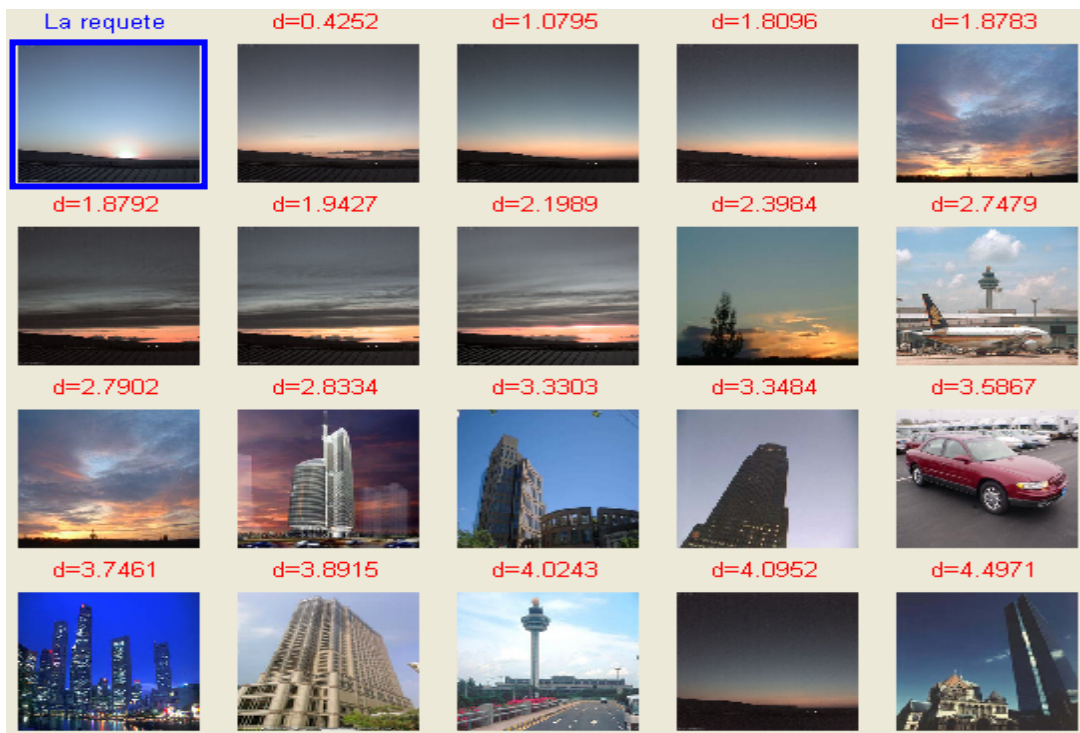
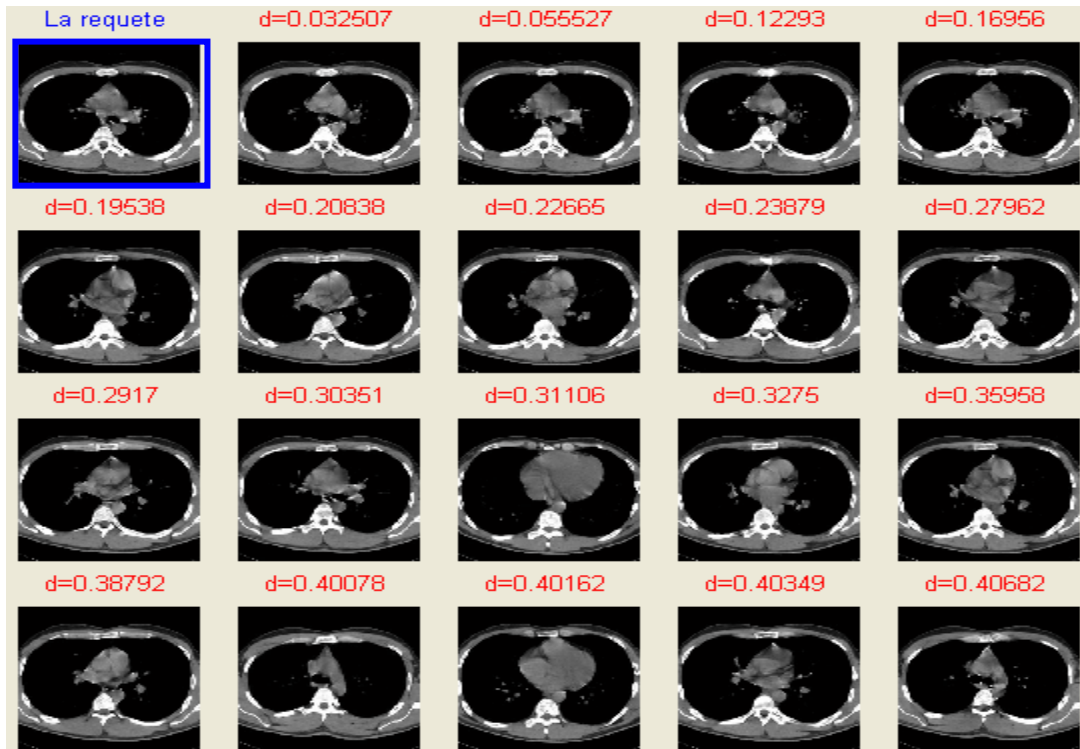
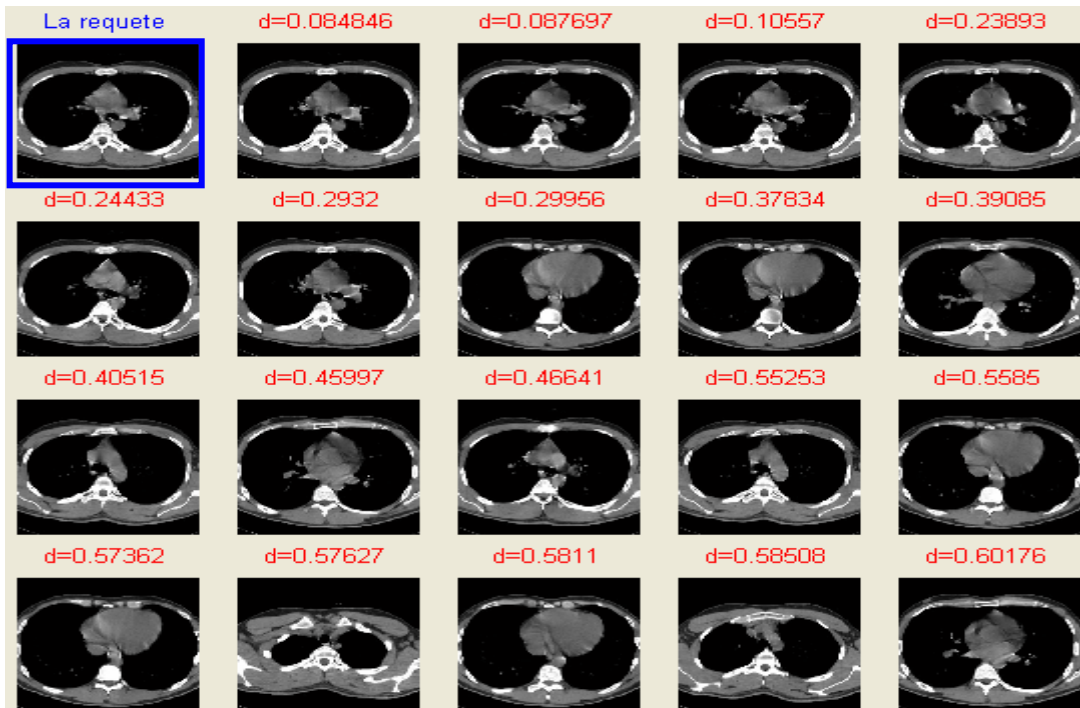


Fig. 12 : résultats de recherche en utilisant le filtre de M.Feilner d'ordre π et avec la distance de mahalanobis.

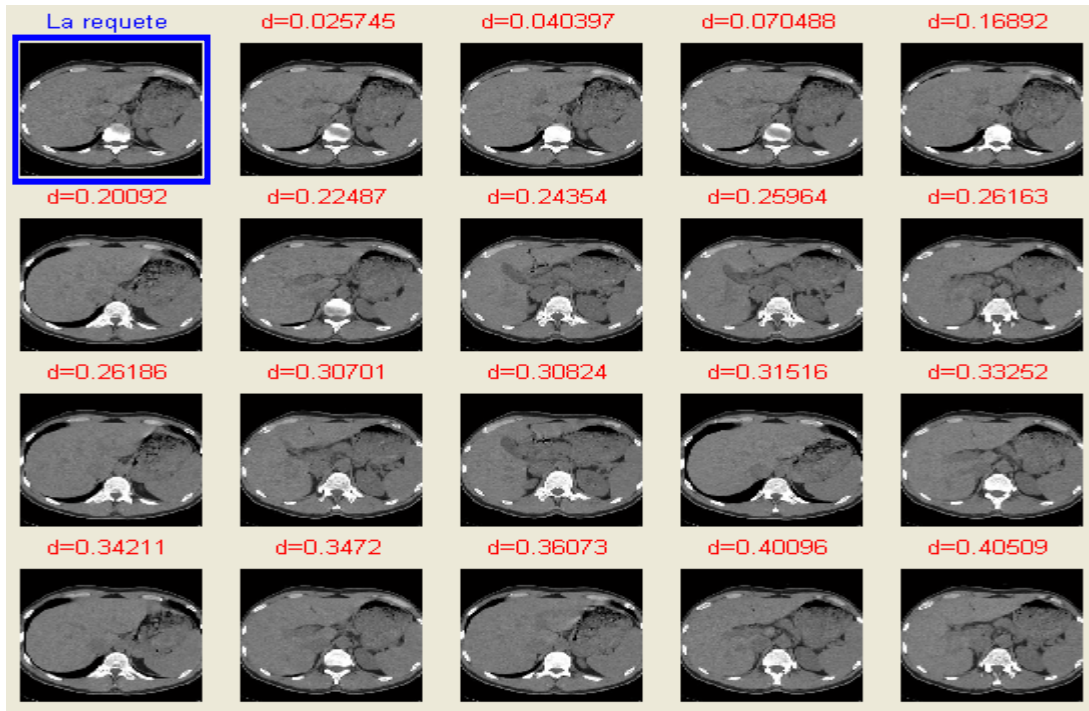


(a)

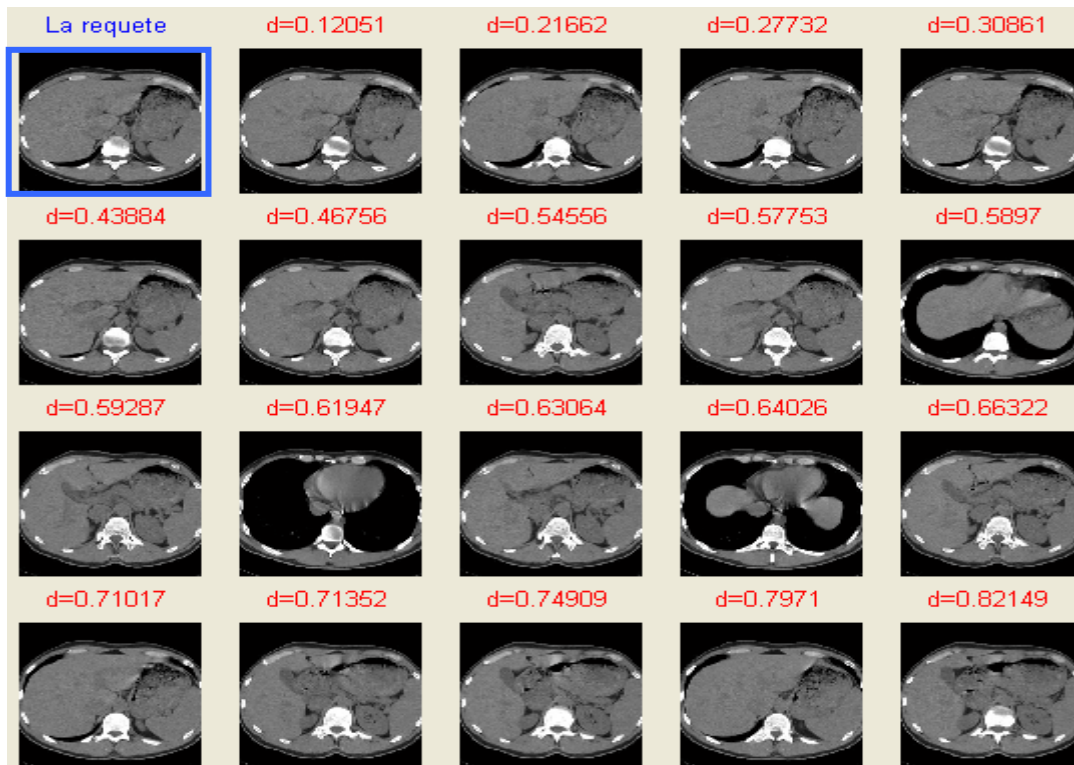


(b)

Fig.13 : Résultats de recherche pour les images médicales (a) avec la transformée en quinconce en utilisant le filtre d'ordre10, (b) avec la transformée en lifting en utilisant le filtre de Neville d'ordre8

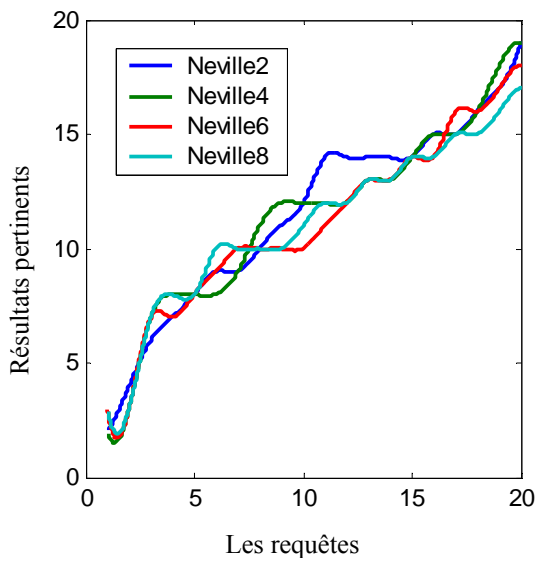


(a)

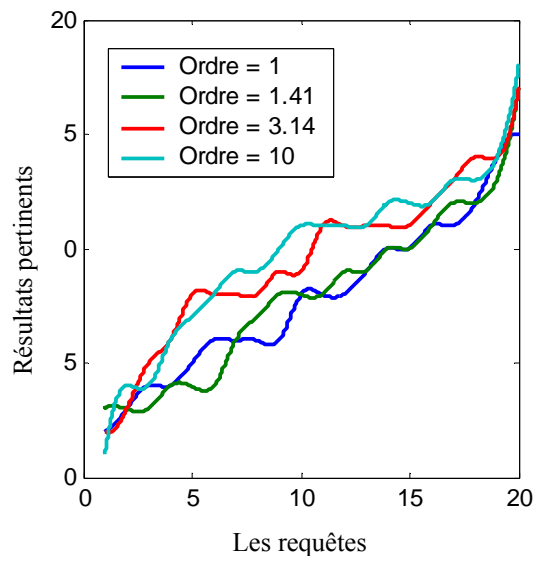


(b)

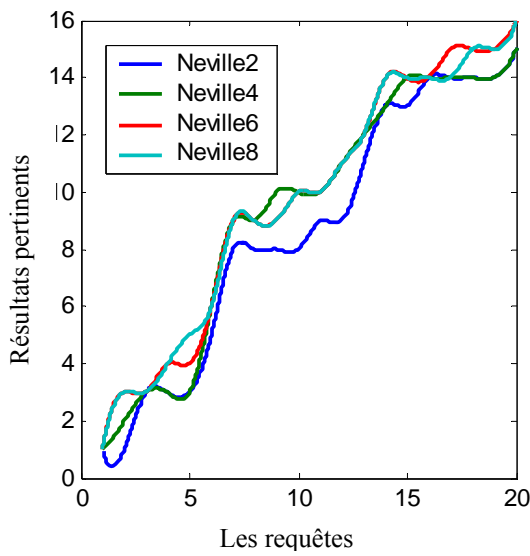
Fig.14 : Résultats de recherche pour les images médicales (a) avec la transformée en quinconce en utilisant le filtre d'ordre10, (b) avec la transformée en lifting en utilisant le filtre de Neville d'ordre8



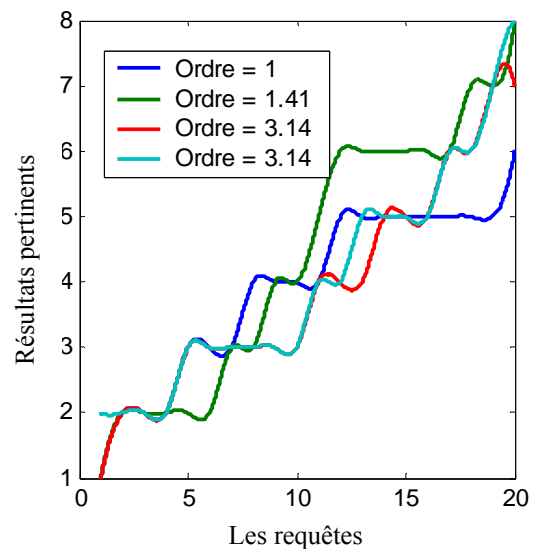
(a)



(b)



(c)



(d)

Fig. 15 : Résultats de la recherche d'images pour les deux méthodes, en utilisant la transformée en quinconce (à gauche) et la transformée en lifting (à droite). Sur (a) et (b) en utilisant la distance euclidienne et sur (c) et (d) en utilisant la distance de mahalanobis.

Sur la figure 16 nous illustrons une comparaisons des performances entre la transformée en quinconce et la transformée en lifting de la part d'indexation, et la distance euclidienne et la distance de Mahalanobis de la part de la mesure de similarité ou de la recherche. Nous constatons sur la figure 16.a que les résultats obtenus avec la distance de Mahalanobis sont mieux qu'avec la distance euclidienne et cela peut être interprété par la stabilité de la distance de Mahalanobis par apport au distance euclidienne.

Sur la figure 16.b nous constatons pratiquement que les résultats pertinents obtenus avec la transformée en lifting sont mieux que ceux obtenus avec la transformée en quinconce.

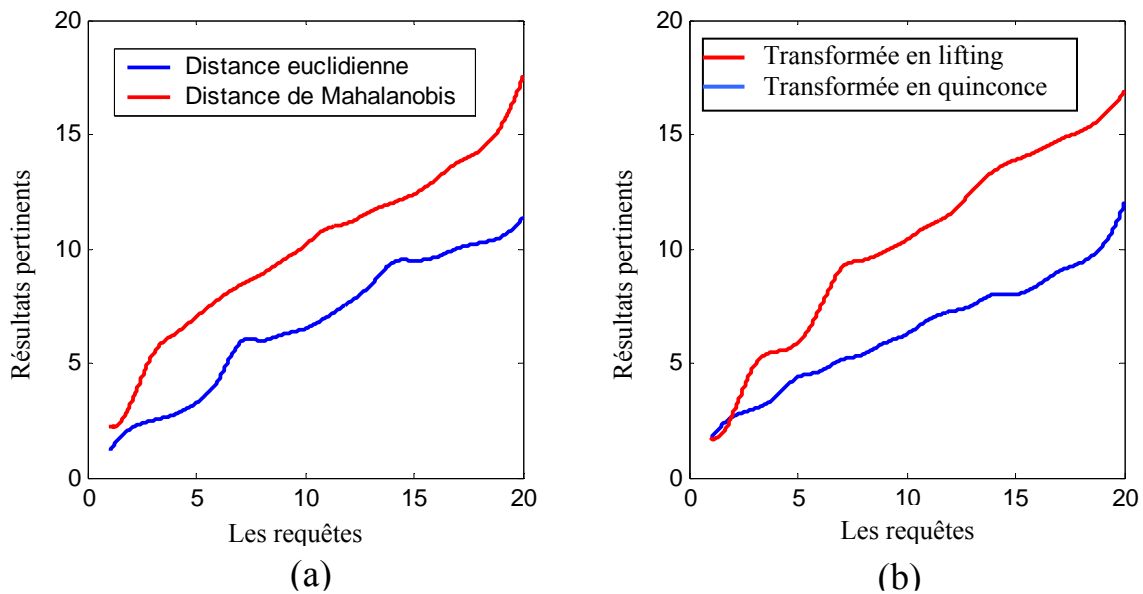


Fig. 16 : Comparaison des résultats de recherche (a) entre distance euclidienne et distance de mahalanobis, (b)entre la transformée en lifting et la transformée en quinconce

11. Conclusion

Dans ce chapitre, nous avons décrit deux algorithmes d'analyse d'images : la transformée en lifting et la transformée en quinconce. L'avantage de la transformée en lifting vient du fait que les calculs s'effectuent dans la même place c'est-à-dire un gain d'espace mémoire, en d'autre part, la réduction multirésolutionnelle (échantillonnage) vient avant les opérations du filtrage. Son inconvénient, la convolution par les réponses impulsionnelles des filtres augmente la complexité temporelle. L'avantage de la transformée en quinconce, vient du fait que les calculs s'effectuent dans le domaine fréquentiel, et cela permet d'éviter les coût de calcul, ainsi une convolution devient une simple multiplication. Son inconvénient, la réduction multirésolutionnelle vient après les opérations du filtrage, c'est-à-dire qu'il y a un effort de calcul de données qui seront tronquées après. Nous avons vu aussi les moments et invariants pour extraire les composants d'index d'une image, ainsi que une comparaison entre la distance euclidienne et de Mahalanobis pour la mesure de similarité.

Chapitre IV :

CBIR Distribué

1. Introduction :

Nous allons voir dans ce chapitre les environnements distribués sur lesquels pourra s'appuyer un CBIR distribué. Nous voyons que le langage Java est le langage adéquat pour tout traitement distribué, et précisément pour un CBIR distribué, alors, dans ce travail on est restreint aux environnements qui font partie intégrante de ce langage. Ainsi, une comparaison entre les différentes infrastructures distribués du langage Java fait le noyau de ce chapitre. Pourquoi le langage Java? Premièrement, à cause de son machine virtuelle JVM, java offre une indépendance matérielle et logicielle vis-à-vis le système sur lequel s'intègre, et cela évite les problèmes d'hétérogénéité des environnements distribués. Deuxièmement, à son paradigme d'objet, Java permet d'abstraire les environnements distribués. Pourquoi un CBIR distribué ? la distribution d'une base d'images centralisée à un ensemble de bases d'images distribuées réduit le nombre d'images par chacune, et cela réduit le temps de recherche d'une part, et d'autre part augmente la qualité des résultats.

2. CBIR distribué :

Nous considérons un CBIR distribué comme un réseau de services distribués connectés via un media de communication (Broker) [KNI 00], voir Figure 1. Il y a deux types de services : les services clients et les services serveurs. Le service client encapsule l'objet qui représente l'interface utilisateur par laquelle interagit l'utilisateur avec le système, tandis que le service serveur encapsule l'objet qui représente l'intermédiaire avec la base d'image dont le rôle est de permettre les opérations de la recherche dans la base d'image. Le service serveur informe le broker qu'il fournit un certain type de service. Le service client interroge le broker d'un certain type de service, s'il est offert, le broker connecte le service client avec le service serveur.

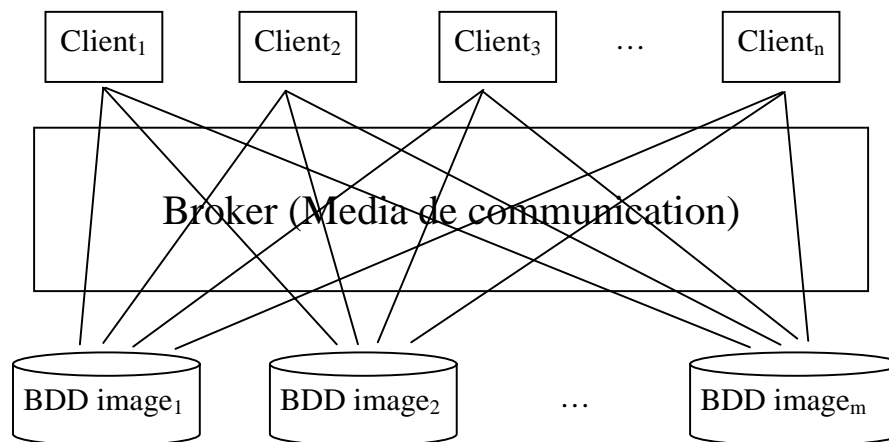


Figure 1 : CBIR Distribué

D'une autre manière, les services clients et serveurs peuvent être considérés comme un ensemble des objets distribués communiquent entre eux via le broker par l'invocation des méthodes à distance. Le broker est l'élément crucial dans un CBIR distribué, il constitue l'infrastructure sur laquelle s'effectuent tous les communications. Dans ce qui suit nous ferons

une description des infrastructures distribuées existantes, ainsi qu'une comparaisons entre eux.

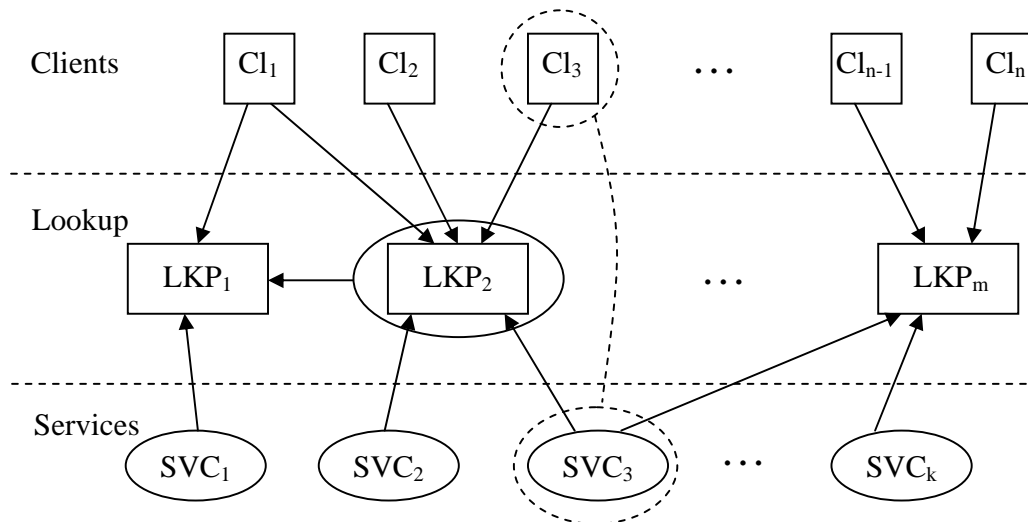


Figure 2 : Communauté Jini

3. L'architecture Jini:

Jini est une technologie développée et distribuée par Sun Microsystems. Elle se définit de plusieurs manière suivant l'utilisation que l'on souhaite en faire :

D'un point de vue général, Jini définit les concepts suivant : *un réseau* Jini appelé généralement communauté, est un ensemble d'entités ayant accès les unes aux autres, voir figure 2. Deux types d'entités existent dans une communauté Jini. La première est dite service. Cette entité est capable de réaliser une certaine tâche pour l'ensemble des autres entités membres de la communauté. L'autre type d'entité, est le client. Celui-ci utilise les services. Chaque communauté regroupe ses membres autour d'un service

centralisateur appelé lookup. Son rôle principale est la gestion de la découverte des services se trouvant dans son entourage. Sa participation comme entité de communication intermédiaire entre le client et le service, se limite au processus de découverte. Une fois établie, la communication s'effectue directement entre le client et le service. Un service n'est pas attaché de manière exclusive à un lookup particulier, il peut être répertorié dans plusieurs lookup en même temps. Le lookup étant un service à part entière, il est également capable de s'enregistrer dans un autre lookup gérant une autre communauté.

D'un point de vue technique, Jini offre une infrastructure logicielle, communément appelée middleware, permettant à des objets Java, appelés services, de se découvrir et de s'utiliser de façon spontanée. Jini exploite cependant la notion de code mobile fournie par Java et qui permet de transférer un objet java complet, données et code compris, d'une JVM(*Java Virtual Machine*) à une autre. Grâce à ce mécanisme, le service fournit dynamiquement au client un objet Java sérialisé capable de piloter le service à distance. Cet objet est appelé *Proxy*. Le service procure, d'une certaine façon, au client, le code que celui-ci ne possédait pas lors de sa création. Le client sait préalablement quel type de service il veut utiliser et comment l'utiliser mais il ne sait pas comment le proxy et le service réalisent cela. Jini est construit en Java, pour Java et exploite la technologie Java RMI (*Remote methode invocation*). Chaque entité désirant faire partie de la communauté Jini doit au moins posséder une partie de code écrite en Java.

Du point de vue du développeur, Jini crée un nouveau paradigme de programmation en proposant un modèle de développement d'application distribué fiables. Le modèle force le programmeur à prendre en compte les

différents problèmes inhérents au application réparties[WKE 01]. Ces problèmes sont retrouvés dans toute application séparée par un ou plusieurs réseaux : fiabilité, tolérance aux pannes, défaillance partielle,..

Du point de vue logiciel et pratique, Jini est développé en Java et se présente comme une extension logicielle pour la plate forme Java 2. Il étend ainsi la hiérarchie des packages standards offerts par Java.

Jini est une technologie qui a trouvé une certaine vocation pour les équipements électronique, qui peuvent s'insérer comme service dans une communauté Jini, et de ce point de vue, nous constatons q'un service représente aussi bien un équipement matériel qu'une application logicielle.

3.1. Fonctionnement du Jini :

3.1.1. Service lookup :

Le service lookup est chargé d'enregistrer les services se trouvant à sa proximité, d'entreposer leur proxy, de contrôler la disponibilité de ces services et enfin de répondre aux demandes des clients.

3.1.2. Protocole de découverte :

Le protocole de découverte est utilisé entre le service lookup et ses clients. Ce protocole en comprend trois praticables :

- Le «**unicast discovery protocol**» est utilisé quand un client connaît le port de service lookup qu'il cherche à joindre et l'adresse IP de la machine serveur sur laquelle se trouve ce lookup voir figure 3. L'objet proxy représentant le service lookup est alors transféré vers le client.

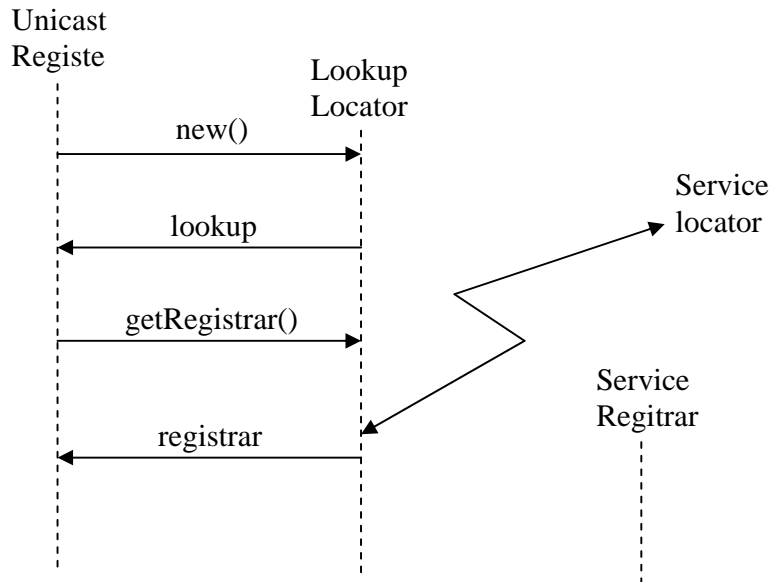


Figure 3 : Le diagramme de s quence UML illustrant Le « unicast discovery protocol »

- Le «**multicast discovery protocol**» est utilis  lorsqu'un client d sire conna tre tous les services lookup se trouvant dans ses alentours voir figure 4. Pour ce faire, le client envoie un paquet multicast UDP avec un petit time-to-live TTL sur un port pr -d fini. De cette mani re nous ne touchons qu'une environnement restreint. Tous les services lookup recevant ce paquet, r pondent en utilisant une version serveur de l'unicast discovery protocol.

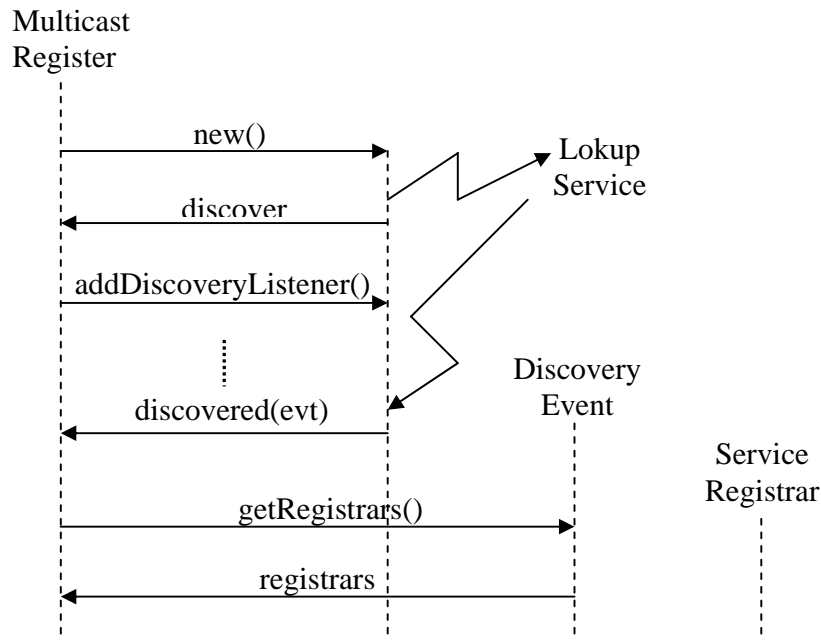


Figure 4 : Le diagramme de s quence UML illustrant Le « multicast discovery protocol »

- Le «**multicast announcement protocol** » est utilis  par le service lookup pour annoncer p riodiquement sa pr sence aux client. De cette fa on, les services sont directement mis au courant de son indisponibilit , suite   une panne mat rielle ou   une erreur logicielle par exemple. De m me lorsqu’il revient en ligne, les services peuvent   nouveau s’enregistrer aupr s de lui s’ils le d sirent. Chaque annonce de pr sence contient l’adresse IP et le port du lookup de mani re   ce que les clients int ress s, c’est- -dire les autres entit s Jini, puissent lui r pondre   l’aide de l’unicast discovery protocol.

3.1.3. Processus d'adhésion :

Lorsqu'un service reçoit le proxy du lookup sur lequel il désire être enregistré, il le fait grâce à la méthode (*registre*) de l'interface (*ServiceRegistre*). Cette interface est standardisée pour tous les proxies de lookup Jini. De cette manière le service client a la capacité de transmettre au lookup son propre proxy. Le service reçoit un numéro d'identification unique et un bail est créé. C'est le service lui-même qui décide des temps de renouvellement et de vie de ce bail. En général, le temps de vie du bail est infini. Le lookup conserve toujours le service. Par contre, le temps de renouvellement varie selon le type de service offert. Il restera alors au service à renouveler périodiquement son bail.

3.1.4. L'indépendance de protocole :

L'indépendance de la technique de communication mise au point entre le proxy et son service ainsi que sa capacité d'implémenter une part de l'intelligence du service du côté du client, donne à Jini une flexibilité qui le distingue de toutes les autres technologies visant à gérer des objets distribués dans un environnement évolutif spontané. Cette indépendance est donc un des principaux atouts de cette technologie et il s'agit de la préserver.

3.1.5. Les groupes :

Jini intègre une notion de groupe. Cette notion permet d'affiner les recherches de services et entraîne donc un meilleur classement de ceux-ci. Chaque client du lookup précise en s'enregistrant dans quel(s) groupe(s) il désire être répertorié. Chaque paquet réseau émis lors du processus de découvert comprend également le ou les groupe(s) visé(s). la notion de

groupe permet donc également de limiter le processus de découverte : seuls les lookups satisfaisant un des groupes demandés par le client répondra «présent » à l'appel. Parmi la multitude des groupes définissables, il y en existe un qui est proposé par défaut : le groupe «public ».

4. L'architecture RMI :

Le protocole RMI (Remote Method Invocation) est le protocole mis en place dans le langage Java pour les objets distribués. Il est livré avec tous les JDK(Java Développement Kit), et est donc fortement standardisé.

RMI permet de faire appel à des méthodes distantes, en connaissant à priori l'URL de celles-ci. RMI est donc proche de JINI, mais il manque la découverte dynamique de cet URL. Jini apporte cette découverte. Lorsque l'on trouve l'URL de la méthode à invoquer, il ne reste plus qu'à utiliser RMI pour l'invoquer.

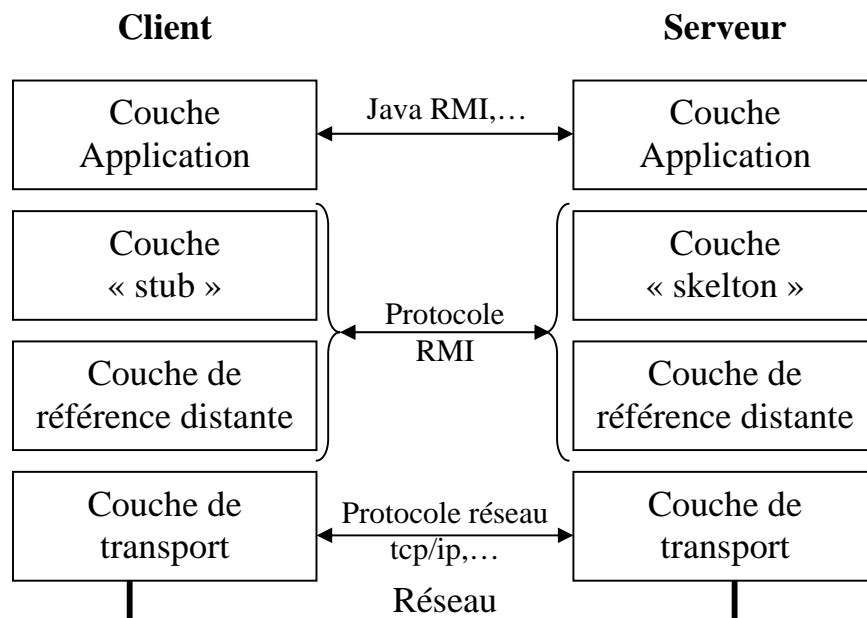


Figure 5 : Architecture RMI

Chaque objet distant nécessite l'utilisation d'un *stub* et d'un *skeleton* pour la sérialisation et la désérialisation des objets , voir Figure 5. Ces *stub* et *skeleton* sont des objets java générés automatiquement par un compilateur appelé *rmic*. Chaque objet distribué doit avoir un *stub* et un *skeleton*.

4.1. Les objets sérialisés :

RMI fait un grand usage des objets sérialisés de Java. Un objet sous la forme sérialisé est représenté par une suite d'octets. Cette suite représente les données nécessaires à la reconstruction de l'objet à partir de la définition de sa classe.

Pour désérialiser un objet, il faut donc deux éléments :la classe et son bytecode à laquelle il appartient et sa suite d'octets issus de la sérialisation. Le mécanisme de sérialisation prend en charge le stockage des attributs de l'objet. Si ces attributs sont eux-mêmes des objets, il sont à leur tour sérialisés.

4.2. Le code mobile :

Puisqu'il y a moyen de sérialiser un objet, et donc de l'envoyer quelque part à travers le réseau, il faut pouvoir être capable de le désérialiser à l'arrivée. Pour ce faire, il faut disposer du bytecode de la classe à laquelle appartient cet objet. RMI met en place un mécanisme très simple pour récupérer ce bytecode : il utilise un serveur http. Il est possible d'implémenter d'autres protocoles.

Lors du lancement d'un serveur RMI, il suffit de spécifier un « codebase » pour pouvoir accéder à la ressource voulue. Le « codebase » est un URL pointant vers le lieu de stockage des classes exportables.

4.3. Les interfaces et serveurs RMI :

Le RMI sert à invoquer des méthodes à distance. La définition des méthodes utilisables se fait par la définition d'une interface au sens Java du terme. Cette interface sera alors exportée par un serveur RMI.

Les serveurs RMI fonctionnent autour du « daemon » *rmid*. Ce programme se charge de créer un lien entre un nom d'URL et les interfaces exportées.

A ce programme, viennent s'ajouter les différents objets Java exportant des interfaces. Il existe plusieurs méthodes pour exporter des interfaces :

- **La méthode simple**, L'objet exportant l'interface est exécuté comme un autre programme. Il s'enregistre auprès de *rmid* et attend les requêtes des clients. L'avantage de cette méthode est d'avoir un serveur toujours prêt. Le désavantage est de monopoliser les ressources du système car il ne tourne pas en tâche de fond.
- **La méthode avec lancement de *rmid***, cette méthode est presque identique à la première. Elle ajoute cependant le lancement du « daemon » *rmid* au sein du serveur. L'avantage de cette méthode est de rendre l'utilisation d'RMI transparente à l'utilisateur, puisque celui-ci n'a alors plus à lancer « à la main » le « daemon » *rmid*.
- **Méthode des objets activables**, Cette troisième méthode est un peu plus compliquée à mettre en œuvre. Plutôt que de bloquer les ressources du système, après s'être enregistré auprès du « daemon » *rmid*, le serveur se met en veille et rend la main. C'est alors au

« dameon » *rmid* que revient la tache d'activer l'objet serveur lorsqu'il y a une demande émanant d'un client.

5. L'architecture CORBA:

CORBA (Common Object Request Broker Architecture) est une architecture logicielle réalisée par l'OMG (Object Management Group) dont le but est de spécifier et de standardiser un environnement permettant à des objets hétérogènes et distribués de communiquer entre eux ,voir figure 6. Ces objets sont dits distribués car nous pouvons y accéder à distance au travers d'un réseau. Il sont hétérogène car ils peuvent être issus de langages et de systèmes différents.

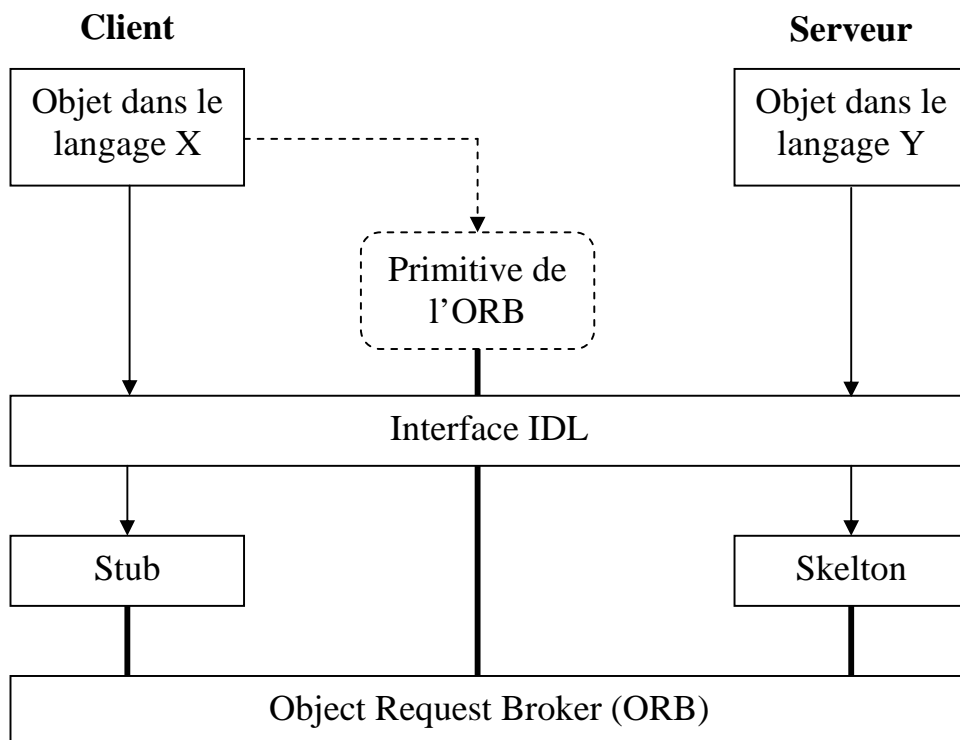


Figure 6 : L'architecture CORBA

5.1. ORB (Object Request Broker):

L'ORB , sorte de « bus logiciel », est l'infrastructure nécessaire à cette communication. Celle-ci est installée sur chaque poste désirant accueillir des objets CORBA. Elle joue le rôle d'intermédiaire entre chaque objet et s'occupe donc du contact (la localisation des objets dans l'environnement et de la transmission des informations entre les objets CORBA). Le « stub » et le « skeleton » évoqués précédemment viennent se placer entre les application et l'ORB. Remarquons que contrairement à RMI, un client doit impérativement posséder le « stub » de l'objet CORBA distant qu'il souhaite appeler.

5.2. Les Protocoles CORBA :

Pour l'échange d'information entre objets CORBA, l'OMG a défini un protocole standard de communication(GIOP : General Inter-Orb Protocol) dont IIOP (Internet IOP) est l'implantation au-dessus de TCP/IP. Remarquons que GIOP nécessite d'être implanté au-dessus d'un protocole de transport fiable, orienté connexion.

Chaque objet CORBA est référencé à l'aide d'une sorte de pointeur générique (IOR :Interoperable Object Reference) précisant le type de l'objet, sa localisation sur le réseau et son identité (clé) sur la machine « serveur » désignée. Ce pointeur permet donc d'identifier de manière unique un objet CORBA.

5.3. Langage IDL:

Pour permettre l'interopérabilité entre objets hétérogènes, l'OMG a défini un langage général de définition d'interface(IDL :Interface Definition

Language) permettant de spécifier de manière exhaustive les attributs et les méthodes que chaque objet CORBA offre aux autres. Ainsi un objet client peut appeler des méthodes d'un objet serveur d'une classe différente, sans se préoccuper du langage avec lequel elles sont implémentées. Bien sur, l'objet client doit connaître le nom et les prototypes des méthodes que l'objet servant met à disposition. L'IDL est un moyen indépendant du langage qui permet de préciser les types de données, les attributs, les opérations, les interfaces,... L'IDL est compilé par un compilateur IDL/Java propre au fournisseur, qui lit le source IDL et génère le code Java, et génère aussi le code qui sera utilisé pour réunir les argument des méthodes et pour réaliser les appels distants. Ce code appelé le code *stub* et le code *skeleton*.

6. Comparaison entre JINI, CORBA, RMI :

La relation cruciale entre JINI et RMI vient du fait que JINI est bâti sur RMI. Il hérite donc l'ensemble des caractéristiques de celui-ci. Mais n'est pas la situation inverse. L'avantage qu'il apporte JINI par rapport RMI, est qu'il exploite intensivement le mécanisme de téléchargement dynamique du code d'un objet pour fournir le proxy du service au client à travers le lookup. A son protocole de découverte, JINI permet la découverte et l'acquisition du proxy de services lookup qui, à leur tour, permettent de découvrir des objets Java (autres services) non plus sur base de leur nom comme le RMI, mais plutôt sur base de leur apparence.

Pour la communication entre les objets, CORBA utilise le protocole GIOP. Ce dernier est fixé d'avance et constitue l'un des aspects d'hétérogénéité de CORBA. JINI pour sa part recommande l'indépendance de protocole entre l'objet proxy et l'objet service. A cette indépendance JINI

peut implémenter le protocole GIOP pour communiquer avec un système CORBA, mais pas inversement. Les services CORBA sont exclusivement exécutés sur la machine hébergeant le service en question, et cela constitue une lenteur dans l'exécution. Par contre, JINI permet d'exécuter une partie ou la totalité du service sur la machine du client par l'intermédiaire du proxy, et cela constitue une rapidité dans l'exécution. Au niveau de la stabilité de la plate forme, CORBA est moins stable que JINI, ainsi lorsqu'une requête CORBA est échouée le système ne peut être précisé, par contre JINI génère des exceptions qui sont traitable par le développeur. Au niveau de l'interopérabilité, tout client d'un objet CORBA doit obligatoirement posséder le stub de celui-ci, sinon, il n'y a pas de communication réalisable. Cette exigence n'existe pas pour JINI et RMI car ils jouissent tous les deux du code mobile. Un stub d'un objet CORBA ne dialogue normalement qu'avec le skeleton de l'objet CORBA pour lequel il a été généré, par contre le proxy de JINI peut dialoguer avec tous les services qu'il désire. Le proxy, contrairement au stub, n'est pas généré mais programmé par le développeur du service.

7. Conclusion :

Dans ce chapitre, nous avons proposé une architecture technique d'un système d'indexation par le contenu distribué, en se basant sur l'architecture Jini. Nous avons aussi montré l'avantage de l'architecture Jini en comparant cette dernière avec d'autres architectures comme CORBA, RMI.

Conclusion

Notre travail rentre dans le cadre de la recherche d'images par le contenu (CBIR). Le premier chapitre se focalise sur les fondements d'un CBIR, le deuxième et le troisième donnent une introduction sur les ondelettes, ainsi que son application dans la phase d'indexation dans un CBIR, le quatrième, expose les différentes plate formes distribuées qui peuvent supporter un CBIR distribué.

L'indexation et la recherche d'images par le contenu sont des problèmes complexes et incontournables étant donnée la place que l'image numérique occupe à présent dans notre quotidien. Internet en est la meilleure illustration. Dans notre travail nous nous intéressons d'une part à l'indexation des images par le contenu et d'autre part à la recherche.

Du point de vue de l'indexation, nous avons proposé un traitement automatique pour le calcul des indexes des images. Ce traitement se base sur l'analyse de la texture, où nous portons une attention à l'utilité des ondelettes dans ce même cadre.

Du point de vue de recherche, nous avons utilisé le principe de la recherche par similarité selon lequel les images sont ordonnées à l'aide d'une mesure de similarité à la requête. Cette dernière prend la forme d'une image exemple.

Durant ce travail nous avons constaté deux points essentiels et cruciales. Le premier, est que la texture est une caractéristique discriminante d'une image, mais son seule utilisation dans un CBIR ne suffit que si les images sont d'un domaine spécifié, alors, une approche utilisant la texture

communément avec d'autres constituants primaires de l'image soit envisageable, cela peut augmenter l'efficacité d'un CBIR, mais elle ne peut pas aboutir la subjectivité de l'utilisateur, alors, que le deuxième, qu'il y a ce qu'on appelle le fossé sémantique, qui sépare les caractéristiques primaires extraites du contenu des images et la sémantique que l'utilisateur place dans sa requête, qui conduit à des résultats souvent erronés. L'utilisation du principe du bouclage de pertinence est l'alternatif pour réduire le fossé sémantique est ainsi améliorer les résultats de la recherche grâce aux annotations de l'utilisateur, et est constitué un future travail.

Bibliographie :

[**ALM 96**]. Alfred Mertins, signal analysis, wavelets, filter banks, time-frequency transform and applications, university of Wollongong, Australia.

[**BOU 00**]. Boujemaa (N.), Nastar (C.) et Malki (J.), Requêtes partielles sans segmentation pour la recherche d'images par le contenu, RFIA'2000, pp. 463-470, Paris, Février 2000.

[**CAR 99**]. Carson (C.), Thomas (M.), Belongie (S.), Hellerstein (J.M.) et Malik (J.).- Blobworld: A system for region-based image indexing and retrieval, Third Int. Conf. on Visual Information Systems, pp. 509-516.-June 1999.

[**COC 95**]. Cocquerez (J.) et Philipp (S.), Analyses d'images: filtrage et segmentation. Paris, Masson, 1995.

[**COX 00**]. Cox (I.J.), Miller (M.L.), Minka (T.P.), Papathomas (T.V.) et Yianilos (P.N.), The bayesian image retrieval system, PicHunter: Theory, implementation and psychophysical experiments. IEEE Transactions on Image Processing, vol. 9, n^o 1, 2000, pp. 20-37.

[**DER 99**]. Derrode (S.), Mezhoud (R.) et Ghorbel (F.), Reconnaissance de formes par invariants complets et convergents : Application à l'indexation de bases d'objets à niveaux de gris. GRETSI'99, Vannes, France, 13-17 Septembre 1999.

[**FEI 05**]. M. Feilner, M. Unser, An orthogonal family of quincunx wavelets with continuously adjustable order, IEEE Transaction on Image Processing. VOL. 14, N^o 4, APRIL 2005

[**FLI 95**]. Flickner (M.). Query by image and video content: the QBIC system. IEEE Computer, vol. 28, n^o9, 1995, pp. 23-32.

[**FOU 02**]. J. FOURNIER, Indexation d'images par le contenu et recherche interactive dans les bases généralistes, thèse de doctorat, 2002.

[**FRA 93**]. J. M. Francos. "Orthogonal decompositions of 2D random fields and their applications in 2D spectral estimation," N. K. Bose and C. R. Rao, editors, *Signal Processing and its Application*, pp.20-227. North Holland, 1993.

[**FUH 04**]. Dr. Fuhui Long, Dr. Hongjiang Zhang and Prof. David Dagan Feng, *Fundamentals of content-based image retrieval*.2004

[**GAB 46**]. D. Gabor, "Theory of communication", *J. Inst. Elect. Eng.*, 93, pp. 429-457, 1946

[**GUY 97**]. G. Uytterhoeven and A. Bultheel, "The red-black wavelet transform", TW Report 271, Dept. Comp.Sc, Katholieke Universiteit Leuven, Leuven, 1997.

[**HAR 88**]. Harris (C.) et Stephens (M.), A combined corner and edge detector. 4th Alvey Vision Conference, pp. 147-151. -1988.

[**HU 62**]. Hu (M.K.), Visual pattern recognition by moment invariants, *IRE Transactions on Information Theory*, vol. 8, n^o 2, 1962, pp. 179-187.

[**ITT 98**]. Itti (L.), Koch (C.) et Niebur (E.), A model of saliency-based visual attention for rapid scene analysis, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, n^o 11, 1998, pp. 1254-1259.

[**MCC 73**]. J. H. McClellan, "The design of two-dimensional digital filters by transformations," in *Proc. Annual Princeton Conf. Inf. Sciences and systems, Princeton(USA),1973,pp.247-251*.

[**KNI 00**]. Knighten, B2000, Peer-to-peer computing.
http://cnscenter.fe.co.kr/resource/hotopic/p2p/PtP_IDF_Rev_1.11-web.pdf

[**KOV 00**]. J. Kovacevic and W. Sweldens, Wavelet families of increasing order in arbitrary dimensions, *IEEE Trans. Imag. Pro.*, 9(3), 480–496, 2000.

[**Law 99**] Lawrence (Steve) et Giles (C. Lee).Accessibility of information on the web. *Nature*, vol. 400, 1999, pp. 107-109.

[**MAL 89**]. STEPHANE G. MALLAT, "A theory for multirésolution signal decomposition: the wavelet representation", *IEEE TRANSACTION ON*

PATTERN ANALYSIS AND MACHINE INTELLIGENCE, Vol 11. NO 7,
JULY 1989, pp. 674-693

[NAS 98]. Nastar (C.), Boujemaa (N.), Mitschke (M.) et Meilhac (C.).
SurImage : Un système flexible d'indexation et de recherche d'images.
CORESA'98. Juin 1998.

[OGI]. Jean-Marc OGIER, Indexation, Segmentation et Recherche par le
contenu, Laboratoire L3i Université de La Rochelle.

[PAS 96]. G. Pass, and R. Zabith, "Histogram refinement for content-based
image retrieval," IEEE Workshop on Applications of Computer Vision, pp.
96-102, 1996.

[PAU 99]. Pauwels (E.J.) et Frederix (G.), Finding salient regions in images
Non-parametric clustering for image segmentation and grouping. Journal of
Computer Vision and Image Understanding, vol. 75, n^o1-2, July-August
1999, pp. 73-85.

[PER 77]. Persoon (E.) et Fu (K.S.), Shape discrimination using fourier
descriptors, IEEE Trans. on Man and Cybernetics, vol. 7, n^o 3, 1977, pp.629-
639.

[ROU 99]. Roussey (C.), Calabretto (S.) et Pinon (J.M.),Etat de l'art en
indexation et recherche d'information. Document numérique, vol. 3, n^o3-4,
1999, pp. 121-149.

[RUI 96]. Rui (Y.), She (A.C.) et Huang (T.S.), Modified fourier descriptors
for shape representation , a pratical approach. Proc. of First International
Workshop on Image Databases and Multi Media Search, 1996.

[SAL 90]. Salton (G.) et Buckley (C.), Improving retrieval performance by
relevance feedback. Journal of the american society for information science,
vol. 41(4), 1990, pp. 288,297.

[SCH 95]. Schmid (C.) et Mohr (R.), Matching by local invariants.
Technical report n^o RR-2644, INRIA, August 1995.

[SEB 00]. Sebe (N.) et Lew (M.S.). -Color based retrieval and recognition. IEEE International Conference on Multimedia and Expo (I), pp. 311{314.-2000.

[SME 00]. Smeulders (A.W.M), Worring (M.), Santini (S.), Gupta (A.) et Jain (R.). Content-based image retrieval at the end of the early years. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, n^o12, December 2000, pp. 1349-1380.

[SMI 98]. Smith (J.R.). Image retrieval evaluation. IEEE Workshop on Content based Access of Image and Video Libraries (CBAIVL'98), pp. 112,113. June 1998.

[SWA 91]. Swain (M.J.) et Ballard (D.H.), Color indexing. International Journal of Computer Vision, vol. 7, n^o 1, 1991, pp. 11-32.

[SWE 95]. W. Sweldens. The lifting scheme: A construction of second generation wavelets. Technical Report 1995:6, Industrial Mathematics Initiative, Department of Mathematics, University of South Carolina, 1995.

[TAM 78]. H. Tamura, S. Mori, and T. Yamawaki, "Texture features corresponding to visual perception," *IEEE Trans. On Systems, Man, and Cybernetics*, vol. Smc-8, No. 6, Juin 1978.

[TUY 99]. Tuytelaars (T.) et Van Gool (L.). -Content-based image retrieval based on local affinity invariant regions. Third Int'l Conf. on Visual Information Systems, Visual99, pp. 493-500. -1999.

[KNI 00]. Knighten, B2000, Peer-to-peer computing.
http://cnscenter.fe.co.kr/resource/hotopic/p2p/PtP_IDF_Rev 1.11-web.pdf

[WIN 99]. Winter (A.) et Nastar (C.). -Differential feature distribution maps for image segmentation and region queries in image databases. IEEE Conference on Computer Vision and Pattern Recognition (CVPR'99). -Ft Collins, Colorado, June 1999.

[WKE 01]. W. Keith Edwards,*Core Jini, Second Edition*. Prentice Hall,2001.

[ZEE 02]. P.M. de Zeeuw, A toolbox for the lifting scheme on quincunx grids (LISQ), REPORT PNA-R0224 , 2002.

Les communications internationaux :

[LAK 05]. M.LAKEHAL, B.BOUDERAH, M.S.MAAMRA, Indexation automatique des images, Congrès international en informatique appliquée, centre universitaire Bordj Bou Arréridj, Novembre 2005.

[LAK 06]. M.LAKEHAL, B.BOUDERAH, M.S.MAAMRA, Application des ondelettes non séparable pour l'analyse de la texture en vue d'une opération d'indexation, Rencontre internationale d'analyse mathématique et ses application (RAMA V), Université de M'sila, Avril 2006.

ملخص:

الفهرسة و البحث عن الصور بواسطة محتواها المرئي يشكلان ميدان بحث جد متشعب و ذلك بالنظر للتطور والانتشار الواسعين للصورة في شتى الميادين.

من جهة الفهرسة، نقترح تطبيق التحويل بالموجات بالتنسيق مع الفينات و اللامتغيرات كأداتين فعاليتين من أجل حساب فهرس الصور و ذلك بالاعتماد على حياكة كل صورة.

من جهة البحث، نستعمل مبدأ البحث بالمشابهة الذي بمقتضاه يتم ترتيب الصور بقياس المشابهة تبعاً للطلب. هذا الأخير عبارة عن صورة. في نفس هذا الإطار نستعمل البعد الهندسي بين الفهارس. كذلك هنالك لمحة عن البيانات الموزعة أين نطمح إلى الفهرسة و البحث الموزعين.

الكلمات المفتاحية:

الفهرسة، البحث، الموجات، المشابهة، الفينات و اللامتغيرات، البيانات الموزعة.

Abstract:

Content based images indexing and retrieval (CBIR), constitute an varied research area, because the expansion of numerical images.

From the indexing point of view, we propose to apply the wavelet transform commonly with moments and invariants as powerful tools to compute signatures summarizing texture information of images.

From the retrieval point of view, we use the search by similarity principle according him, images are ranked according to a similarity measure with reference to the query. The request is made of an example image. In the same border, we use the geometrical distance between the signatures. Also there are an flying over distributed architecture, where we hope a distributed CBIR.

Key words:

indexing, retrieval, wavelet, similarity, moments and invariants, distributed architecture.

Résumé:

L'indexation et la recherche des images par le contenu (CBIR), constituent un domaine de recherche très varié étant donné le développement de l'image numérique.

Du point de vue de l'indexation, nous proposons d'appliquer la transformé en ondelettes communément avec les moments et invariants comme des outils puissants pour le calcul de signatures en se basant sur l'information texture relative des images.

Du point de vue de la recherche, nous utilisons le principe de la recherche par similarité selon lequel les images sont ordonnées à l'aide de la mesure de similarité à la requête. Cette dernière prend la forme d'une image exemple. Dans ce même cadre, nous utilisons la distance géométrique entre les signatures. aussi il y a un survol sur les architectures distribuées, où nous espérons un CBIR distribué.

Mots clés :

Indexation, recherche, ondelettes, similarité, moments et invariants, architectures distribuées.