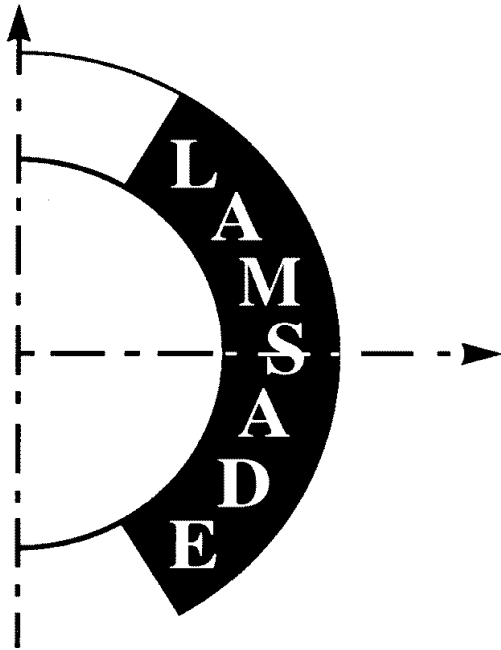


Bases de Données Images



Maude Manouvrier

LAMSADE - Université Paris Dauphine

manouvrier@lamsade.dauphine.fr

http://www.lamsade.dauphine.fr/~manouvri/BD/BDMM_M2ISI.pdf

Plan

- 1. Quelques définitions ...**
- 2. Représentation et interrogation des images par le contenu**
- 3. Structures multidimensionnelles et recherche par similarité**
- 4. Représentation des images basée sur le stockage**
- 5. Exemples de domaines d'application**
- 6. Conclusions**

1. Quelques définitions ...

- **Média / Multimedia**
- **Intégration des données multimédia dans les bases de données**
- ***Content-Based Image Retrieval (CBIR) / Recherche d'Images par le contenu***

Quelques définitions ...

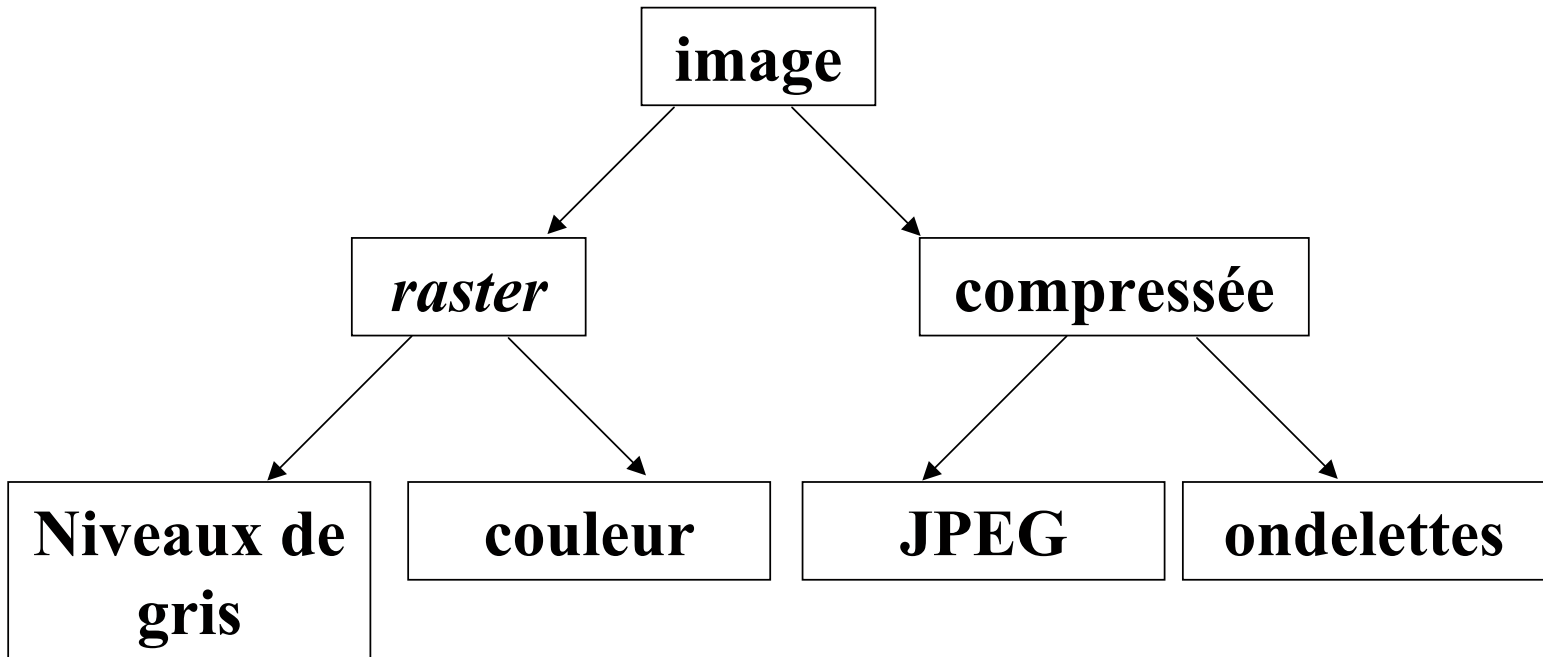
Media : types d'information ou de représentation d'information

- Texte
 - Image
 - Son
 - Vidéo
- media statique
- media dynamique

MultiMedia : collection de différents médias

Quelques définitions ...

Chaque média a ses propres caractéristiques :



Bases de données multimédia ...

Un système de gestion de bases de données multimédia doit :

- **offrir les mêmes caractéristiques qu'un SGBD classique (persistance, gestion des transactions, interrogation ...)**
- **permettre de représenter, interroger et manipuler tout type de media**

Les recherches actuelles :

**représentation, stockage, interrogation,
liaison image et sémantique**

Intégration du multimédia dans les BD classiques (1/4)

Exemple : package (ORDSYS) de classes fourni par Oracle
(8 - objet-relationnel) pour gérer des données multimédia :

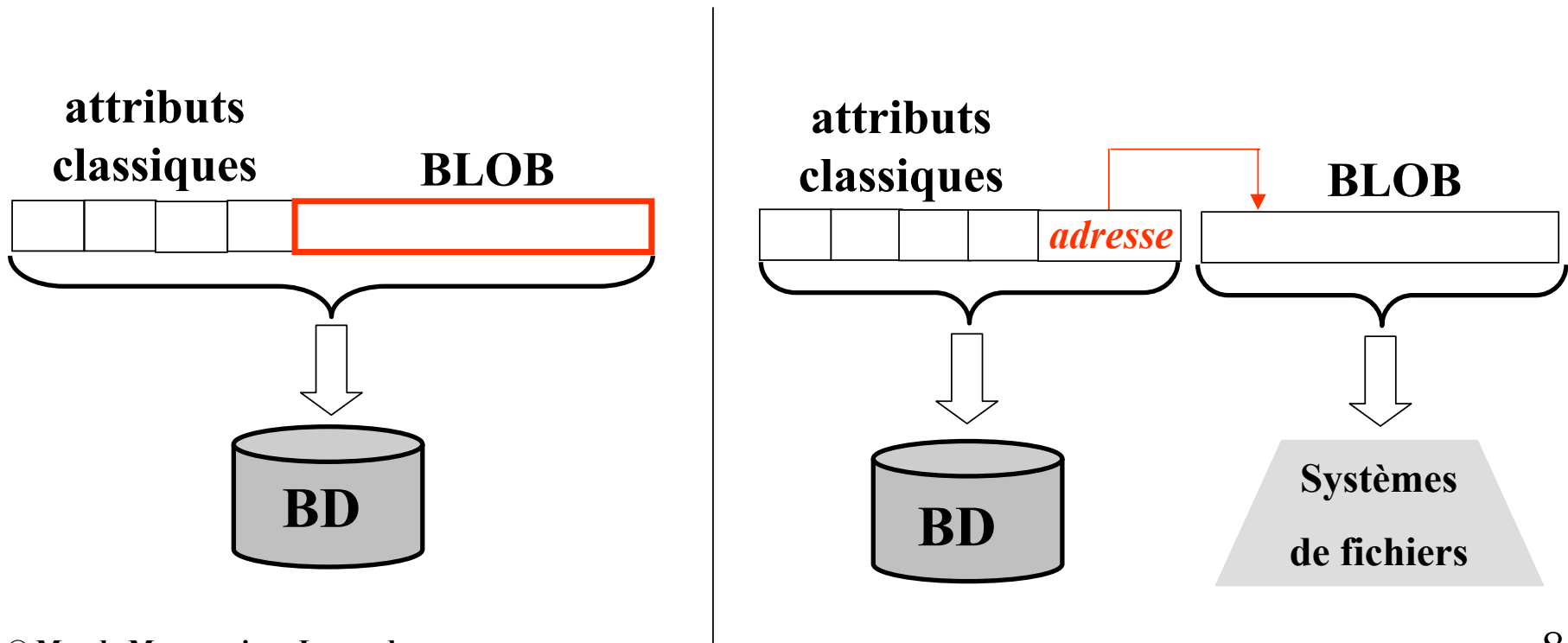
Classe ORDSOURCE

```
localData BLOB,  
srcType VARCHAR2(4000),  
srcLocation  
VARCHAR2(4000),  
srcName VARCHAR2(4000),  
updateTime DATE,  
local NUMBER
```

- **localData** : contient les données multimédia stockées localement comme un BLOB (*Binary Large Object* - jusqu'à 4 GB).
- **srcType** : identifie le type de source des données (FILE, HTTP, ...)
- **srcLocation** : identifie où les données peuvent être trouvées (dépendant de srcType (FILE -> directory, HTTP -> URL ou user-defined).
- **srcName** : identifie le nom des données.
- **updateTime** : la date à laquelle le fichier a été modifié pour la dernière fois.
- **local** : valeur qui détermine si le fichier est local ou pas (1 = BLOB, 2 = externe).

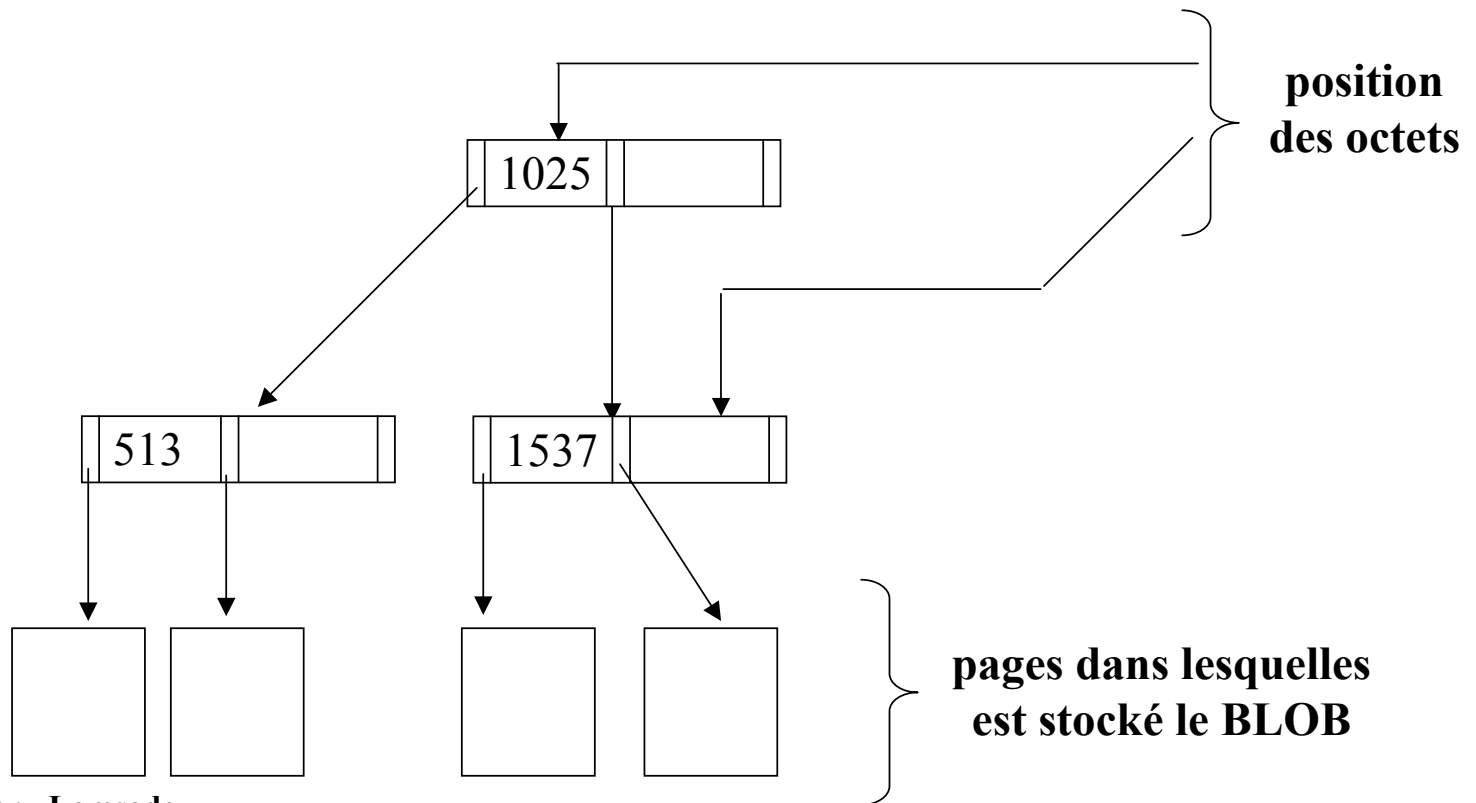
Intégration du multimédia dans les BD classiques (2/4)

BLOB (*Binary Large Object*)



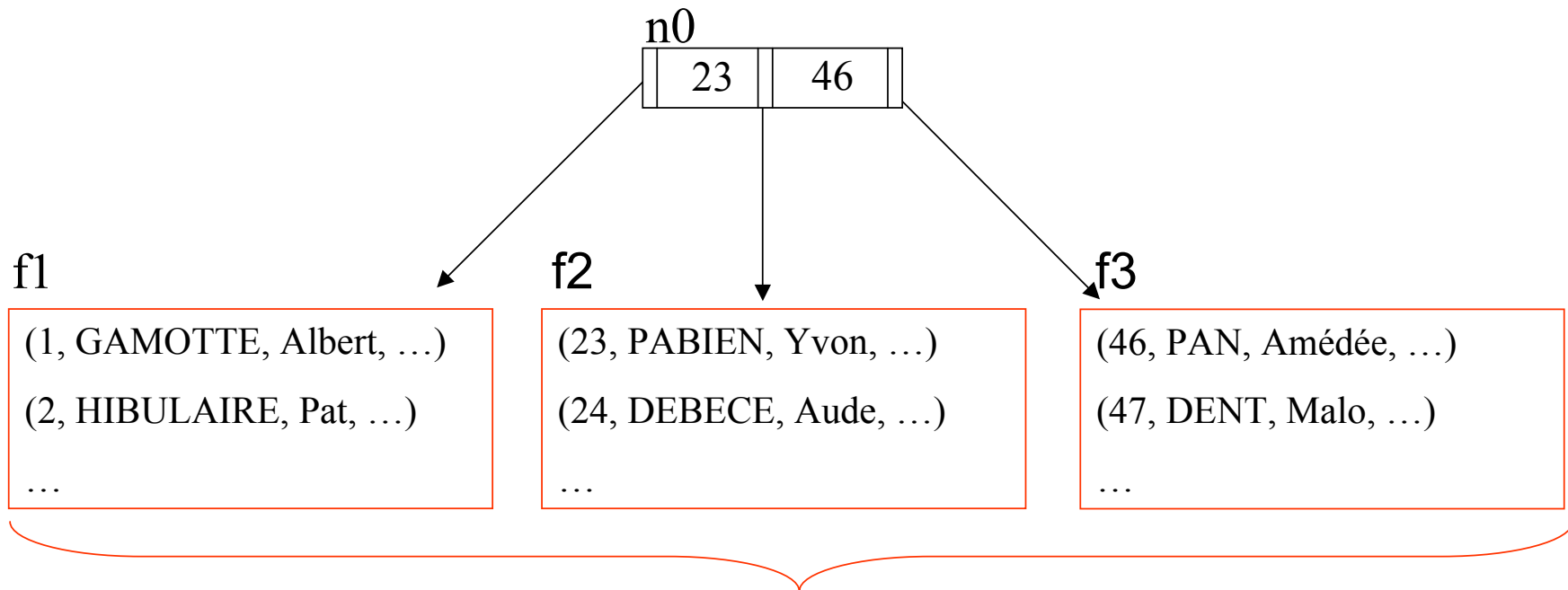
Intégration du multimédia dans les BD classiques (3/4)

Structures de données utilisée : arbre B positionnel



Arbre B+ (1/3)

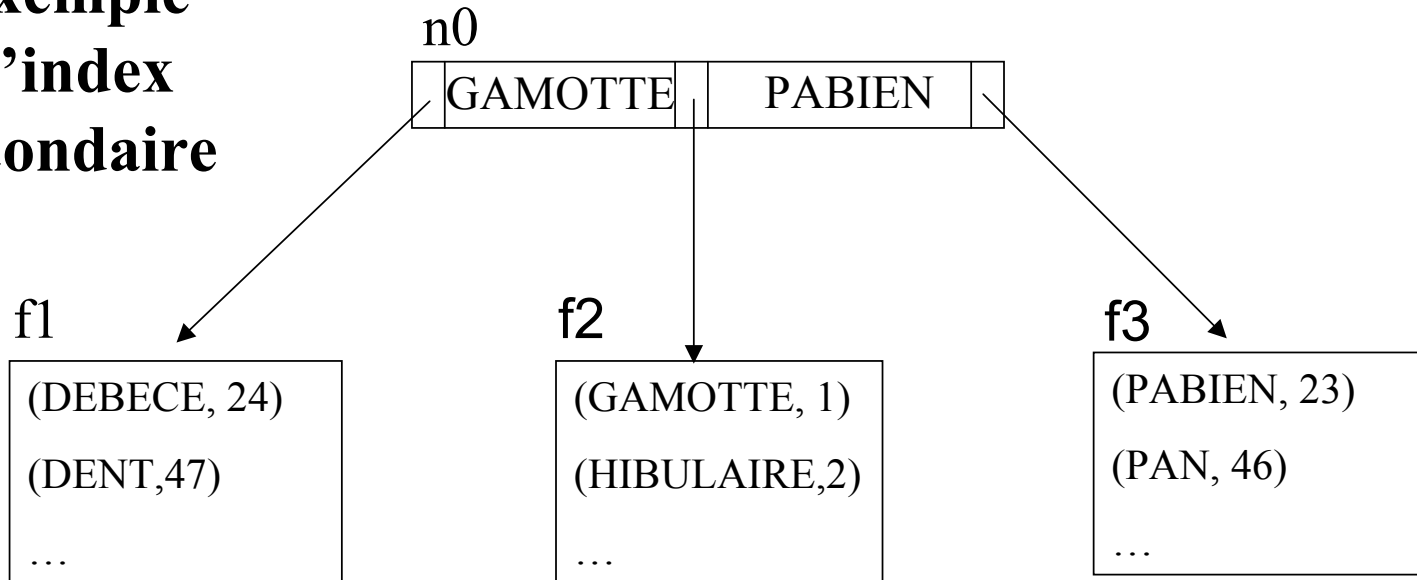
Exemple d'index primaire



Pages de la relation et feuilles de l'index

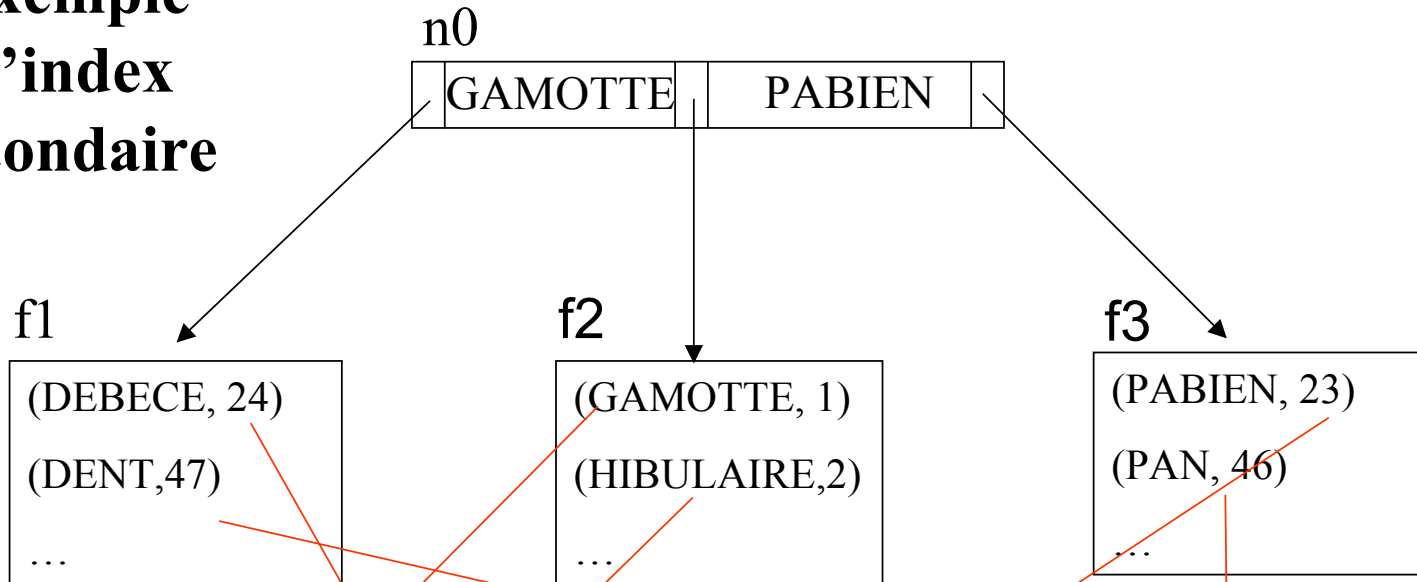
Arbre B+ (2/3)

Exemple d'index secondaire

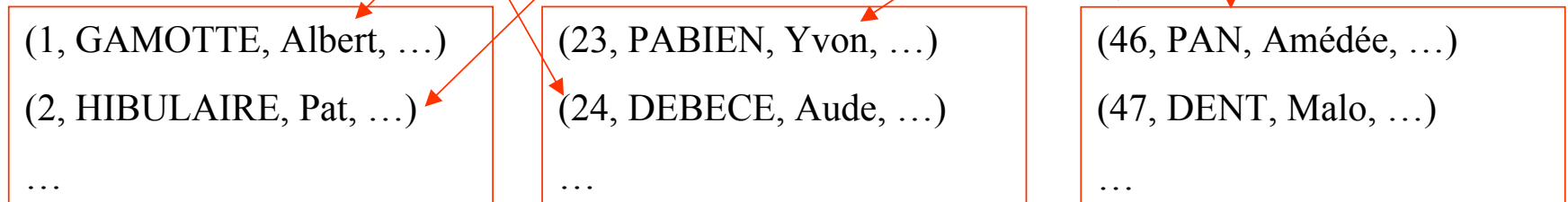


Arbre B+ (2/3)

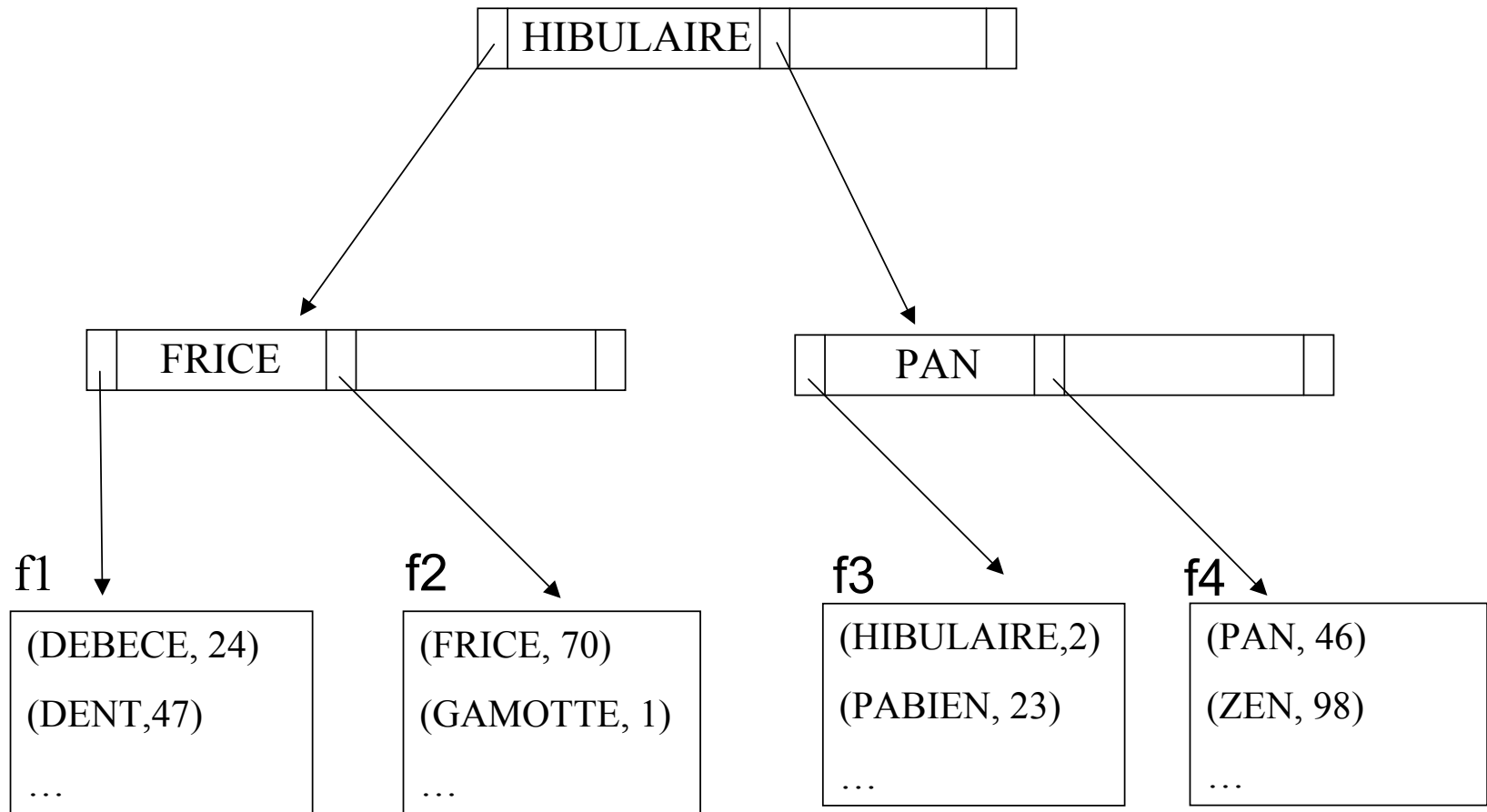
Exemple d'index secondaire



Pages de la relation

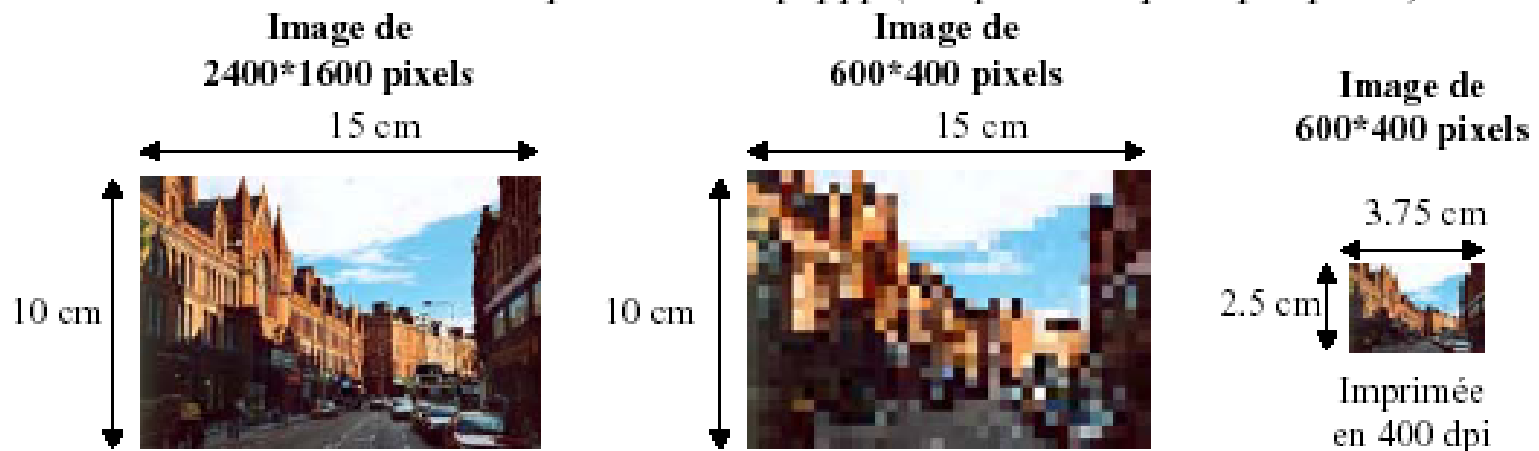


Arbre B+ (3/3)



Images et pixels

- Le facteur de résolution d'une image
 - Une image est caractérisée par ses dimensions en nombre de pixels et non en centimètres.
 - *Fondamental pour l'impression ou l'affichage à l'écran*
 - **Définition** : nombre de pixels par unité de longueur (pouce)
 - *Unité* : ppi/ppp (pixel per inch / pixel par pouce)
 - *Ex* : cas de l'écran : 72 dpi pour un 15'' en 800*600.
 - *≠ résolution d'impression en dpi/ppp (dot per inch / point par pouce)*



Intégration du multimédia dans les BD classiques (4/4)

Classe ORDImage pour gérer les données image sous Oracle :

Classe ORDImage

```
source ORDSOURCE,  
height INTEGER,  
width INTEGER,  
contentLength INTEGER,  
fileFormat VARCHAR2(4000),  
contentFormat  
VARCHAR2(4000),  
compressionFormat  
VARCHAR2(4000),  
mimeType VARCHAR2(4000)
```

- **source** : source de l'image
- **height** : la hauteur de l'image, en pixels
- **width** : la largeur de l'image, en pixels
- **contentLength** : la taille de l'image, en bytes
- **fileFormat** : le type dans lequel l'image est stockée (jpg, tiff, etc)
- **contentFormat** : le type de l'image (monochrome, gris 8-bit, etc)
- **compressionFormat** : l'algorithme de compression (LZW, JPEG,...)
- **mimeType** : le type MIME de l'image

Interrogation recherche par le contenu

Une base de donnée classique permet de faire des requêtes exactes :

```
SELECT e.nom  
FROM Etudiants E, Inscription I  
WHERE I.Intitulé=« Bases de Données Images »  
AND I.EtudiantID=E.EtudiantID
```

L'interrogation de données multimedia n'est pas exacte :
« je veux des images de villes »

Recherche d'information par le contenu
(Content-Based Image Retrieval)

Annotation des images et recherche par annotation

Précédente → Recherche Favoris Média

Adresse <http://images.google.com/images?hl=en&lr=&ie=UTF-8&oe=UTF-8&q=dauphine+paris> OK Liens

Google
Image Search

[Advanced Image Search](#) [Preferences](#) [Image Search Help](#)

dauphine paris

[Moderate SafeSearch is on](#)

[Web](#) **[Images](#)** [Groups](#) [Directory](#) [News](#)

Searched images for **dauphine paris** Results 1 - 20 of about 1,040. Search took 0.18 seconds.



Dauphine.jpeg
450 x 311 pixels - 61k
instruct1.cit.cornell.edu/.../FranceGermany/Dauphine.jpeg



VendomeFacade.gif
450 x 311 pixels - 107k
instruct1.cit.cornell.edu/.../VendomeFacade.gif
[[More results from instruct1.cit.cornell.edu](#)]



1.jpg
380 x 283 pixels - 22k
pixal.free.fr/Sorties/Dauphine/1.jpg



Grandmont.jpg
600 x 401 pixels - 30k
www.sima.sintef.no/FSL-99/photos/Grandmont.jpg



LeTallec.jpg
600 x 401 pixels - 30k
www.sima.sintef.no/FSL-99/photos/LeTallec.jpg



paris.dauphine.JPG
402 x 260 pixels - 31k
www.activetravel.org/paris.foto7.html



01h.jpg
480 x 320 pixels - 37k
tourisme.voila.fr/.../paris/fra/sit/ville/16_eguim/zb01.htm



Plan.gif
325 x 209 pixels - 26k
www.ceremade.dauphine.fr/images/Plan.gif



© Maude Manouvrier - Lamsade

16

Annotation des images et recherche par annotation

Précédente Recherche Favoris Média Liens

Adresse <http://images.google.com/images?hl=en&lr=&ie=UTF-8&oe=UTF-8&q=dauphine+university&btnG=Google+Search> OK

Google™
Image Search

[Advanced Image Search](#) [Preferences](#) [Image Search Help](#)

dauphine university

[Web](#) **[Images](#)** [Groups](#) [Directory](#) [News](#)

Searched images for **dauphine university**. Results 1 - 20 of about 49. Search took 0.06 seconds.

Tip: In most browsers you can just hit the return key instead of clicking on the search button.



P16FacDauACHR.jpg
377 x 500 pixels - 48k
www.chez.com/fransforprojets/fransforparis/P16FacDauACHR.jp



ImageBienvenue.jpg
473 x 727 pixels - 146k
viab.dauphine.fr/



Ch.jpg
893 x 1049 pixels - 228k
www.bourin-conseil.fr/com/CH_UK.htm



PlanAccesDauphine.gif
580 x 600 pixels - 49k
cercles2002.cip.dauphine.fr/.../PlanAccesDauphine.gif



eseurat.jpg
152 x 202 pixels - 6k
www.cirpack.com/companyinfo/management.shtml



Plan.gif
325 x 209 pixels - 26k
www.ceremade.dauphine.fr/images/Plan.gif



ll.gif
177 x 245 pixels - 29k
www.dassault-developpement.fr/partners.htm



gabonmap.GIF
502 x 649 pixels - 25k
www.ansci.cornell.edu/courses/as625/2001term/dauphine/

Maude Manouvrier Lamsade DEA127 Microsoft Powe... Internet Exp... Symantec AntiV... FR 18:17

Annotation des images et recherche par annotation

Précédente Recherche Favoris Média OK Liens

Adresse <http://images.google.fr/images?hl=fr&lr=&ie=UTF-8&oe=UTF-8&imgc=gray&q=+lamsade+site%3Adauphine.fr+filetype%3Ajpg&btnG=Reche> OK

Google uphine.fr filetype:jpg Recherche Web 0 bloquée(s) Options lamsade site:dauphine.fr filetype:jpg

Google™ Images -- Recherche avancée Préférences Images -- Aide

lamsade site:dauphine.fr filetype:jpg Recherche Google

Recherche d'image

Web Images Groupes Répertoire Actualités

Google a recherché des images correspondant à **lamsade site:dauphine.fr filetype:jpg**. Résultats : 1 - 5 sur 5. Recherche effectuée en 0.97 secondes.

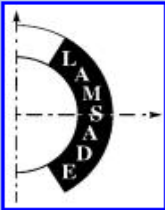
Conseil : avec la plupart des navigateurs, vous pouvez lancer la recherche en appuyant sur la touche Entrée (au lieu de cliquer sur le bouton Recherche Google).



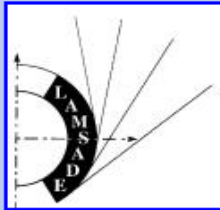
Entete.jpg
152 x 208 pixels - 20 ko
www.lamsade.dauphine.fr/ia/bandeau.html



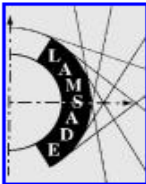
PERSPECT.jpg
576 x 281 pixels - 31 ko
www.lamsade.dauphine.fr/opti/PERSPECT.jpg



NET.jpg
329 x 450 pixels - 23 ko
l1.lamsade.dauphine.fr/local/logos/images/NET.jpg



BIAIS.jpg
435 x 450 pixels - 30 ko
l1.lamsade.dauphine.fr/local/logos/images/BIAIS.jpg



logo_lamsade.jpg
200 x 276 pixels - 23 ko
l1.lamsade.dauphine.fr/~moreaux/biomodelisation/

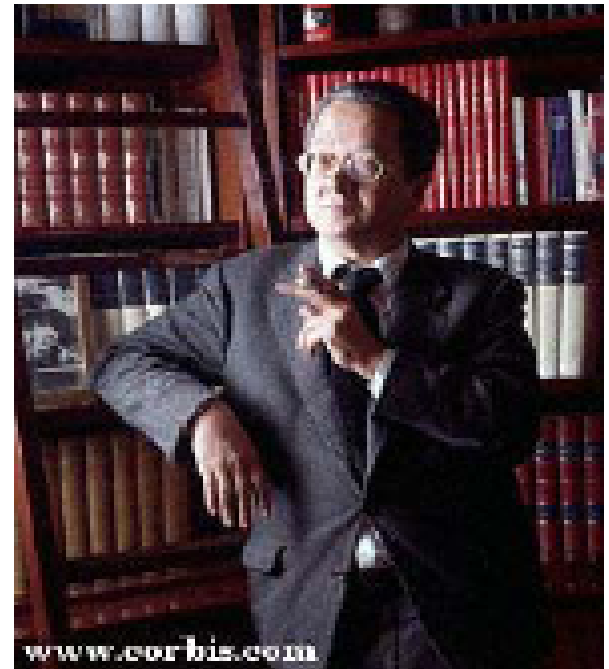
Marde Manouvrier Lamsade download Microsoft Powe... Recherche Goo... Base de donnee... FR 18

Annotation des images et recherche par annotation

→ Thésaurus, ontologie ...



"Avocat" ?

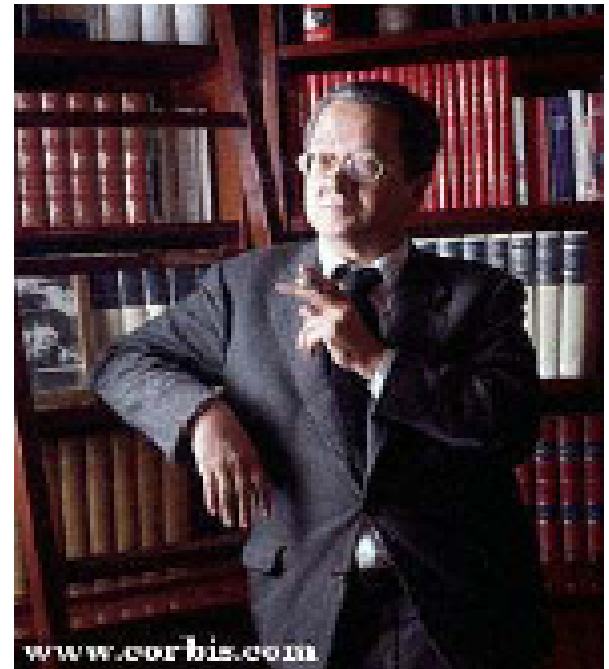


Annotation des images et recherche par annotation

→ Thésaurus, ontologie ...



"Avocat" ?

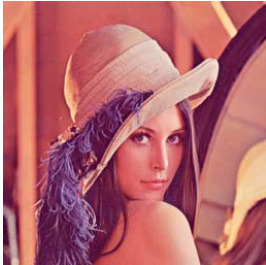


Fossé sémantique / semantic gap

2. Représentation et Interrogation des Images par le Contenu

- **Images et Bases de Données et domaine d'application**
- **Caractéristiques visuelles des images**
- **Représentation des images**
- **Différentes méthodes d'interrogation**
- **Exemples de requêtes sur des systèmes *online***

Ce qui est stocké dans la base de données

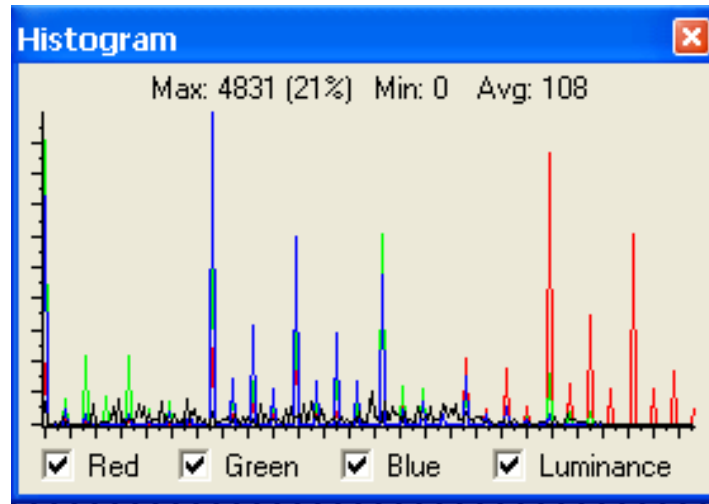
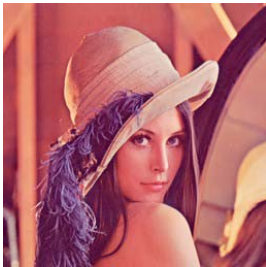


Ce qui est stocké dans la base de données



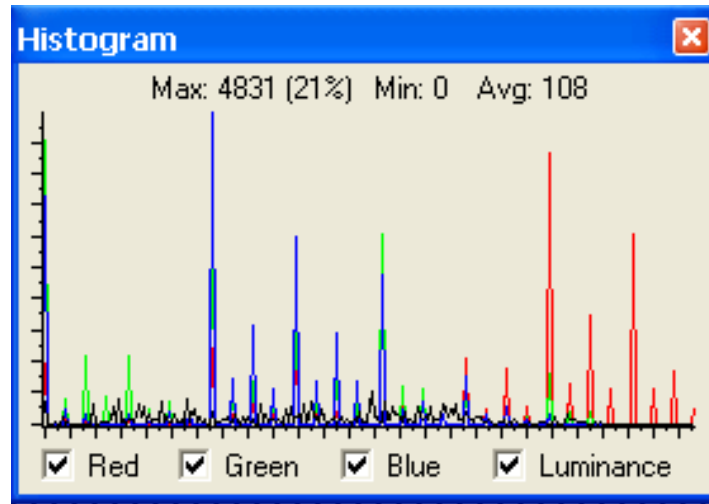
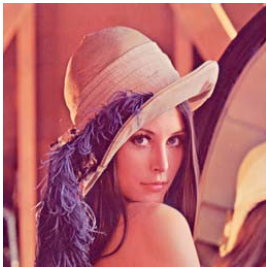
Traitement d'images

Ce qui est stocké dans la base de données



Traitement d'images

Ce qui est stocké dans la base de données



Traitement d'images

Ce qui est stocké dans la base de données

Histogramme de couleurs
(23,34,5,6, 66, ...)

Descripteur de contour 1
(120,45, ...)



Descripteur de contour 2
(10,234, ...)

Descripteur de texture
(72,34,6, ...)

Traitement d'images

Ce qui est stocké dans la base de données

Histogramme de couleurs
(23,34,5,6, 66, ...)

Descripteur de contour 1
(120,45, ...)

Descripteur de contour 2
(10,234, ...)

Descripteur de texture
(72,34,6, ...)

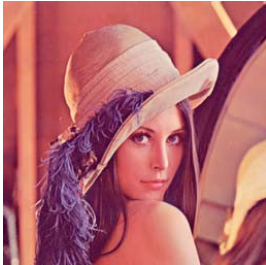


Image de Lena

Domaine public

Traitement d'images

**Texte ajouté par
l'utilisateur**

Ce qui est stocké dans la base de données

Histogramme de couleurs
(23,34,5,6, 66, ...)

Descripteur de contour 1
(120,45, ...)

Descripteur de contour 2
(10,234, ...)

Descripteur de texture
(72,34,6, ...)

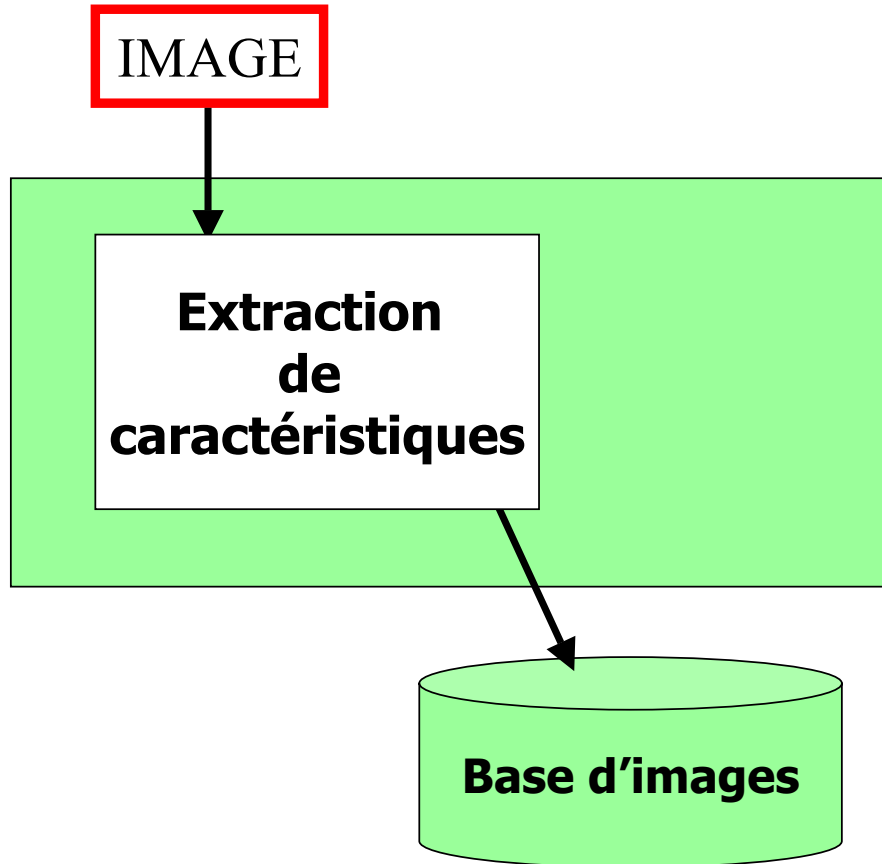


~~*Image de Lena*~~
~~*Domaine public*~~

Recherche par le contenu
Content-Based Image Retrieval (CBIR)

Texte ajouté par
l'utilisateur

Recherche d'images par le contenu



La base de données contient les **descripteurs** (ou **signatures** ou **vecteurs de caractéristiques**) extraits des images et représentant leur contenu visuel

Recherche d'images par le contenu : domaines d'application

- **Domaine scientifique**

- **Bases d'images satellites**
- **Bases d'images médicales**

Ex : retrouver les images présentant un caractère pathologique, dans un but éducatif ou diagnostic.

- **L'audiovisuel**

Ex : retrouver un plan spécifique d'un film ou d'un journal télévisé - annotation automatique de vidéos.

- **L'authentification/sécurité – la lutte contre le crime**

Ex : détecter les contrefaçons de modèles déposés, identifier un visage ou des empreintes digitales etc.

Recherche d'images par le contenu : domaines d'application (suite)

- Le design, la publicité, l'architecture

Ex : rechercher une texture spécifique pour l'industrie textile, illustrer une publicité par une photo adéquate.

- L'art, l'éducation

Ex : recherche encyclopédique d'illustrations, d'un tableau ou d'un œuvre d'art spécifique (Ministère de la culture, numismatique).

- Les télécommunications

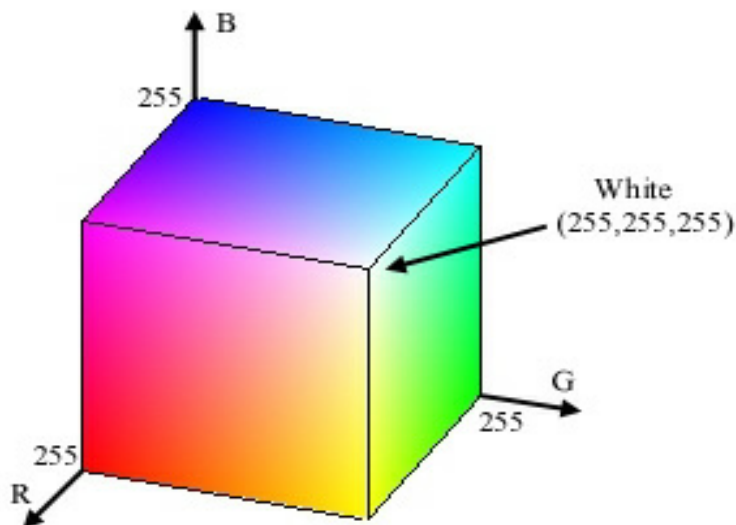
- Internet

Articles/chapitres/ouvrages *survey* sur la recherche d'images par le contenu

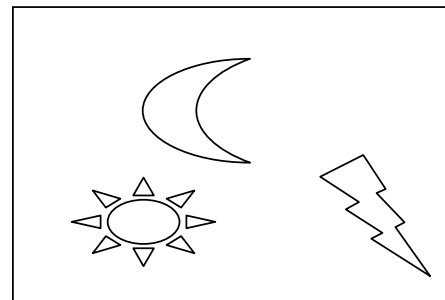
- Arnold W. M. Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta, Ramesh Jain: *Content-Based Image Retrieval at the End of the Early Years*. IEEE Transactions on Pattern Analysis and Machine Intelligence 22(12): 1349-1380 (2000)
- R. C. Veltkamp, M. Tanase *Content-based image retrieval systems: A survey* - Rapport Technique UU-CS-2000-34 – Revised version 2002
<http://www.aa-lab.cs.uu.nl/cbirsurvey/cbir-survey/cbir-survey.html> ou <http://give-lab.cs.uu.nl/cbirsurvey/>
- *Image Databases - Search and Retrieval of Digital Imagery*
Edité par V. Castelli et L.D. Bergman, Wiley, 2002
- *State-of-the-Art in Content-Based Image and Video Retrieval*
Edité par R.C. Veltkamp, H. Burkhardt et H-P. Kriegel, Kluwer Academic Pub., 2001
- *Content-Based Image and Video Retrieval*
O. Marques et B. Furhy, Kluwer Academic Pub., 2002
- *Multimedia Mining : A Highway to Intelligent Multimedia Documents*
Edité par C. Djeraba , Kluwer Academic Pub., 2003

Caractéristiques des images

■ Couleur



■ Forme



■ Texture



Tapis tissé



Granit



Papier kraft

Couleur

- Représentation à l'aide de trois composantes RGB (*Red, Green, Blue*)

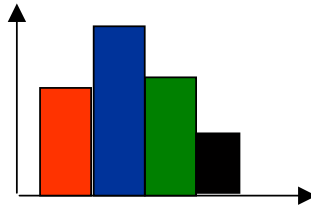
Un pixel = un triplet (r, g, b) avec $r, g, b \in [0, 255]$

$(0, 0, 0)$ pour noir,

$(255, 255, 255)$ pour blanc,

$(200, 60, 60)$ pour rose

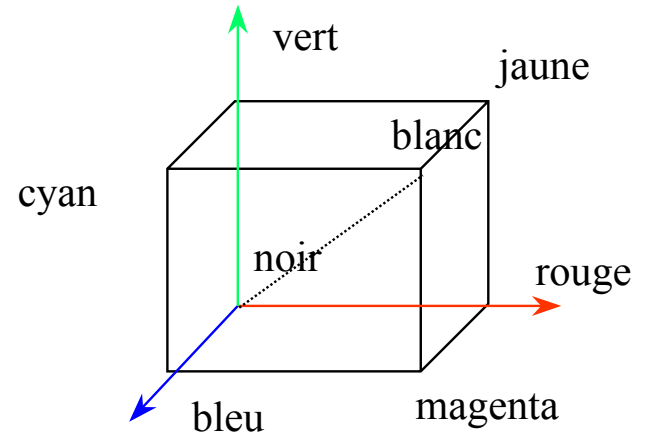
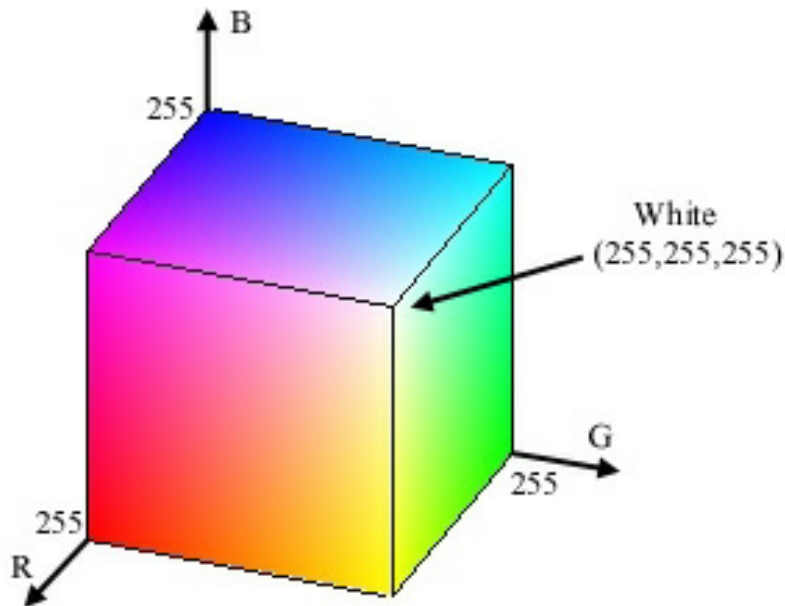
- Représentation à l'aide d'un histogramme de couleurs



Modèles de représentation de la couleur (1/6)

Espace de couleurs RGB (*Red, Green, Blue*)

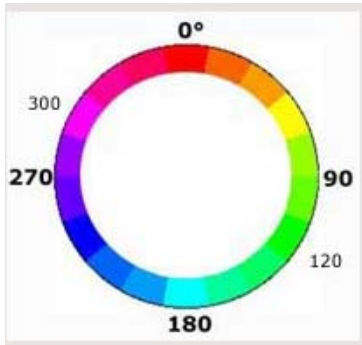
- système de couleurs généralement dépendant des périphériques informatiques : Couleurs primaires pour la lumière (dédié aux moniteurs).
- système non perceptiblement uniforme



Modèles de représentation de la couleur (2/6)

Espace de couleurs HSV - TSL

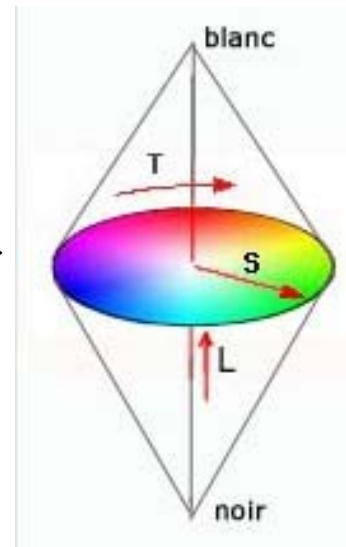
(*Hue - Teinte, Saturation, Value - Luminance*)



- **Teinte** : indication du type de couleur: (rouge, jaune, bleu, etc.)
Exprimée par un nombre correspondant à sa position angulaire sur le cercle chromatique (à partir du haut, dans le sens horaire).
ex : rouge : 0; vert : 120; magenta : 300.



- **Saturation** : indication de la pureté de la couleur (pâle - tons pastels - ou vive).
Représentée sur le rayon du cercle, par un pourcentage de pureté : maximale sur le cercle (100%) et minimale au centre (0 = gris)



- **Luminosité** : indication de la clarté de la couleur.
Exprimée par un pourcentage : de 0% (noir) à 100% (blanc)

Modèles de représentation de la couleur (3/6)

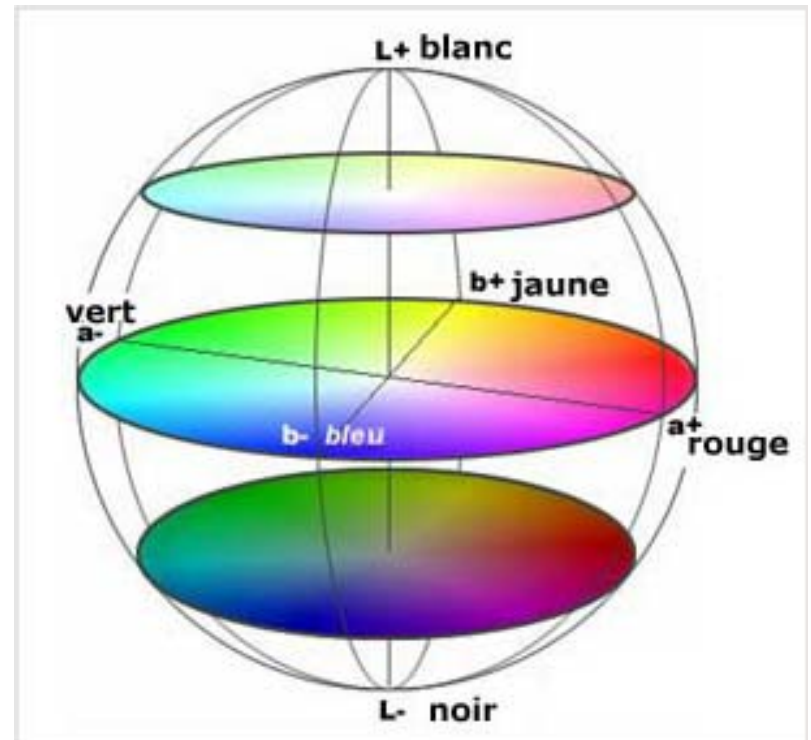
Espace CIE - $L^*a^*b^*$

(Commission Internationale de l'Eclairage)

- L – luminosité de noir (0%) à blanc (100%)
- a – gamme du vert au rouge
- b – couleurs du bleu au jaune

Modèle de définition des couleurs indépendant de la technologie du périphérique utilisé

Dans cet espace, deux couleurs proches en distance le sont aussi pour l'œil.



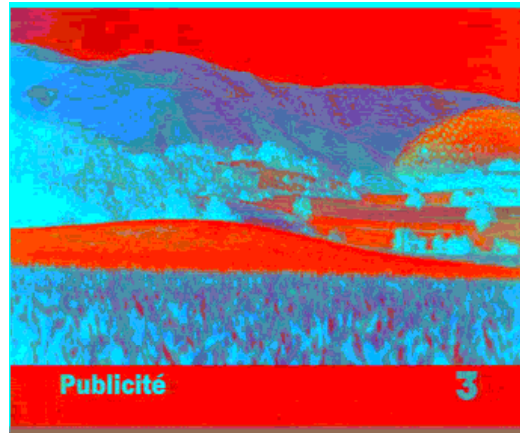
Modèles de représentation de la couleur (4/6)

RGB

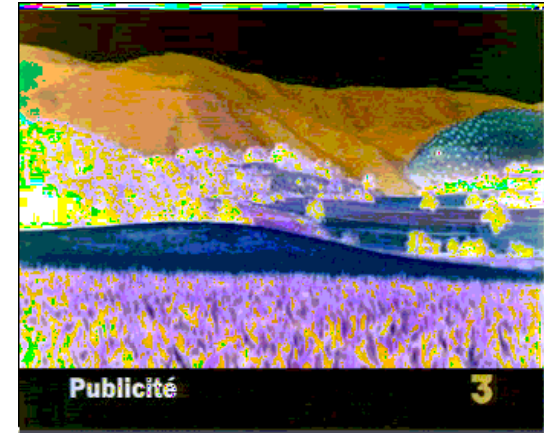
Représentation d'une image dans les différents espaces de couleur



HSV



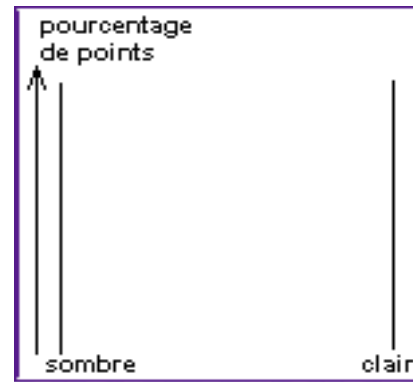
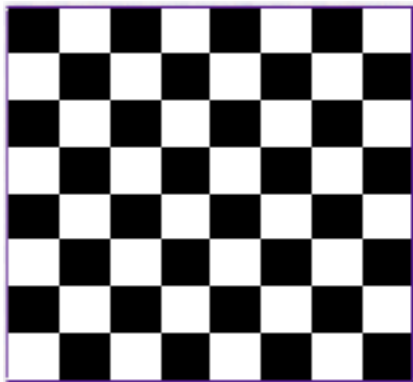
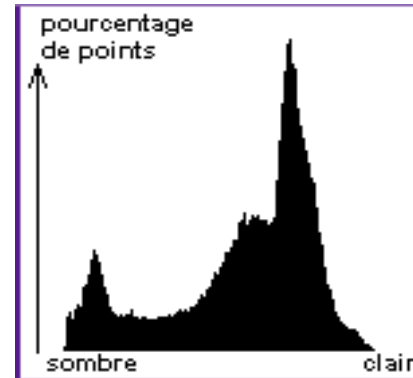
CMY (*imprimantes couleurs à impact, et à jet d'encre*)



YUV (*PAL et SECAM des standards de la télévision couleur*)

Modèles de représentation de la couleur (5/6)

Représentation des images en niveaux de gris par des histogrammes de couleurs



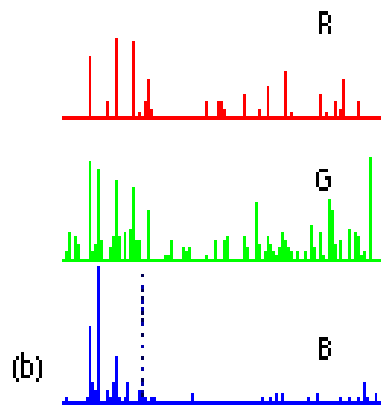
Modèles de représentation de la couleur (6/6)

Représentation des images couleurs

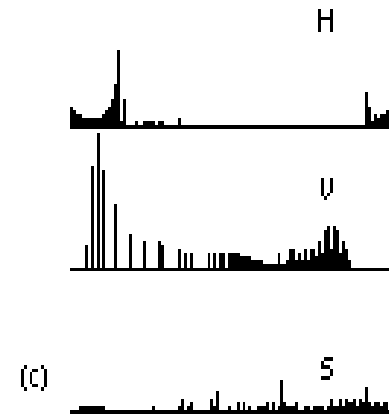
- Histogrammes «uni-dimensionnel» à partir desquels on construit un histogramme tri-dimensionnel



(a)



(b)

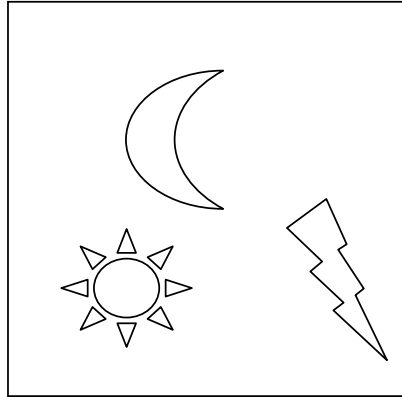


(c)

- Moments statistiques (moyenne, écart-type...)
- Couleur dominante

Forme ou contour

- Caractérisation des objets contenus dans l'image
- Représentation par de composantes telles que *périmètre, axes d'orientation* etc.



Modèles de représentation de la forme (1/5)

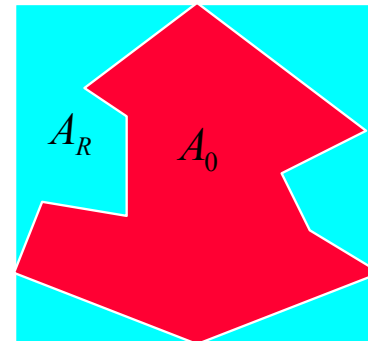
Caractéristiques internes : description de la région occupée par l'objet (surface, transformée de Fourier...)

Caractéristiques externes : description du contour de l'objet représenté par le périmètre, circularité, rectangularité, le codage de Freeman, les descripteurs de Fourier...

• $\text{élongation} = \frac{W}{H}$ avec W : largeur
H : hauteur

• $\text{circularité} = \frac{T^2}{4\pi A}$ avec A : surface
T : périmètre

• $\text{rectangularité} = \frac{A_o}{A_R}$

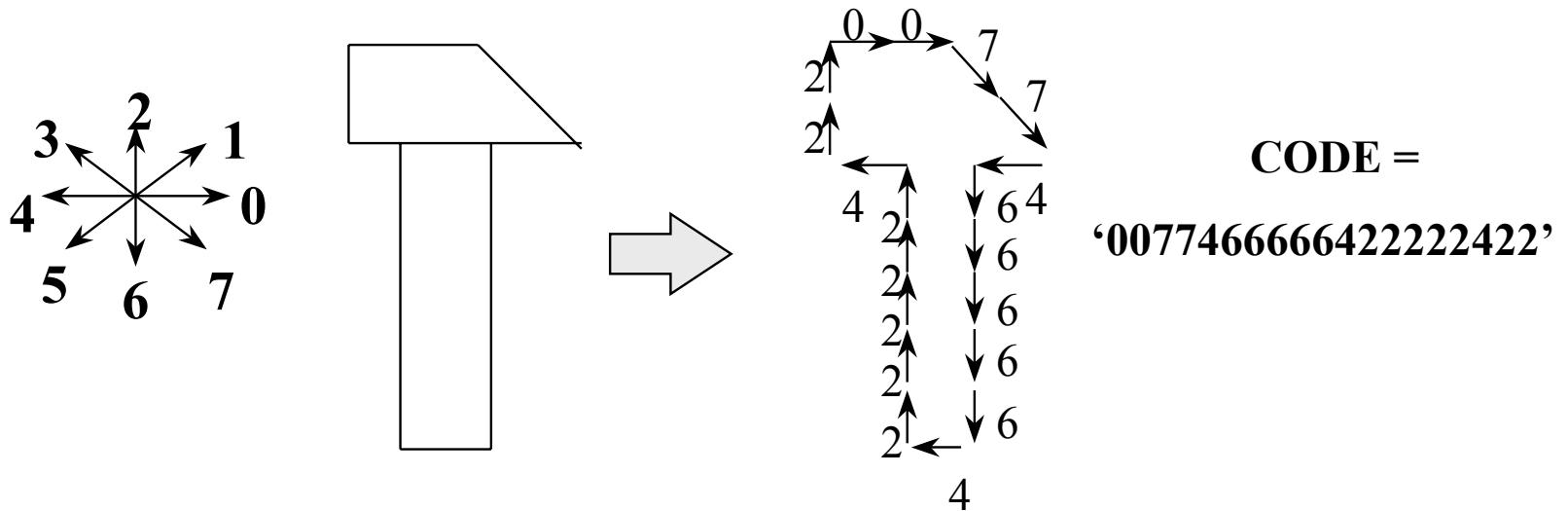


A_R surface du rectangle
d'encadrement

A_0 surface de la région

Modèles de représentation de la forme (2/5)

Codage de Freeman



Modèles de représentation de la forme (3/5)

Transformée de Fourier

Image = répartition d'intensités lumineuses dans un plan
= **un signal à deux dimensions continu ou discret**
(ex.cas particulier des images traitées par ordinateur).

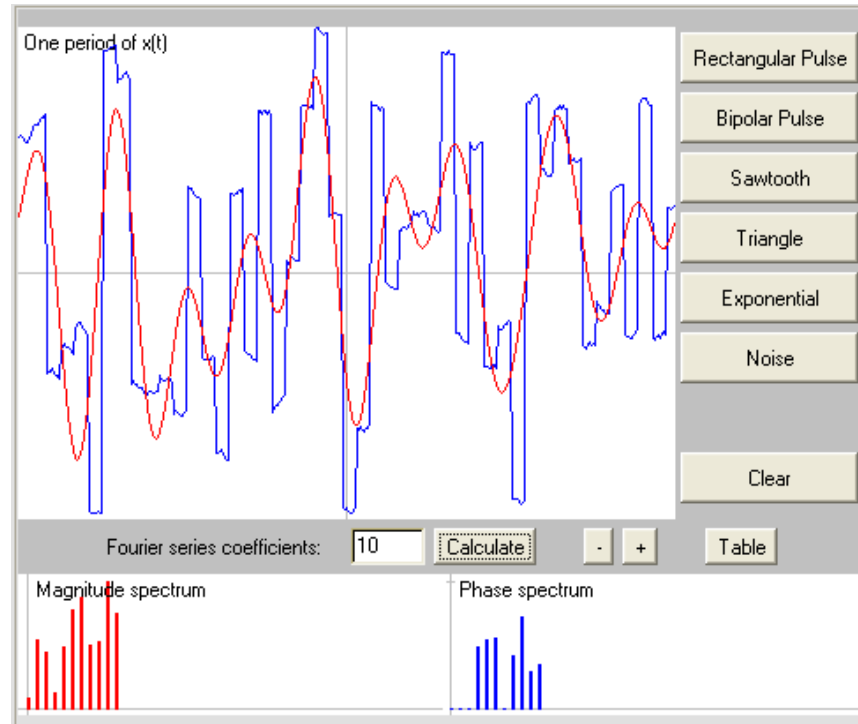
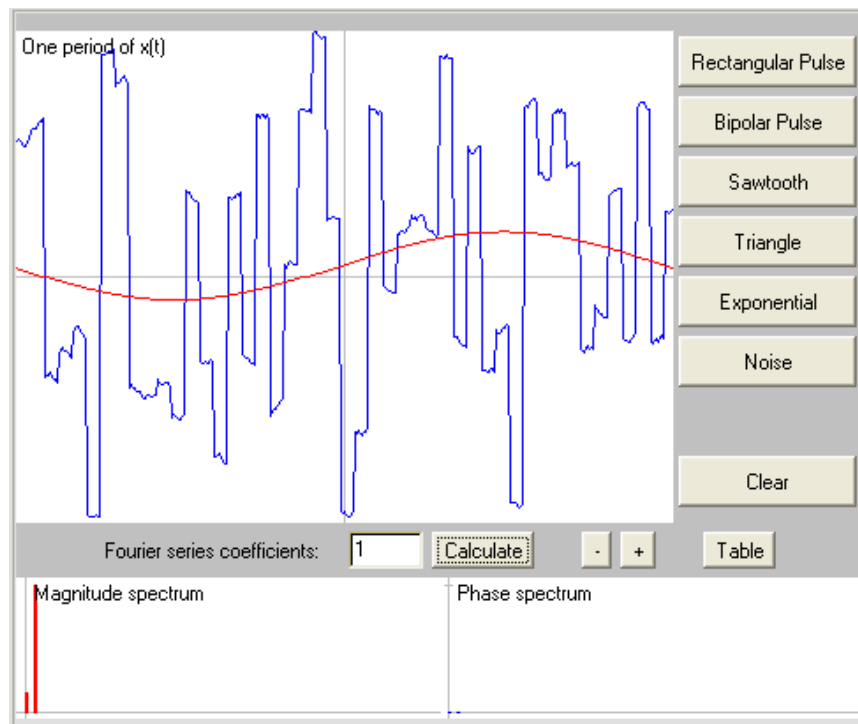
Intensité en un point de l'image : peut être représentée par un nombre,
ex. le niveau de gris pour une image en noir et blanc.

Pour les images " complexes » : représentation de l'intensité en un point non plus par un nombre réel mais **par un nombre complexe**.

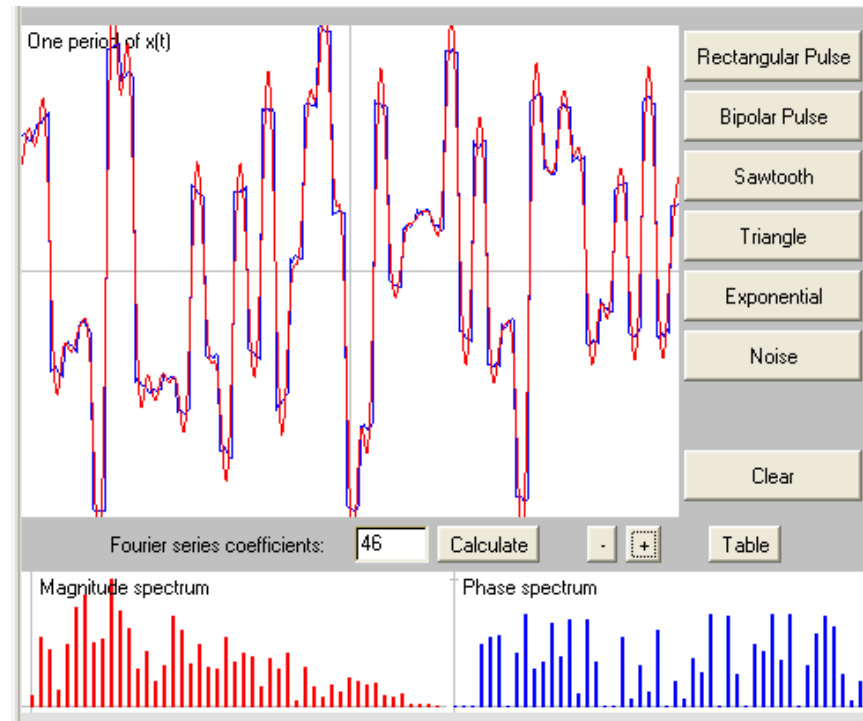
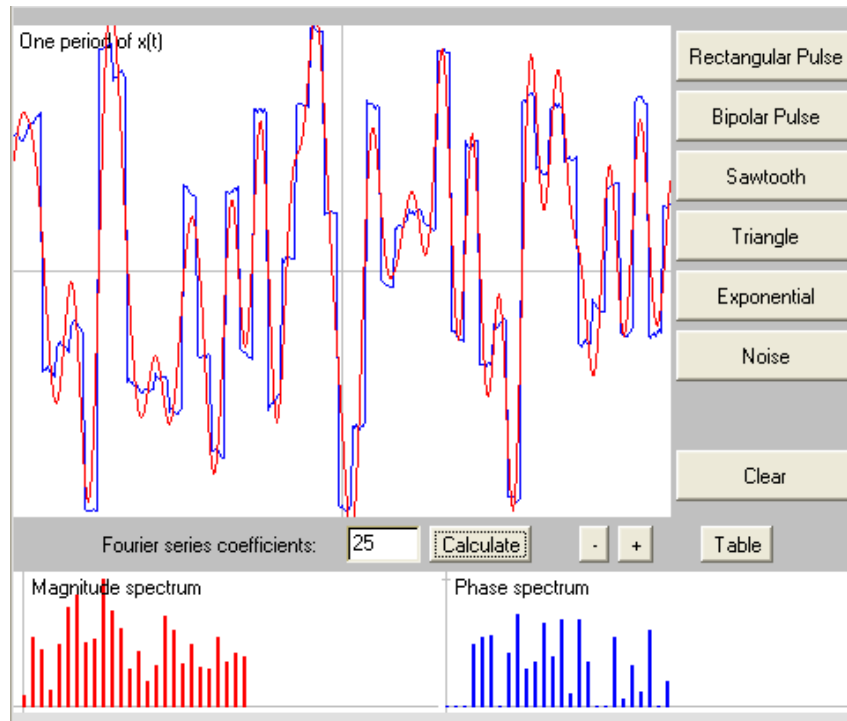
Transformée de Fourier = séries d'opérations mathématiques qui permet d'associer, à un signal, une série de sinusoides de fréquences, d'amplitudes et de phases déterminées (ou *partiels*)

Adaptée aux signaux numériques échantillonnés $\Rightarrow$ **Transformée de Fourier Discrète (DFT)**

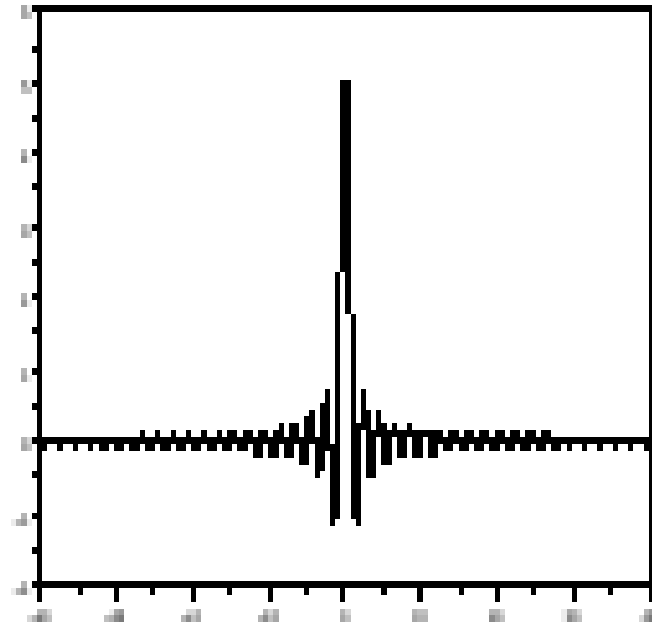
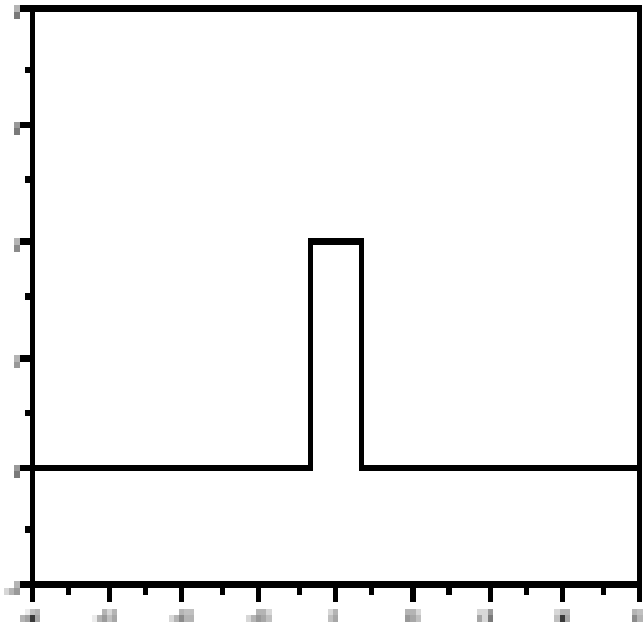
Transformée de Fourier



Transformée de Fourier



Transformée de Fourier



Un signal et sa transformée de Fourier

Transformée de Fourier

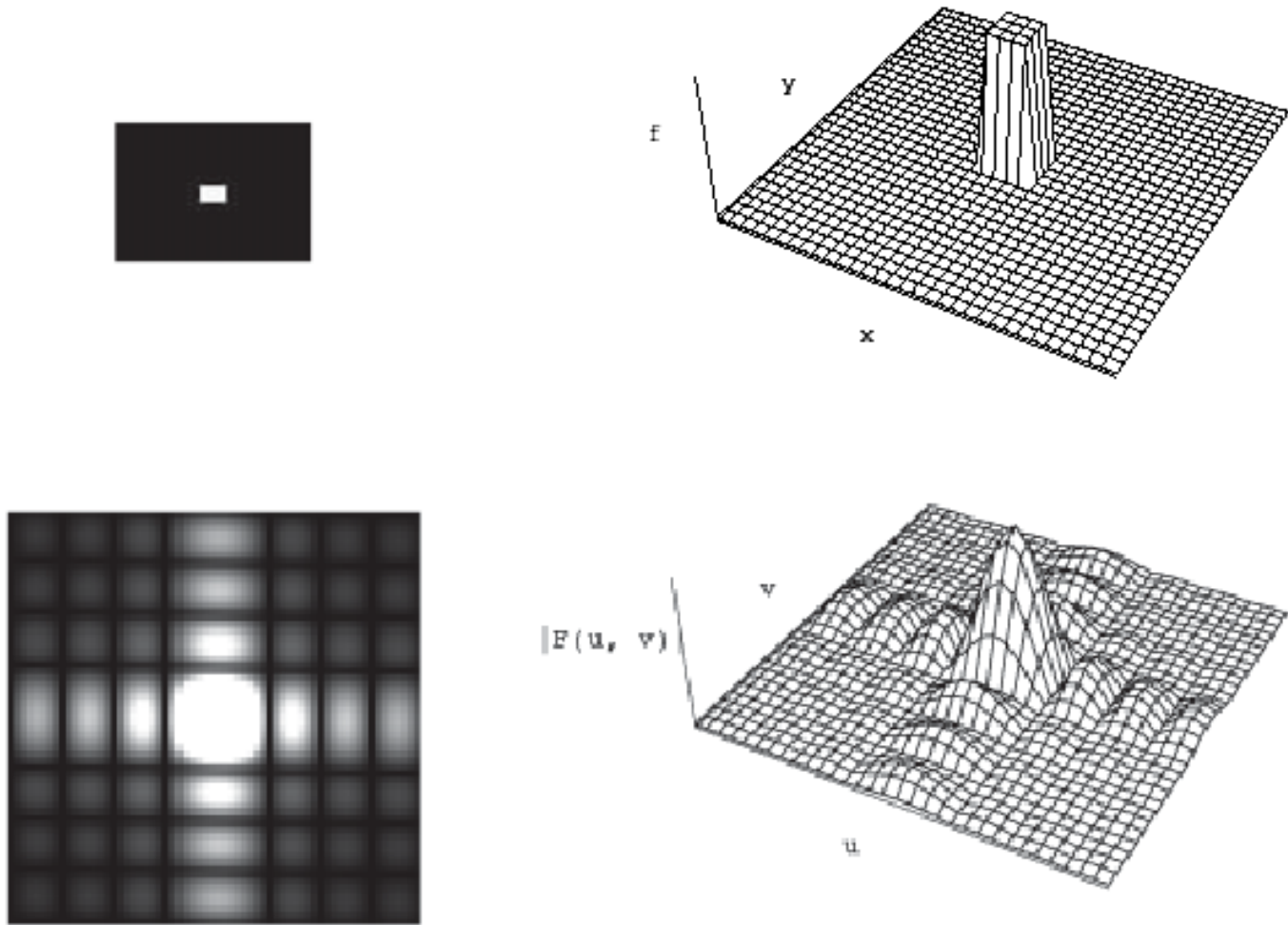
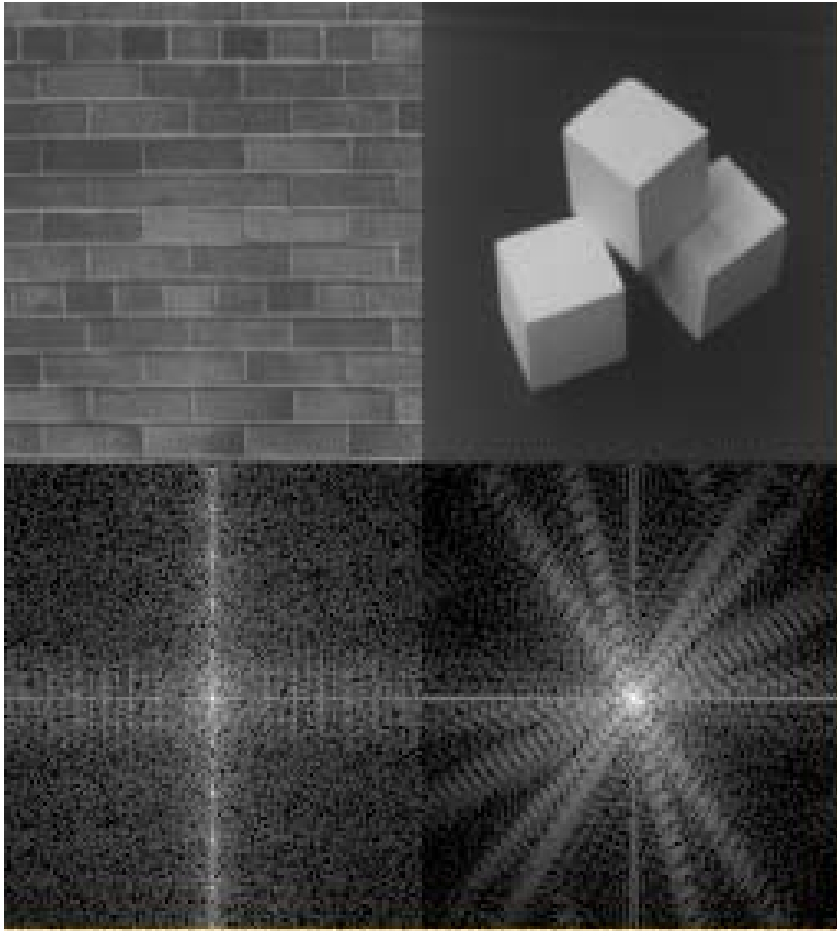


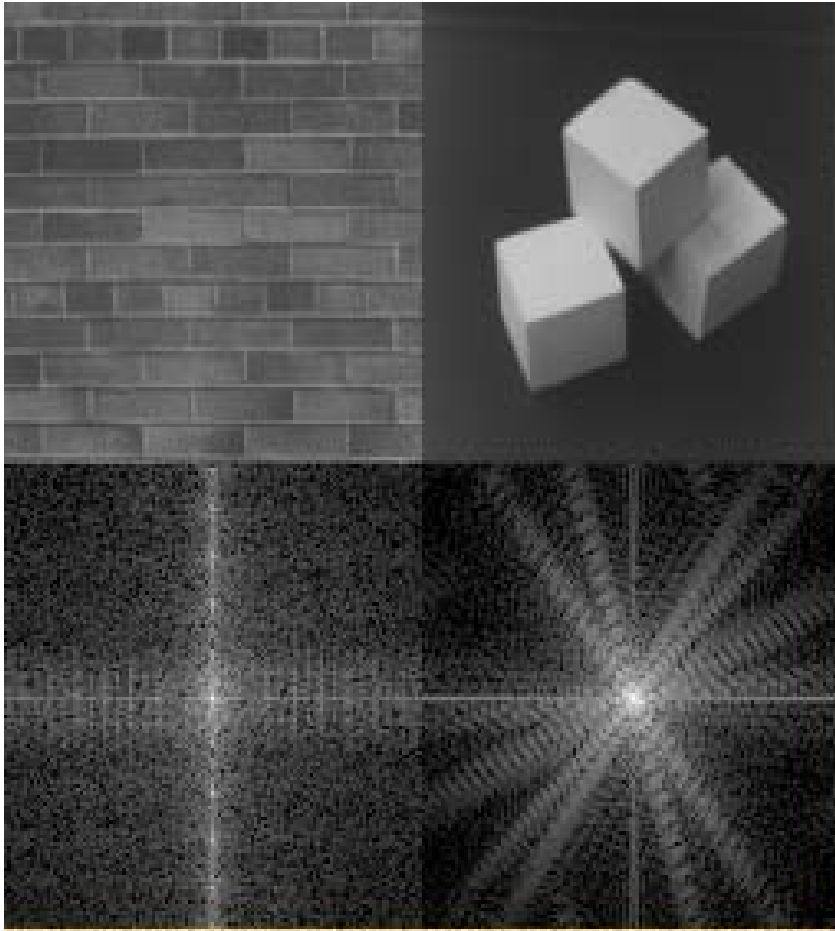
Figure 1.8 Une image bidimensionnelle et sa transformée de Fourier.

Modèles de représentation de la forme (4/5)

Modèles de représentation de la forme (4/5)

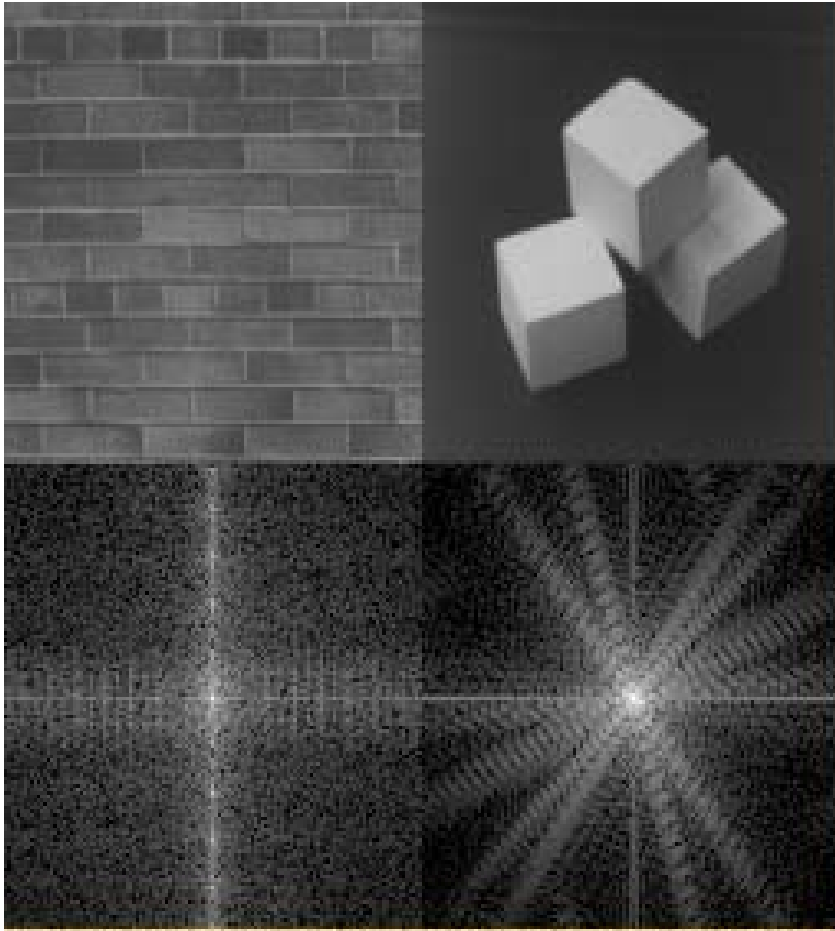


Modèles de représentation de la forme (4/5)



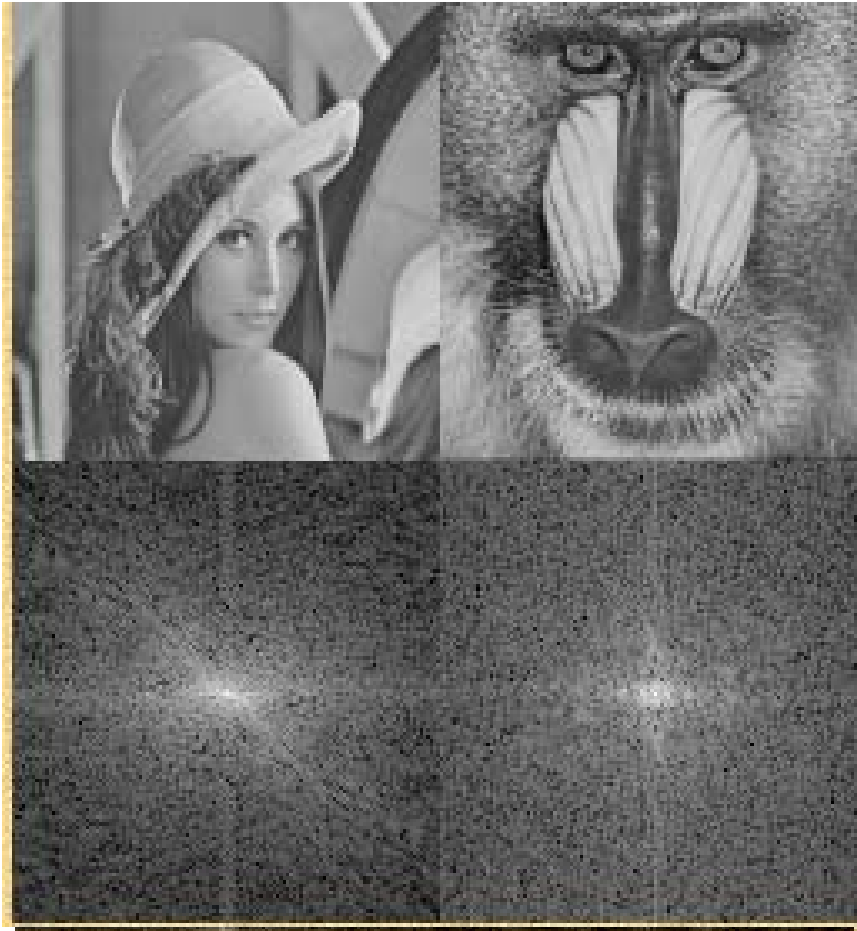
Le mur de l'image de gauche est composé de briques réparties avec une certaine périodicité selon les composantes horizontales et verticales ; aux différentes couches de briques (en verticale) correspond dans la transformée de Fourier du dessous la ligne verticale, qui est une ligne dominante. Aux différents espacements de briques (en horizontale) correspond dans la transformée de Fourier la ligne horizontale. Les périodicités verticales et horizontales des briques se retrouvent dans la transformée de Fourier sous forme de points ou de raies, régulièrement espacés, qui strient les lignes de la transformée.

Modèles de représentation de la forme (4/5)



Les cubes de l'image de gauche laissent apparaître des arêtes lumineuses (les cubes se détachent sur le fond). A chacune de ces arêtes correspond sur la transformée de Fourier une ligne qui lui est perpendiculaire. On peut remarquer que les lignes de la transformée vont facilement vers les hautes fréquences, ce qui traduit la force des contrastes des arêtes de cubes.

Modèles de représentation de la forme (4/5)



Modèles de représentation de la forme (4/5)



« La transformée de Fourier de l'image de gauche contient une ligne transversale qui correspond probablement au chapeau qui barre l'écran et qui contraste nettement avec les cheveux et avec la plume ».

« La transformée de l'image du babouin est plus claire, ce qui traduit une présence plus importante de hautes fréquences ».

Modèles de représentation de la forme (5/5)

Codage par descripteurs de Fourier

courbes paramétriques $x(t)$ et $y(t)$ d'un contour fermé

décomposition en
série de Fourier

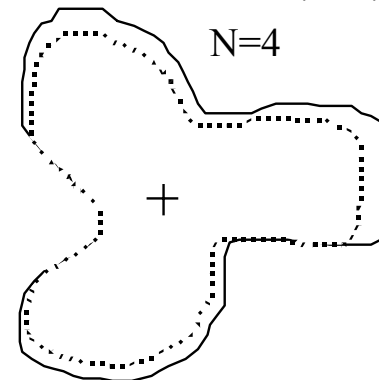
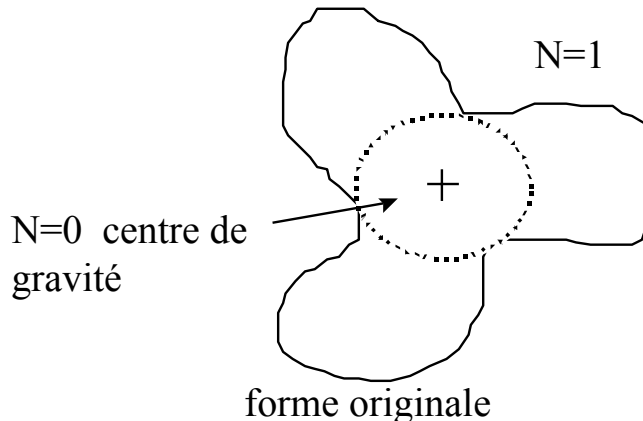
coefficients de Fourier
réels sur N harmoniques



$$x(t) = A_0 + \frac{1}{T} \sum_{n=1}^N a_n \cos(2\pi \frac{n}{T} t) + b_n \sin(2\pi \frac{n}{T} t)$$

$$y(t) = C_0 + \frac{1}{T} \sum_{n=1}^N c_n \cos(2\pi \frac{n}{T} t) + d_n \sin(2\pi \frac{n}{T} t)$$

Caractérisation d'une forme par les N premiers descripteurs de Fourier (a_n, b_n, c_n, d_n)



Références

Informations sur la couleur :

<http://www-prima.imag.fr/Prima/jlc/Courses/2000/DEA.VO/DEA-IVR.VO.S3.pdf>

Informations sur la forme :

<http://www-prima.imag.fr/Prima/jlc/Courses/2000/DEA.VO/DEA-IVR.VO.S4.pdf>

Texture

- **Motif visuel composé de pixels (texture de bois, etc.)**



Tapis tissé



Granit



Papier kraft

- **Représentation par les composantes telles que *contrastes, grosseur, direction ...***

Modèles de représentation de la texture

Texture stochastique

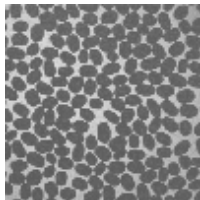
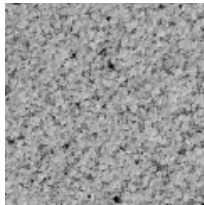


Texture structurée

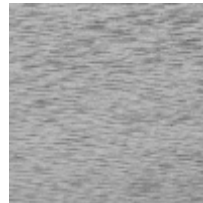
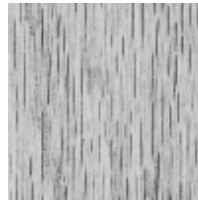


Caractéristiques qualitatives d'une texture :

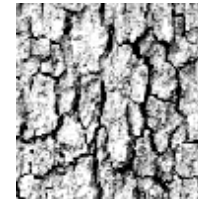
granularité



directionnalité



contraste



Méthodes d'analyse de texture : dans le domaine spatial (moments statistiques, filtres...), dans le domaine fréquentiel (Fourier...)

Caractéristiques des images

- Raimondo Schettini, Gianluigi CIOCCA, Silvia ZUFFI. *A Survey Of Methods For Colour Image Indexing And Retrieval In Image Databases.*— Exploiting Digital Media, John Wiley, 2001
<http://citeseer.nj.nec.com/489355.html>
- B.S. Manjunath and W-Y. Ma. *Image Databases : Search and Retrieval of Digital Imagery, chapter 12 - Texture Features for Image Retrieval*, pages 313--344. Wiley Inter-Science, 2002. V. Castelli and L.D. Bergman (Eds) - ISBN: 0-471-32116-8
- B.B. Kimia. *Image Databases : Search and Retrieval of Digital Imagery, chapter 13 - Shape Representation for Image Retrieval*, pages 345--372. Wiley Inter-Science, 2002. V. Castelli and L.D. Bergman (Eds) - ISBN: 0-471-32116-8

Représentation des images (1/3)

- **Signature / descripteur visuel** : représentation numérique des caractéristiques de l'image
- Représentation d'une image par un **vecteur de caractéristiques / point dans un espace multidimensionnel**
- **Description globale et description locale** des images

Représentation des images (2/3)

- Représentation du contenu visuel des images par un descripteur
- Représentation de chaque image par un point dans un espace multidimensionnel
- Similarité des images définie comme étant proportionnelle à la proximité des points dans l'espace

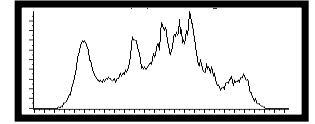
Représentation des images (2/3)



- Représentation du contenu visuel des images par un descripteur
- Représentation de chaque image par un point dans un espace multidimensionnel
- Similarité des images définie comme étant proportionnelle à la proximité des points dans l'espace

Représentation des images (2/3)

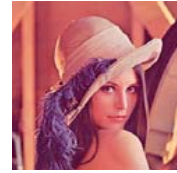
- Représentation du contenu visuel des images par un descripteur
- Représentation de chaque image par un point dans un espace multidimensionnel
- Similarité des images définie comme étant proportionnelle à la proximité des points dans l'espace



Histogramme de couleurs

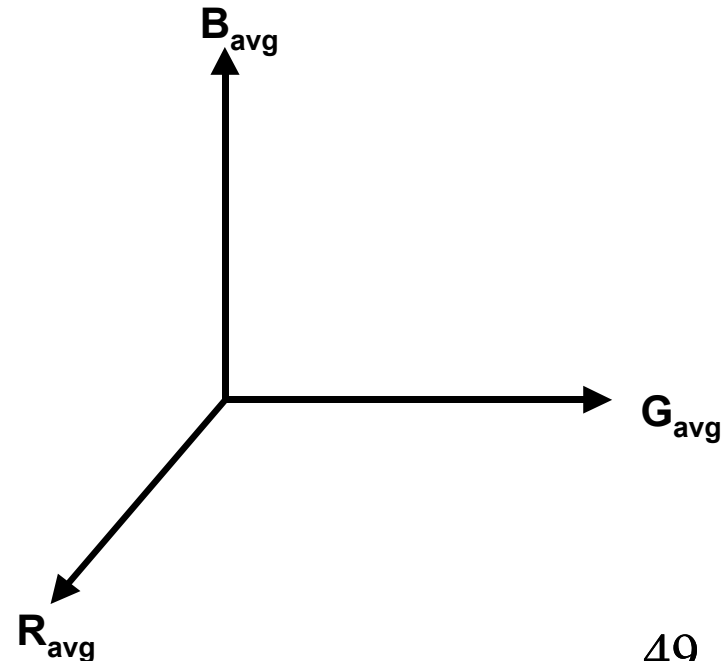
Représentation des images (2/3)

- Représentation du contenu visuel des images par un descripteur
- Représentation de chaque image par un point dans un espace multidimensionnel
- Similarité des images définie comme étant proportionnelle à la proximité des points dans l'espace



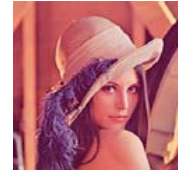
(180, 98, 105)

Vecteur moyenne des
couleurs



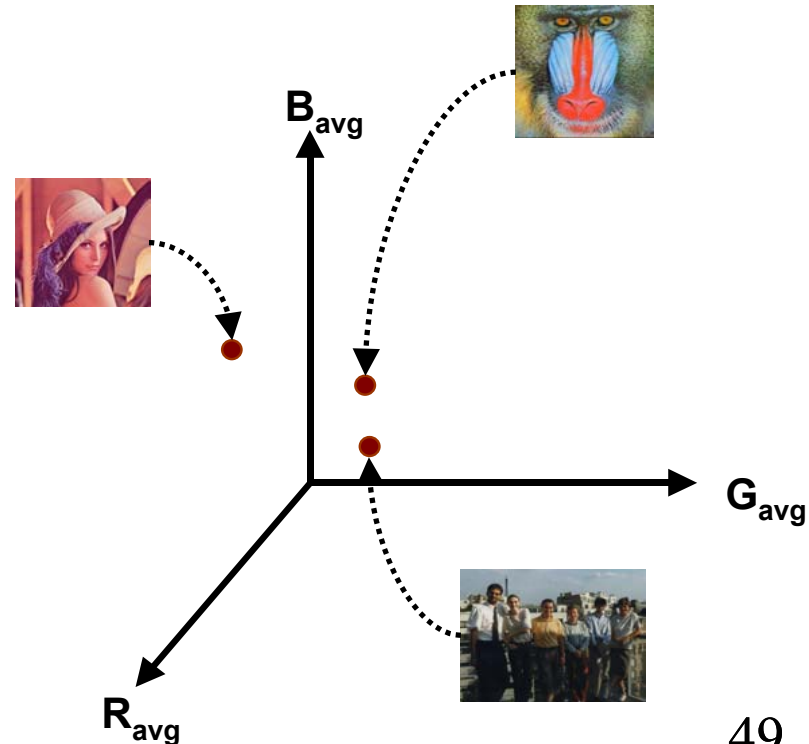
Représentation des images (2/3)

- Représentation du contenu visuel des images par un descripteur
- Représentation de chaque image par un point dans un espace multidimensionnel
- Similarité des images définie comme étant proportionnelle à la proximité des points dans l'espace



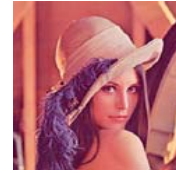
(180, 98, 105)

Vecteur moyenne des couleurs



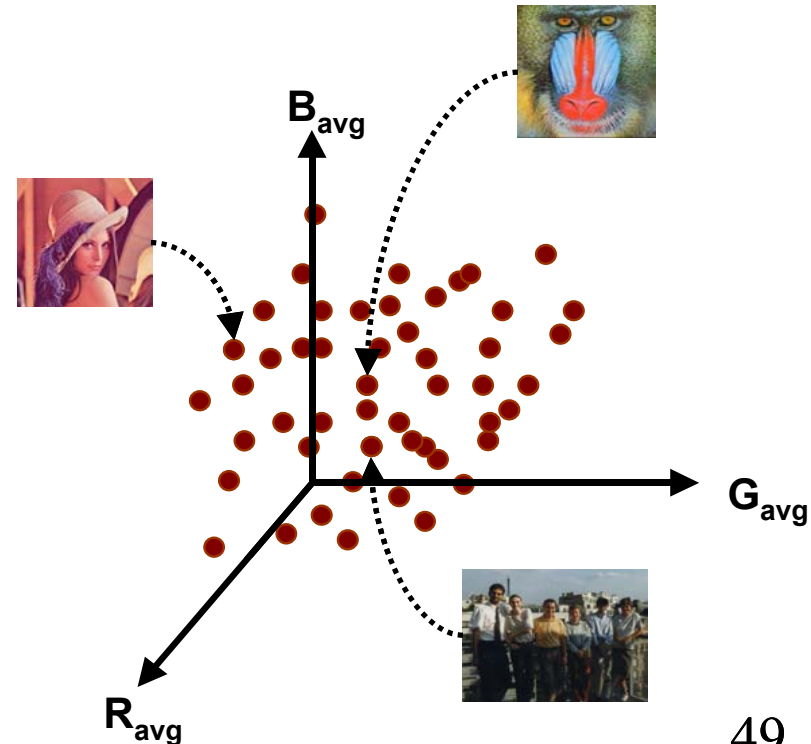
Représentation des images (2/3)

- Représentation du contenu visuel des images par un descripteur
- Représentation de chaque image par un point dans un espace multidimensionnel
- Similarité des images définie comme étant proportionnelle à la proximité des points dans l'espace



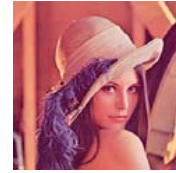
(180, 98, 105)

Vecteur moyenne des couleurs



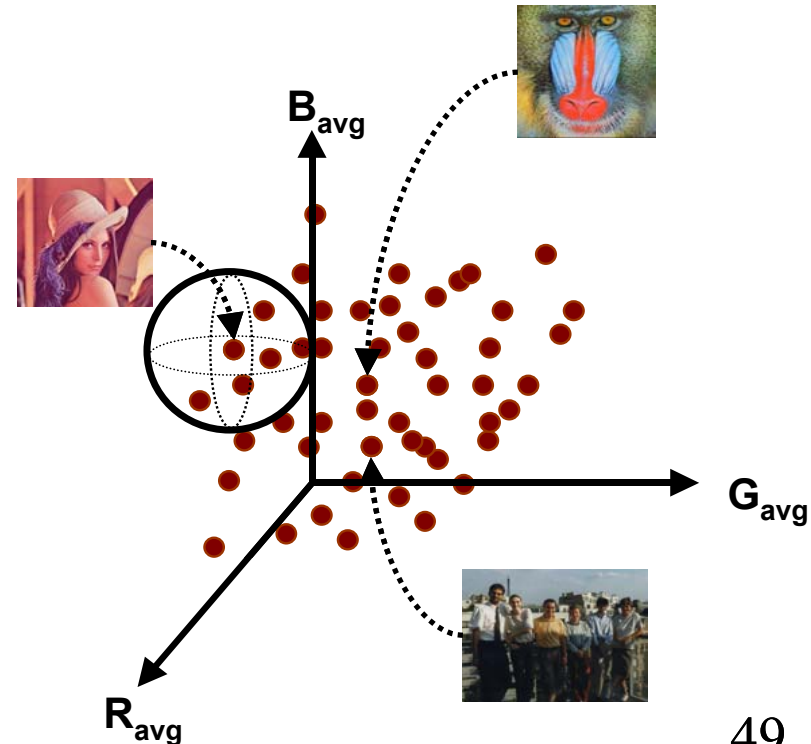
Représentation des images (2/3)

- Représentation du contenu visuel des images par un descripteur
- Représentation de chaque image par un point dans un espace multidimensionnel
- Similarité des images définie comme étant proportionnelle à la proximité des points dans l'espace



(180, 98, 105)

Vecteur moyenne des couleurs



Description locale de l'image

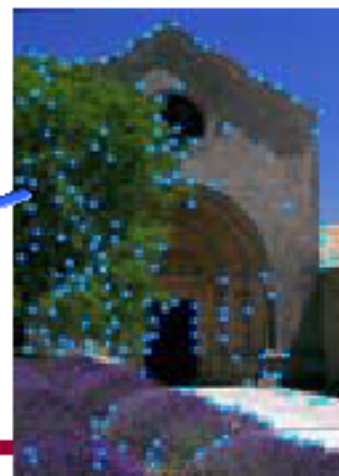
Segmentation en régions

Requête à base de régions



Extraction de points d'intérêt

Requêtes à base de points



$$R = \begin{pmatrix} - \\ - \\ \vdots \\ - \end{pmatrix}$$

$$P = \begin{pmatrix} - \\ - \\ \vdots \\ - \end{pmatrix}$$

Représentation des images (3/3)

Représentation d'une image par un ensemble d'objets d'intérêt [OTOL+97]



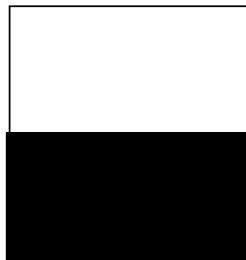
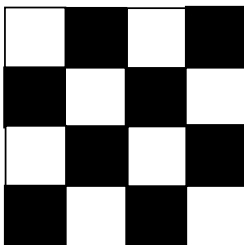
Localisation verticale des objets : [*arbre* < (*maison* : (*chien* = *soleil*))]

Localisation horizontale des objets : [(*maison* : *chien*) < *arbre* < *soleil*]

Interrogation des images (1/5)

Interrogation par similarité :

- Deux images sont similaires si leur **distance** est égale à ε
- La distance se calcule par rapport à certaines caractéristiques (couleur, texture ...) et par rapport à certaines signatures



Deux images ayant le même histogramme de couleurs

Distances entre images

$$d : I \times I \rightarrow \mathbb{R}$$

Vérifiant $\forall i, j, k$ trois images :

- $d(i, j) \geq 0$
- $d(i, j) = d(j, i)$ symétrie
- $d(i, i) = 0$ réflexivité
- $d(i, j) \leq d(i, k) + d(k, j)$ inégalité triangulaire

Distances entre images

Distance de la forme Minkowski L_p

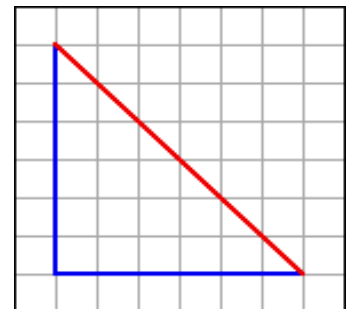
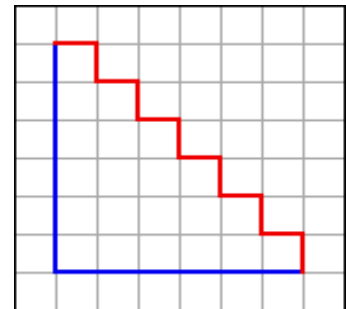
$$X(x_1, x_2, \dots, x_n) \quad Y(y_1, y_2, \dots, y_n)$$

$$d(I, J) = \left(\sum_i |x_i - y_i|^p \right)^{1/p}$$

L_1 : distance cityblock,
ou distance de Manhattan

L_2 : distance Euclidienne

L_∞ : distance Maximum



Similarité d'images

- Point de vue utilisateur :

similarité par rapport à une caractéristique

- Point de vue système :

distance entre vecteurs de caractéristique

Similarité : fonction décroissante de la distance

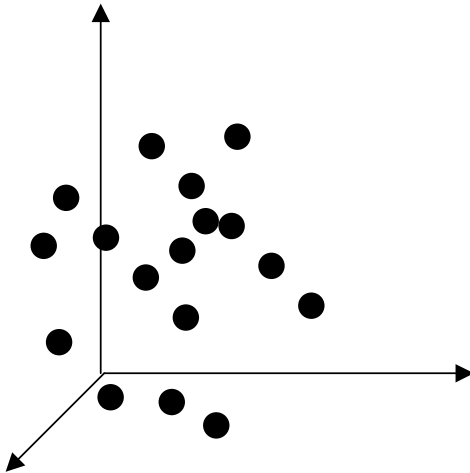
$$\text{similarité} = 1 - \frac{d}{d_{\max}}$$

avec d la distance entre deux images

et $d_{\max}$ la distance maximale entre deux images

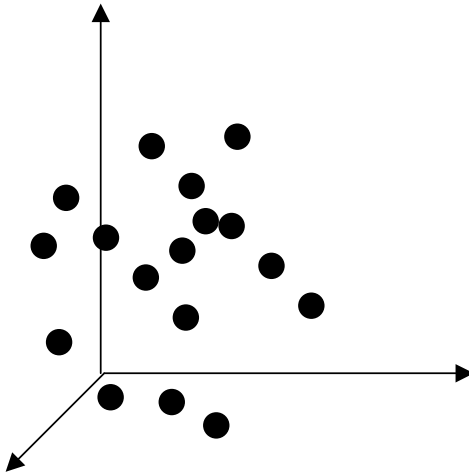
Recherche d'images

Recherche d'images

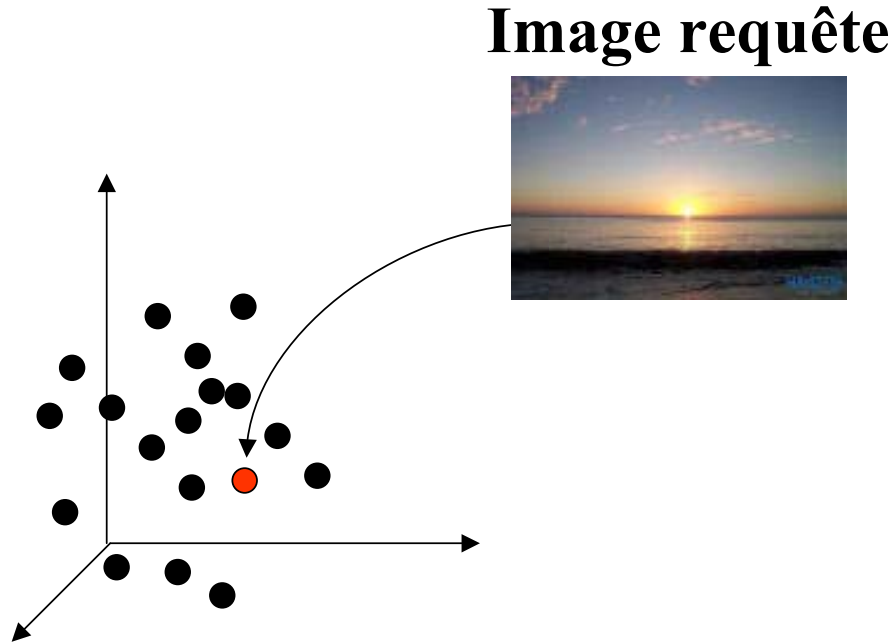


Recherche d'images

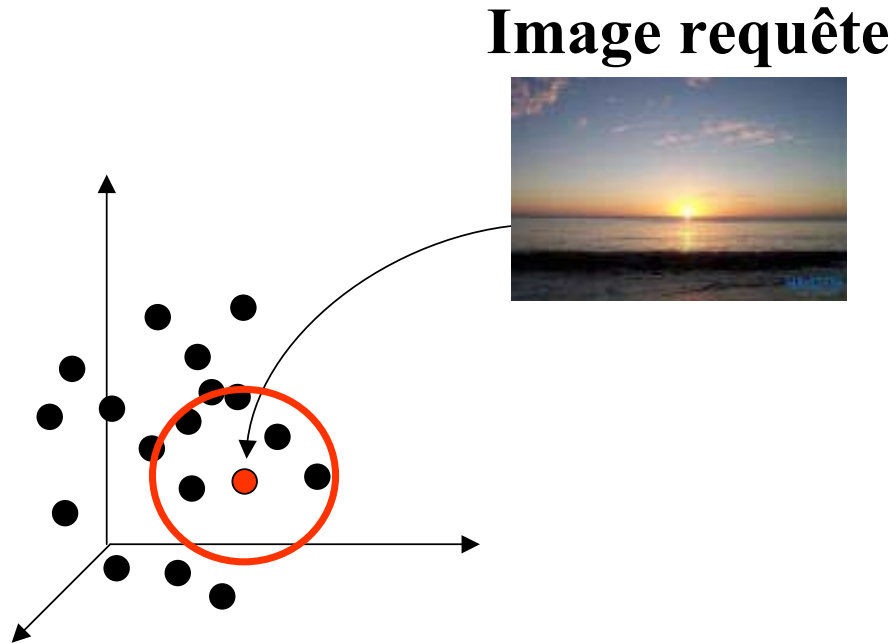
Image requête



Recherche d'images



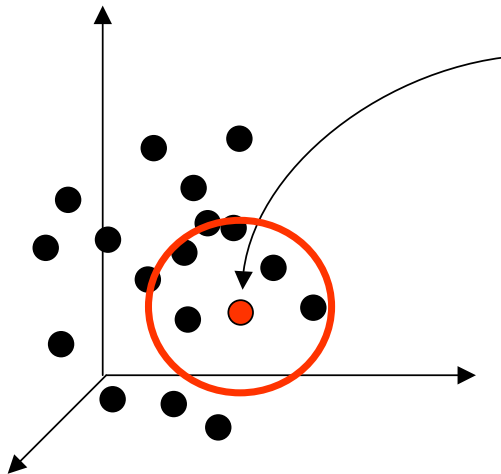
Recherche d'images



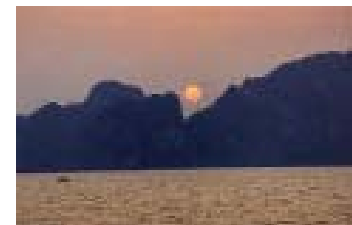
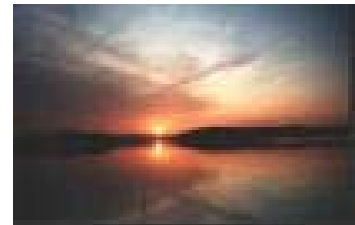
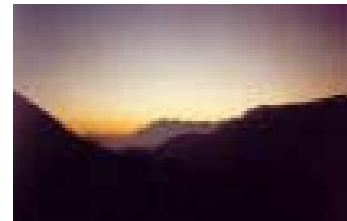
$$\{ \text{image } i / d(i,q) < \varepsilon \}$$

Recherche d'images

Image requête



$$\{ \text{image } i / d(i,q) < \varepsilon \}$$



Images résultat

Requêtes sur les images

B : une base d'images

i : une image

q : une image requête

Q : le résultat de
la requête

s : un seuil

■ Requêtes d'intervalle

$$Q = \{ i / d(q, i) \leq s, i \in B \}$$

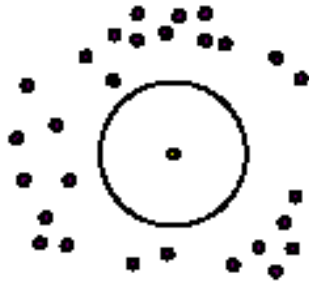
■ Requêtes de voisinage

$$\forall i \in Q, \forall j \in B, j \notin Q \Rightarrow d(q, i) \leq d(q, j)$$

avec $\|Q\| = k$

Requêtes sur les images

■ Requêtes d'intervalle



no results



too many results

■ Requêtes de voisinage

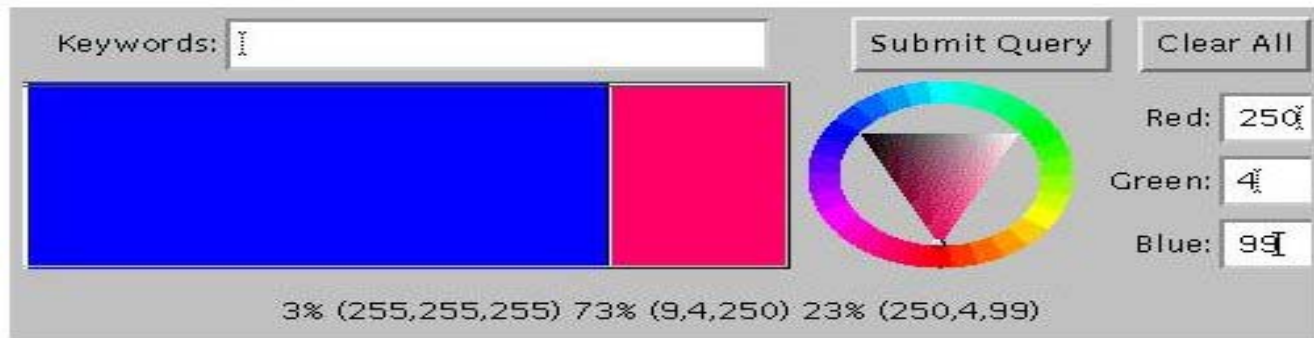


k -nearest neighbors: $r_q(\mathcal{T}_k)$

Interrogation des images (2/5)

Interrogation sur une caractéristique

« je veux des images contenant 3% de blanc, 73% de bleu et 23% de rouge »



Keywords:

Submit Query Clear All

Red: 250
Green: 4
Blue: 99

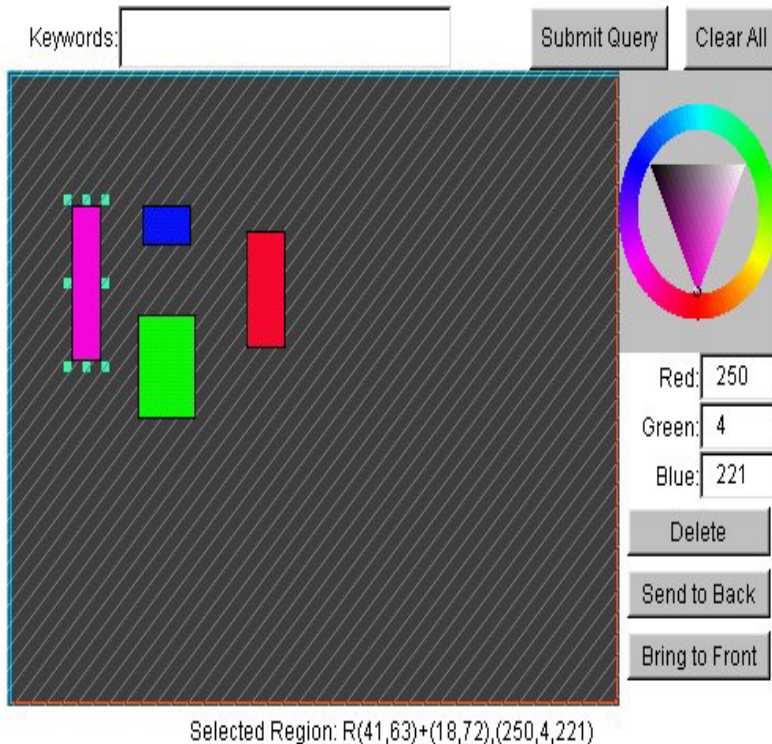
3% (255,255,255) 73% (9,4,250) 23% (250,4,99)



Interrogation des images (3/5)

Interrogation la localisation des caractéristiques

« je veux des images la couleur bleue est située en haut à droite »

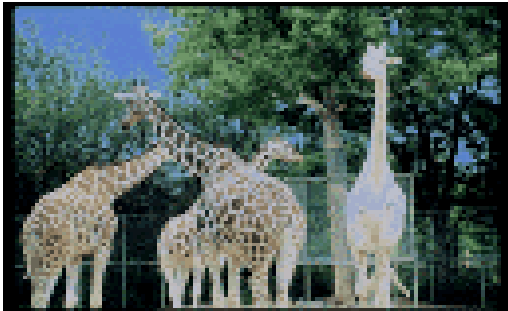


www.qbic.almaden.ibm.com

Interrogation des images (4/5)

Interrogation par l'exemple

« je veux des images dont la texture est similaire à cette image »



Distance entre les histogrammes
de couleur des deux images =
0.018



Distance entre la texture des
deux images = 0.03

<http://www.ctr.columbia.edu/~jrsmith/>
(VisualSEEK)

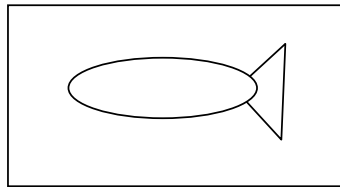
Interrogation des images (5/5)

- **Interrogation sur plusieurs caractéristiques**

« je veux des images contenant 50% de rouge, 25% de texture de granit »

- **Interrogation par croquis**

« je veux des images contenant un objet de la forme : »



- **Interrogation par objets**

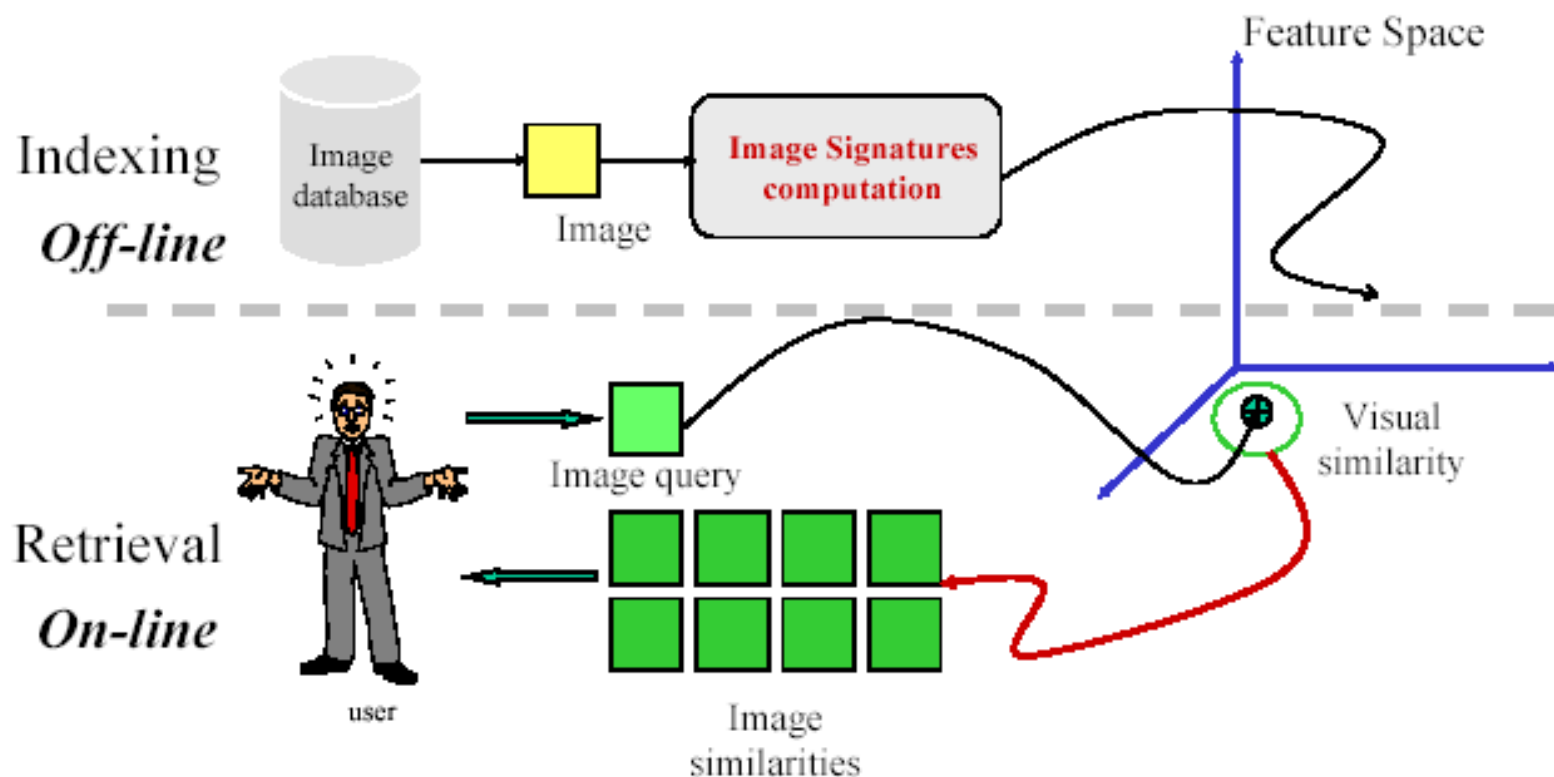
« je veux des images contenant une voiture rouge »

« je veux des images où Chirac est à côté de Clinton »

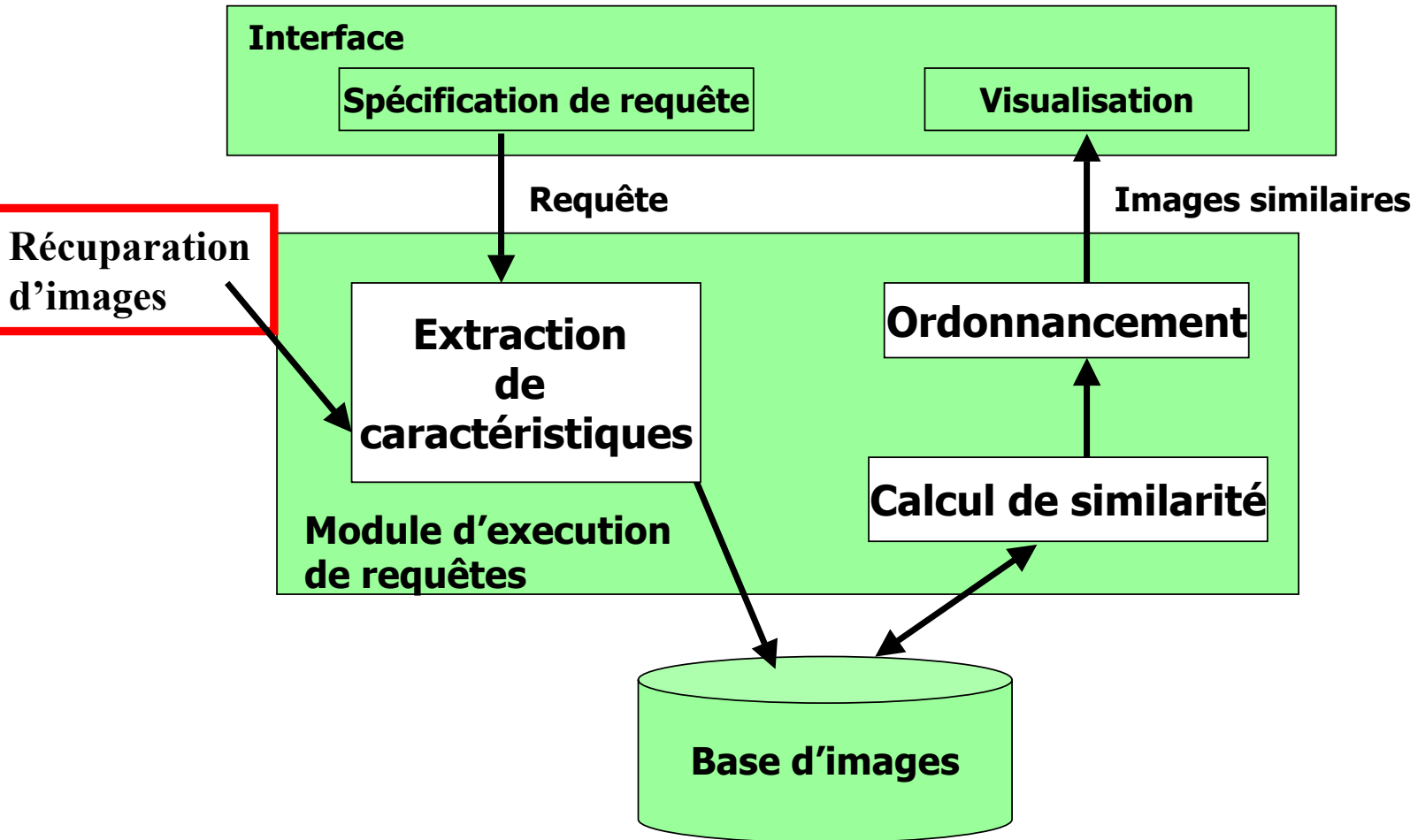
- **Interrogation par méta-données**

« je veux des images de résolution 300dpi , créées en 1999 »

Architecture Générale



Recherche d'images par le contenu : création de la base



Exemple de ce qui est proposé dans Oracle

Classe **ORDVir**

image ORDImage signature RAW(2000)

Principales différences entre les images ORDImage et **ORDVir** :

- permet l'indexation des images au travers de leur signature
- permet des recherches basées sur le contenu des images, (couleur, structure et texture).

La classe ORDVIR d'Oracle

méthode `analyze()` : génère les signatures des images
(couleurs locales et globales, structure et texture)

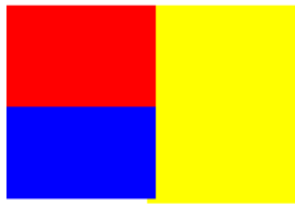


Image 1

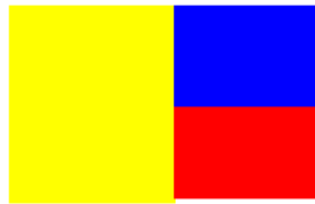


Image 2

Globalement les images ont les mêmes couleurs mais localement les couleurs ne sont pas à la même position (utilisation d'un arbre quaternaire pour comparer les couleurs locales)



Image 1

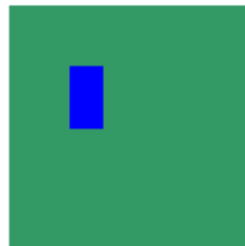


Image 2

Les structures des images sont similaires si on ne tient pas compte de la couleur

La classe ORDVIR d'Oracle

méthode `VIRSimilar` :
retourne vraie (1) si une image
requête est similaire à une
image source ou faux (0) sinon

```
VIRSimilar ( signature IN RAW,  
quersignature IN RAW,  
weightstring IN VARCHAR2,  
threshold IN FLOAT  
[referencetoScore IN NUMBER])  
;
```

- **signature** : signature de l'image (source) qui va être comparée.
- **quersignature** : signature de l'image à comparer
- **weightstring** poids associés aux différentes caractéristiques
('globalcolor="val"
localcolor="val" texture="val"
structure="val" ')
- **threshold** : seuil auquel le poids additionné de l'image doit correspondre
- **referencetoScore** : paramètre optionnel utilisé lorsqu'on utilise l'opérateur `VIRScore` dans la requête.

La classe ORDVIR d'Oracle

Exemple d'utilisation de la méthode `VIRSimilar` :

Seuil de 25, poids pour la couleur globale 20, pour la couleur locale 20 etc.

```
DECLARE
  t_img ORDSYS.ORDVir ;
  i INTEGER ;
  query_signature RAW(2000) ;
BEGIN
  SELECT id, img INTO i, t_img FROM t WHERE
    ORDSYS.VIRSimilar(t_img.signature, query_signature,
      'globalcolor="20" localcolor="20"
        texture="10" structure="50"', 25 ) = 1 ;
END ;
```

La classe ORDVIR d'Oracle

opérateur VIRScore :
retourne un score en fonction
de la similarité d'une image
source et d'une image requête

```
VIRScore(referencetoSimilar  
IN NUMBER) ;
```

Si deux images sont
similaires le score
est de 0 .

Si deux images sont
complètement
différentes, le score
est de 100.

Cet opérateur s'utilise avec VIRSimilar. Le paramètre **referencetoSimilar** permet de faire le lien entre l'invocation de VIRSimilar et l'invocation de VIRScore correspondante.

La classe ORDVIR d'Oracle

Exemple d'utilisation de la
méthode VIRScore :

DECLARE

img_score NUMBER ;

i INTEGER ;

query_signature RAW(2000) ;

*Pour chaque identificateur
d'image on aura un score
compris entre 0 (similaire)
et 100 (non similaire)*

BEGIN

SELECT photo_id, ORDSYS.VIRScore(123) INTO i, img_score

FROM T WHERE

ORDSYS.VIRSimilar(img.signature, query_signature,

'globalcolor="0.2" localcolor="0.2"

texture="0.1" structure="0.5", 50, 123) = 1 ;

END ;

Exemples de requêtes sur des systèmes en ligne

- QBIC - IBM's Query By Image Content
- IKONA – Prototype de l'INRIA Rocquencourt
- BlobWorld - Berkeley

Similarité d'images : quelques références

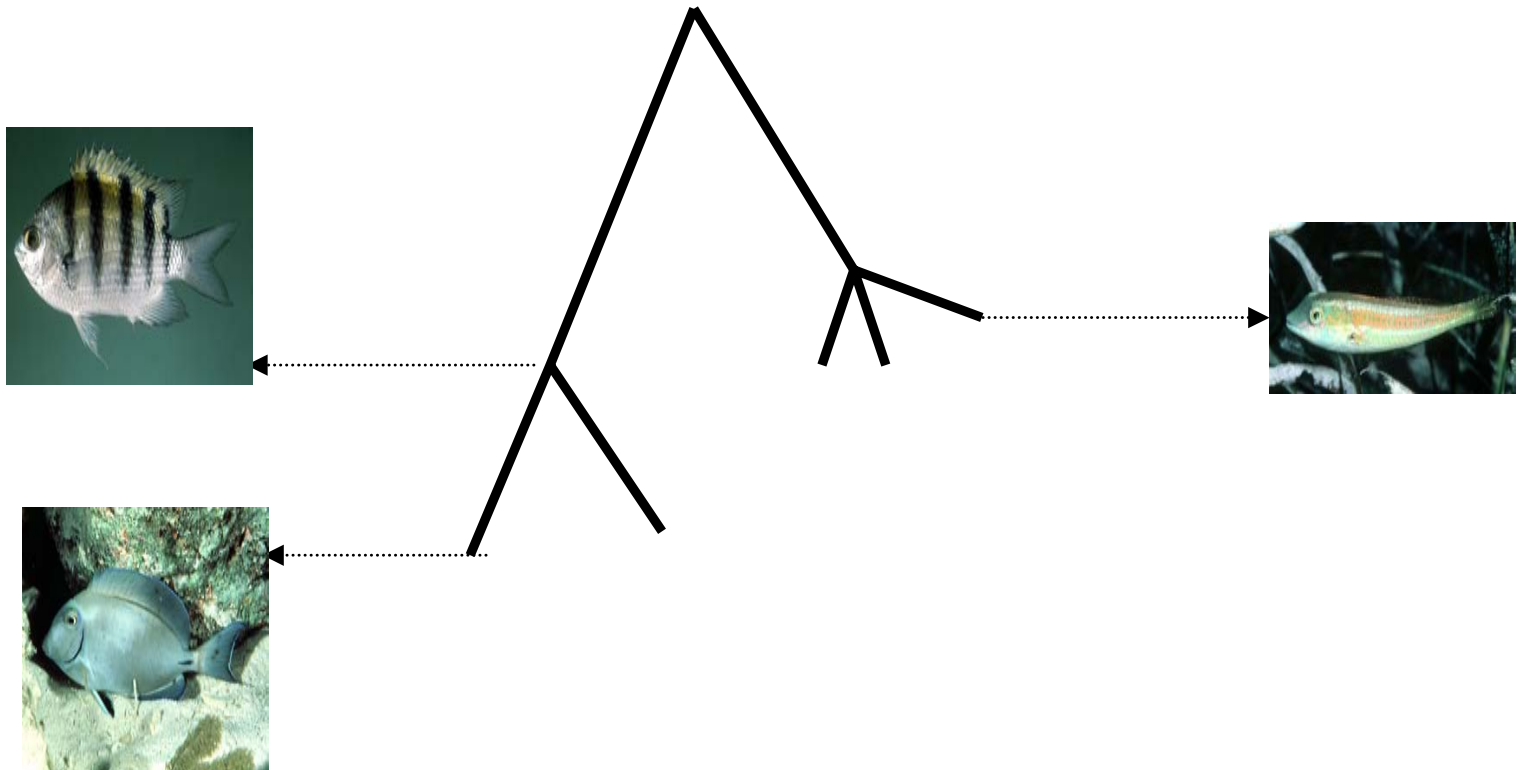
- J. Puzicha, Y. Rubner, C. Tomasi, and J.M. Buhmann. *Empirical Evaluation of Dissimilarity Measures for Color and Texture*. In Proc. of the IEEE Int. Conf. on Computer Vision (ICCV'99), pages 1165--1173, 1999
- N. Boujemaa, J. Fauqueur and V. Gouet *What's beyond query by example ? "Trends and Advances in Content-Based Image and Video Retrieval"*, L. Shapiro, H.P. Kriegel, R. Veltkamp (ed.), LNCS, Springer Verlag, 2004
- Y. Li, C.-C.J. Kuo, and X. Wan. *Image Databases: Search and Retrieval of Digital Imagery, chapter 10: Introduction to content-based image retrieval - Overview of key techniques*, pages 261--284. Wiley Inter-Science, 2002. V. Castelli and L.D. Bergman (Eds).

3. Structures multidimensionnelles

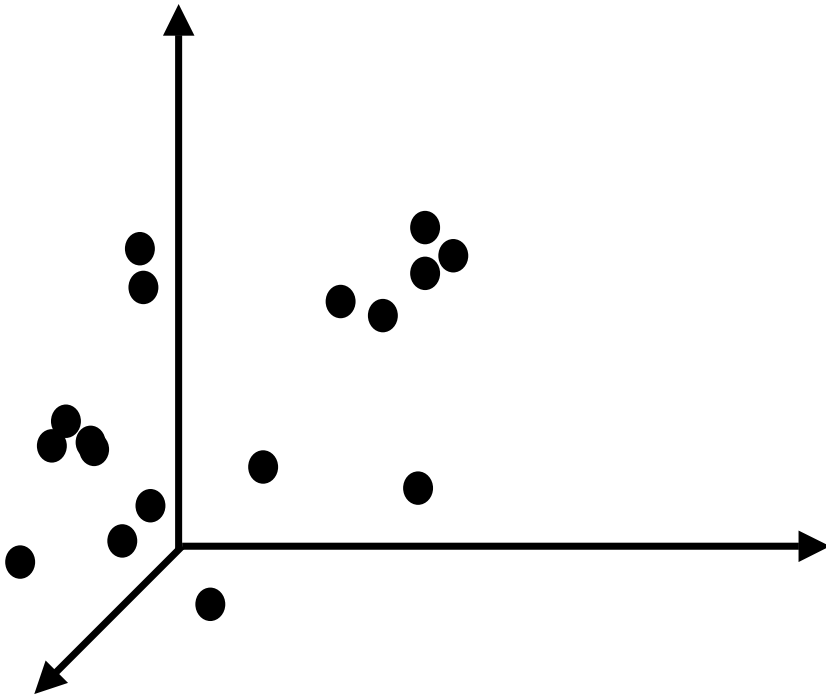
- **Arbre R et Arbre SR**
- **Arbre M**
- **Arbre Quaternaire**
- **Exemple d'approches**

Structure d'index

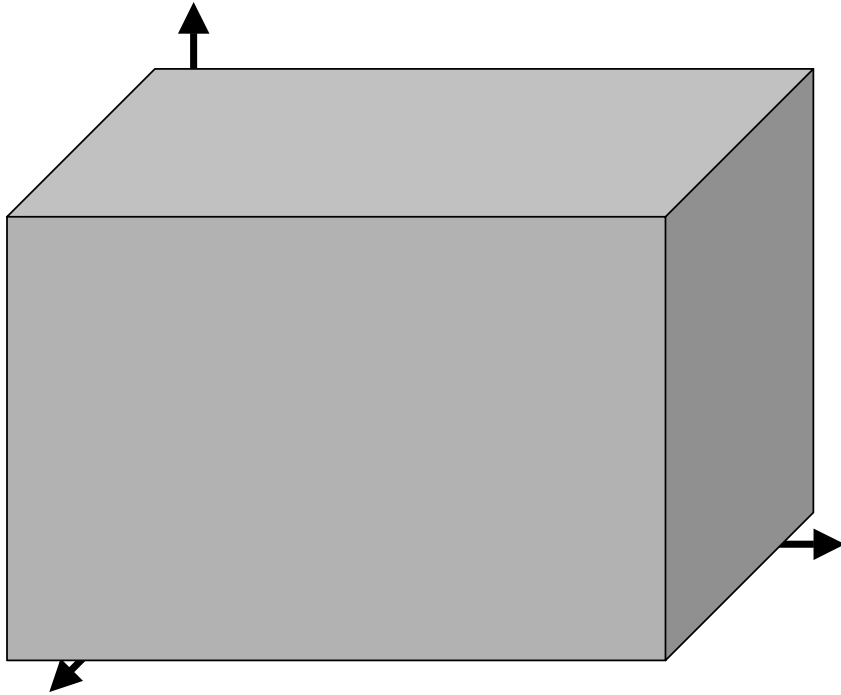
Diviser l'espace de recherche en ensembles d'images similaires



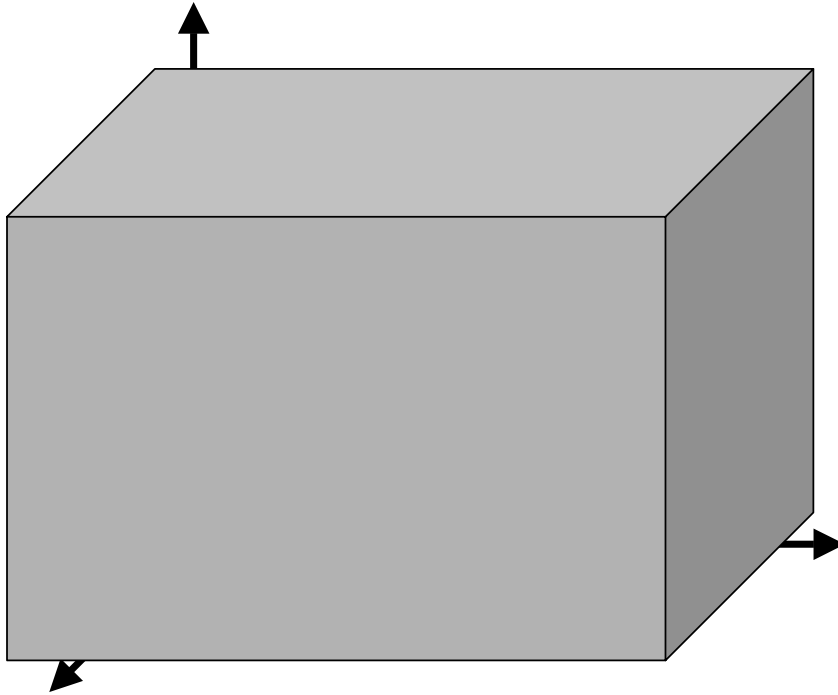
Structure d'index : arbre R



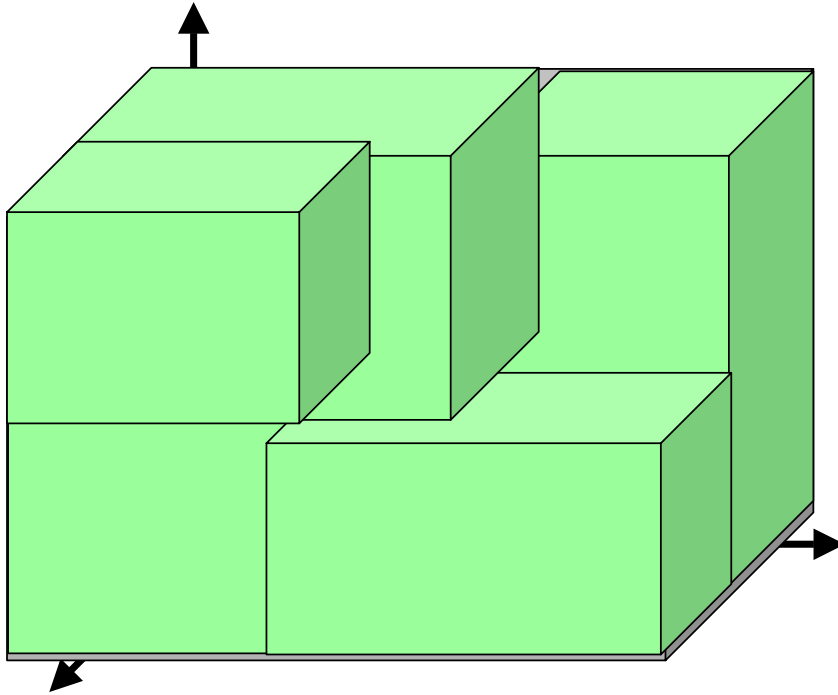
Structure d'index : arbre R



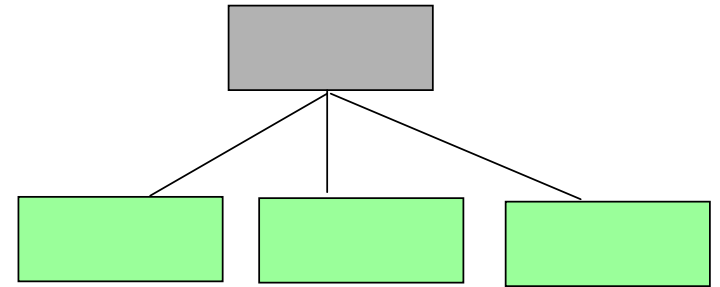
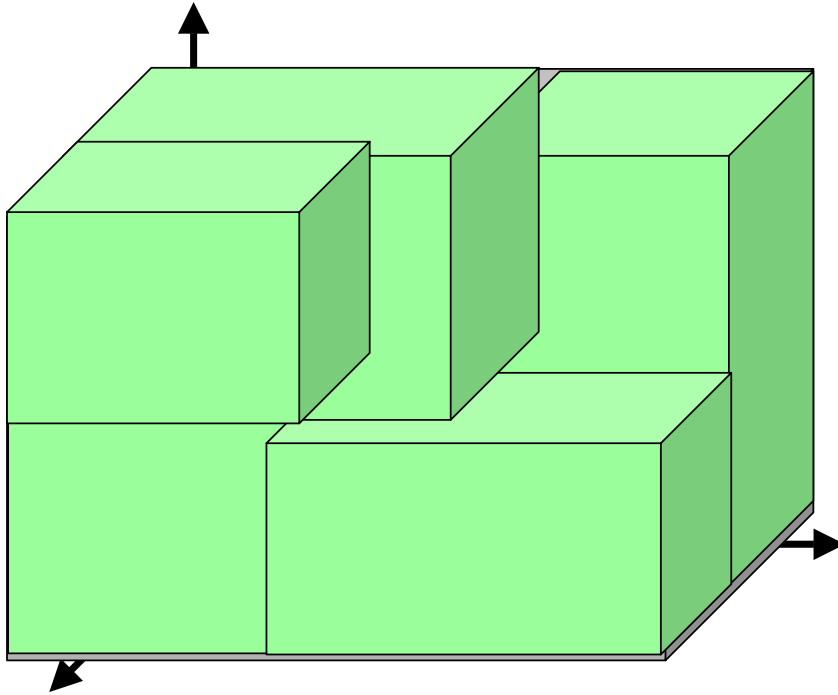
Structure d'index : arbre R



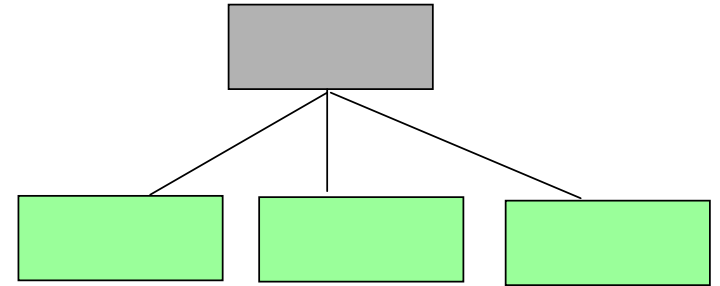
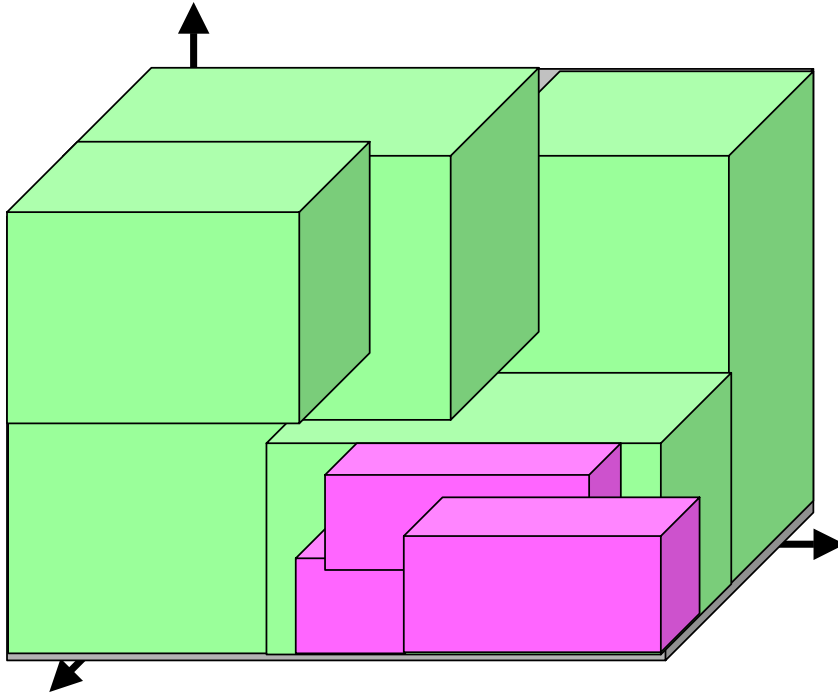
Structure d'index : arbre R



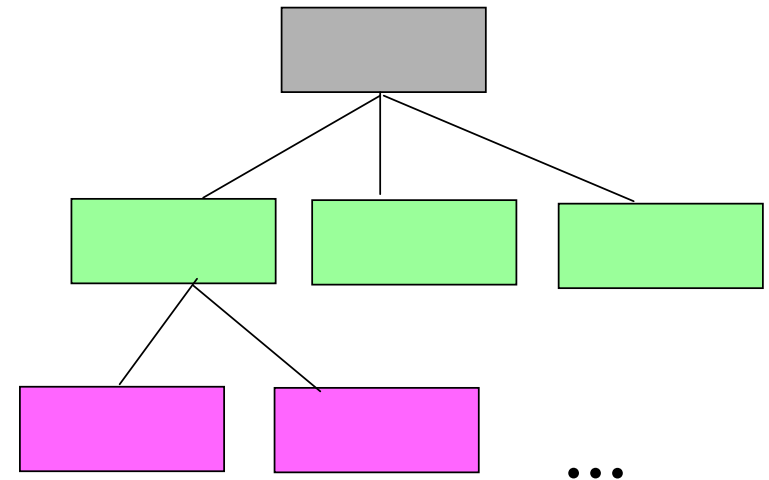
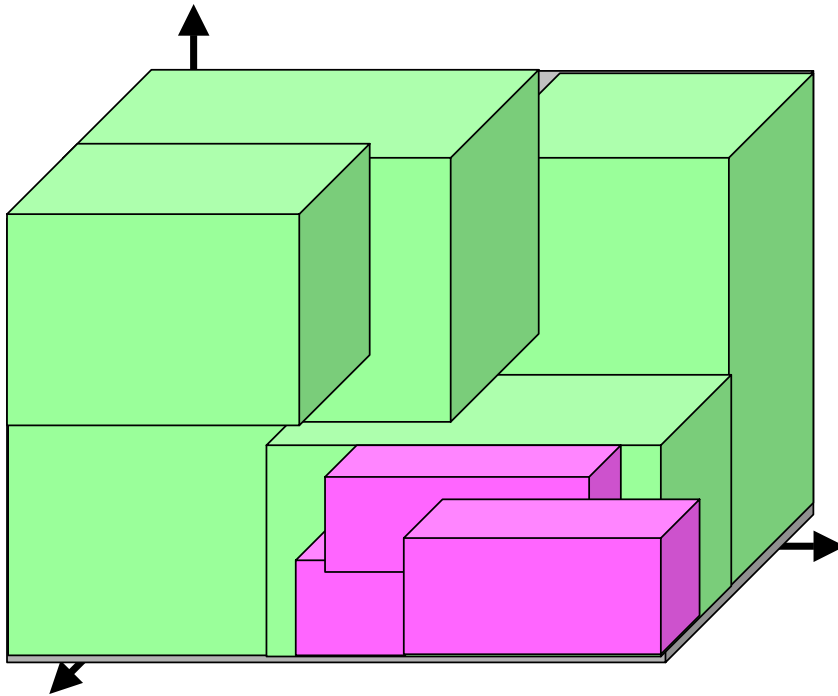
Structure d'index : arbre R



Structure d'index : arbre R

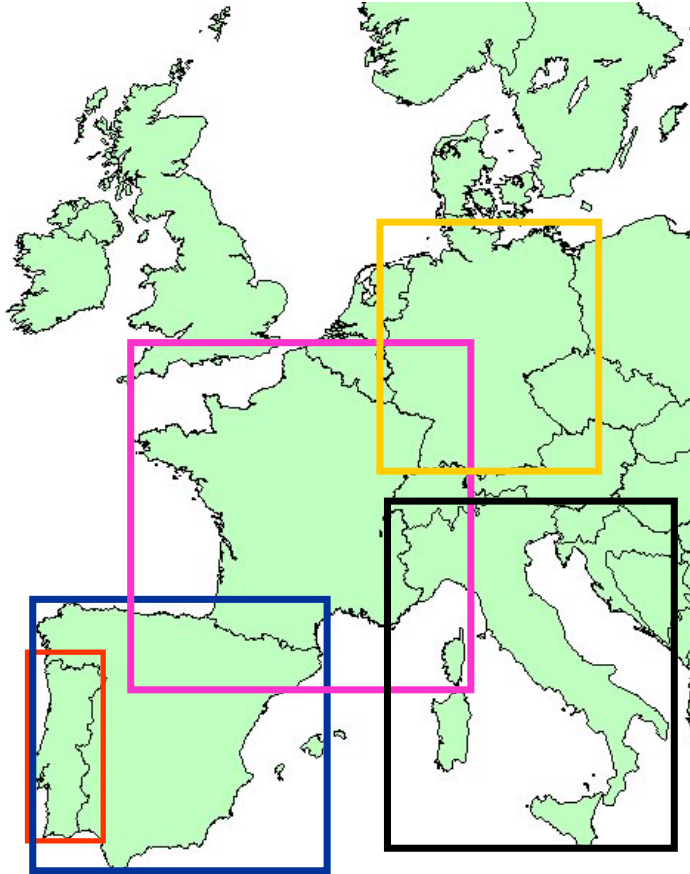


Structure d'index : arbre R



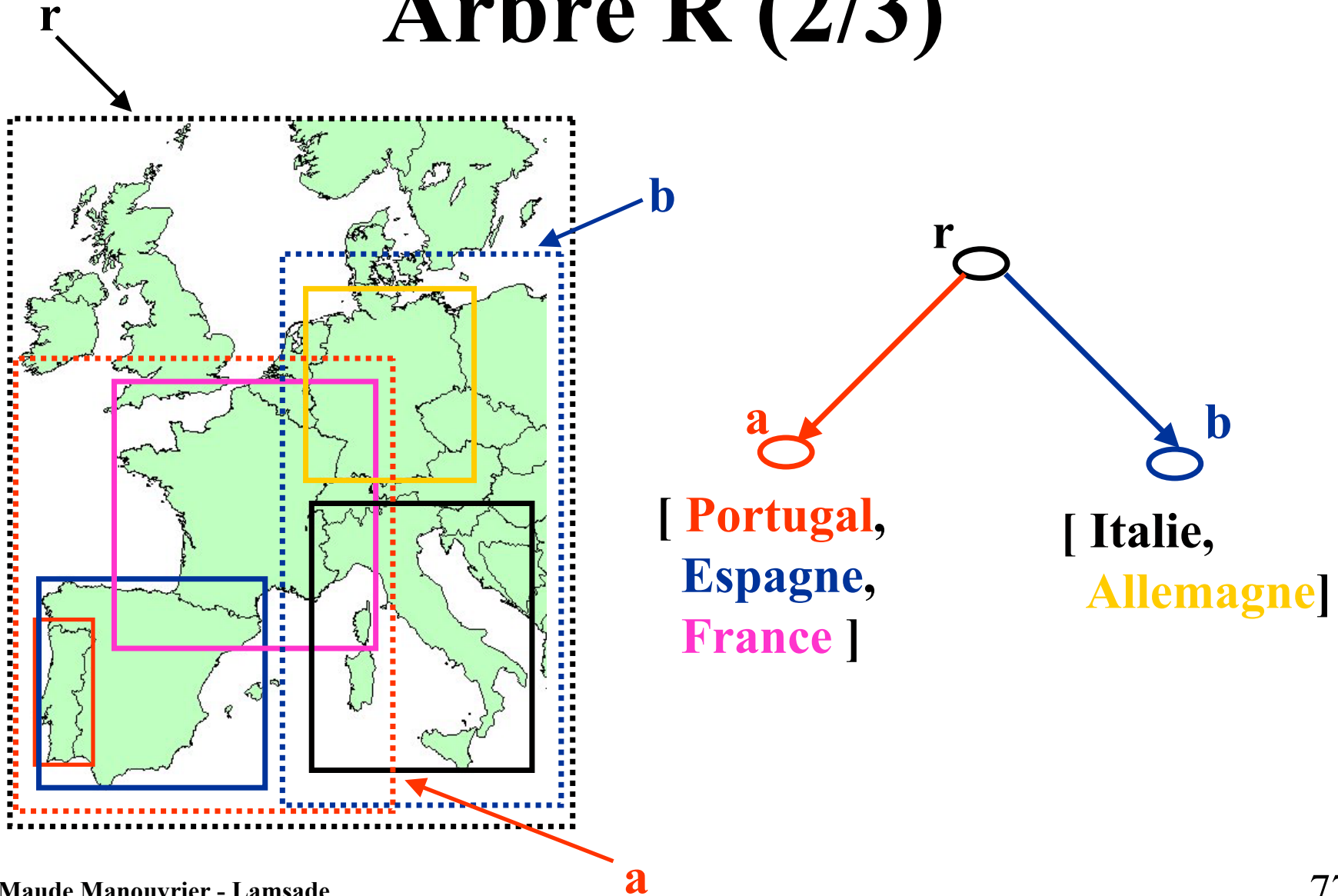
Arbre R (1/3) [Gut84]

Idée de base : indexation d'objets spatiaux

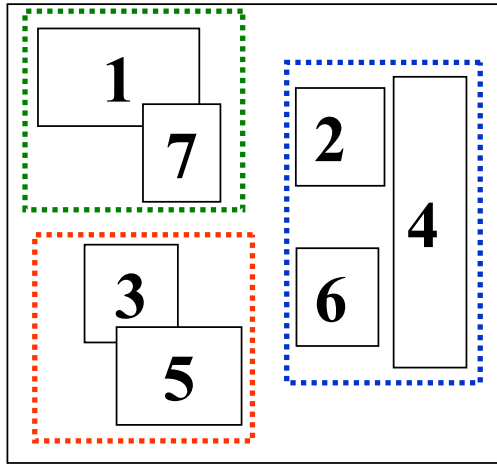


- représentation des objets spatiaux par des **rectangles englobants minimums**
- chaque nœud de l'arbre R représente un **rem** dans l'espace étudié
- Les rectangles englobants minimums peuvent se chevaucher

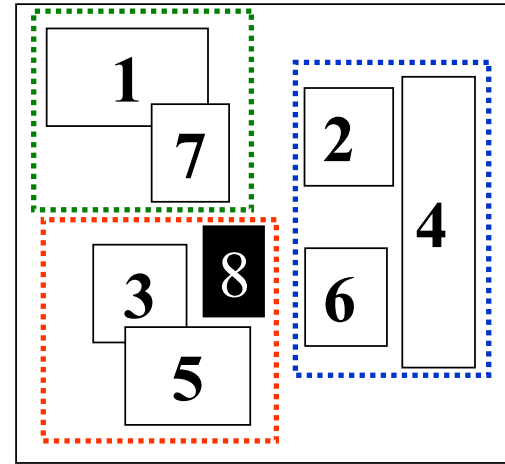
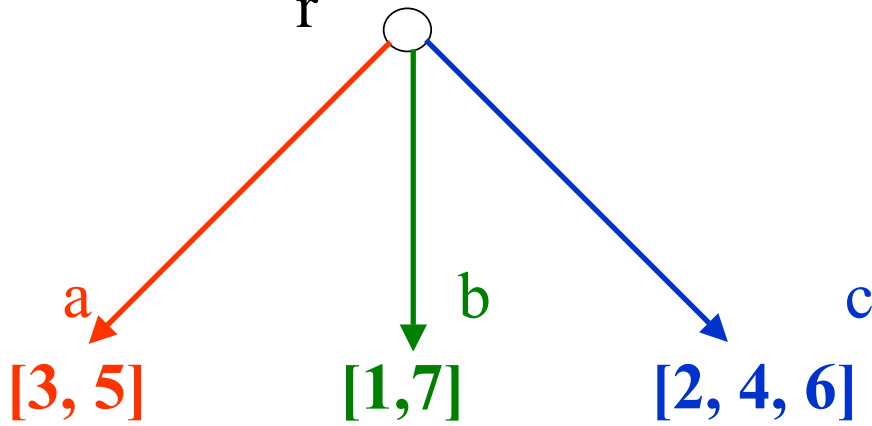
Arbre R (2/3)



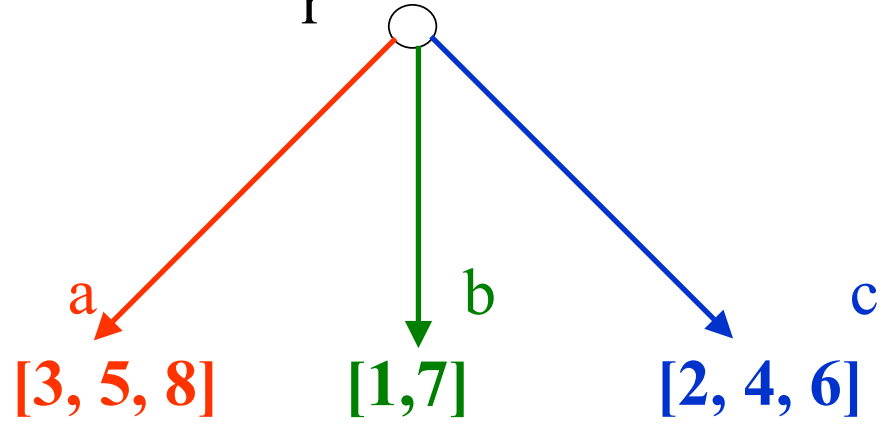
Arbre R (3/3)



r



r



Arbre R appliqué aux images

**Les feuilles de
l'index contiennent
les vecteurs de
caractéristiques des
images**

Capacité du rectangle= 4

Arbre R appliqué aux images

Les feuilles de
l'index contiennent
les vecteurs de
caractéristiques des
images



Capacité du rectangle= 4

Arbre R appliqué aux images

•
p1

Les feuilles de
l'index contiennent
les vecteurs de
caractéristiques des
images

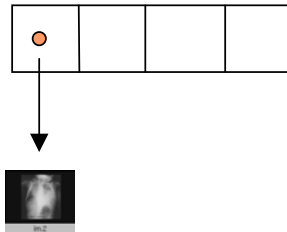


Capacité du rectangle= 4

Arbre R appliqué aux images

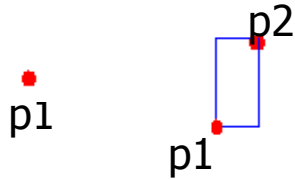
•
p1

**Les feuilles de
l'index contiennent
les vecteurs de
caractéristiques des
images**

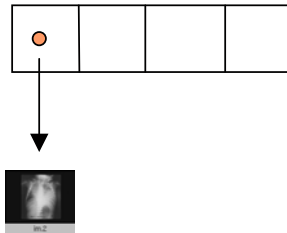


Capacité du rectangle= 4

Arbre R appliqué aux images

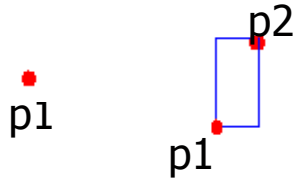


Les feuilles de l'index contiennent les vecteurs de caractéristiques des images

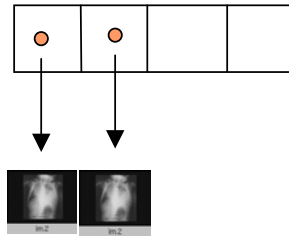


Capacité du rectangle= 4

Arbre R appliqué aux images

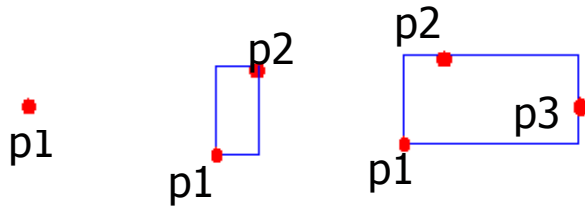


Les feuilles de l'index contiennent les vecteurs de caractéristiques des images

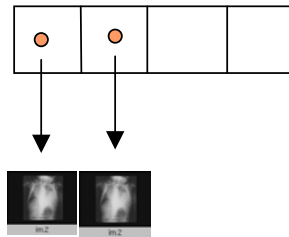


Capacité du rectangle= 4

Arbre R appliqué aux images

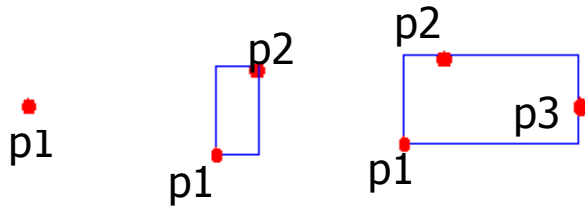


Les feuilles de l'index contiennent les vecteurs de caractéristiques des images

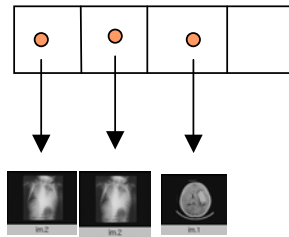


Capacité du rectangle= 4

Arbre R appliqué aux images

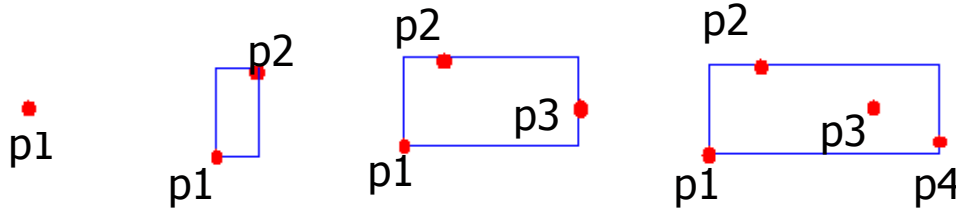


Les feuilles de l'index contiennent les vecteurs de caractéristiques des images

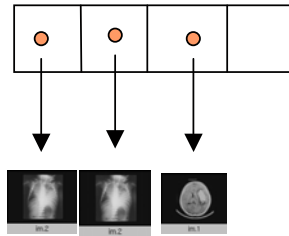


Capacité du rectangle= 4

Arbre R appliqué aux images

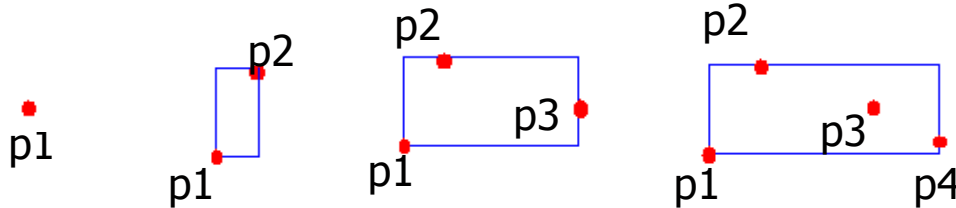


Les feuilles de l'index contiennent les vecteurs de caractéristiques des images

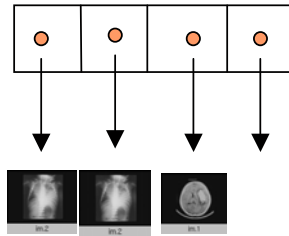


Capacité du rectangle= 4

Arbre R appliqué aux images

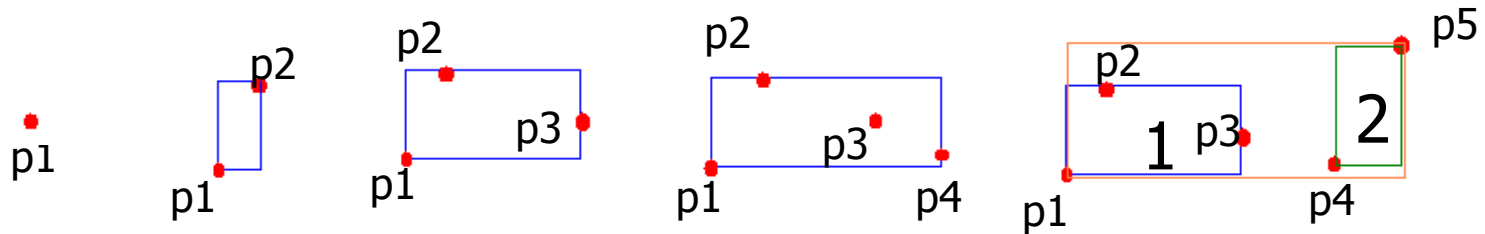


Les feuilles de l'index contiennent les vecteurs de caractéristiques des images

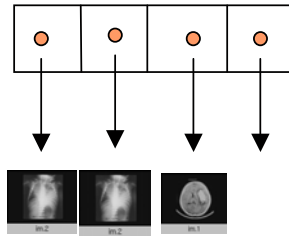


Capacité du rectangle= 4

Arbre R appliqué aux images

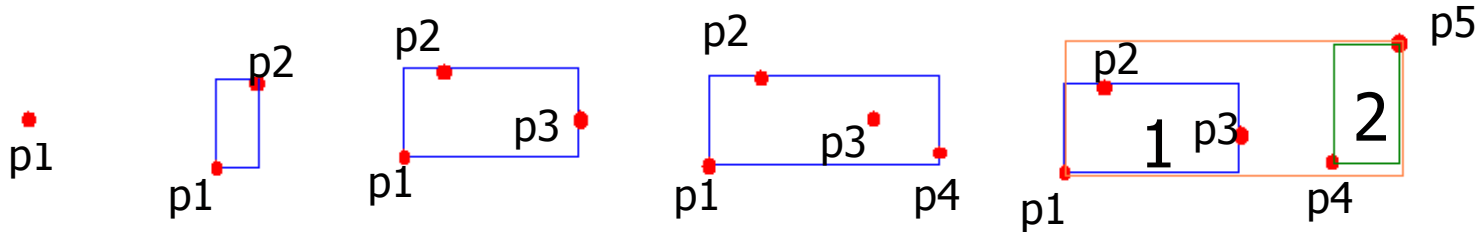


Les feuilles de l'index contiennent les vecteurs de caractéristiques des images

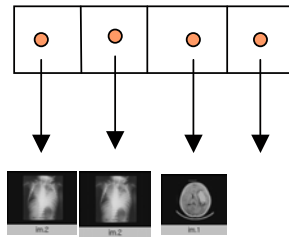


Capacité du rectangle= 4

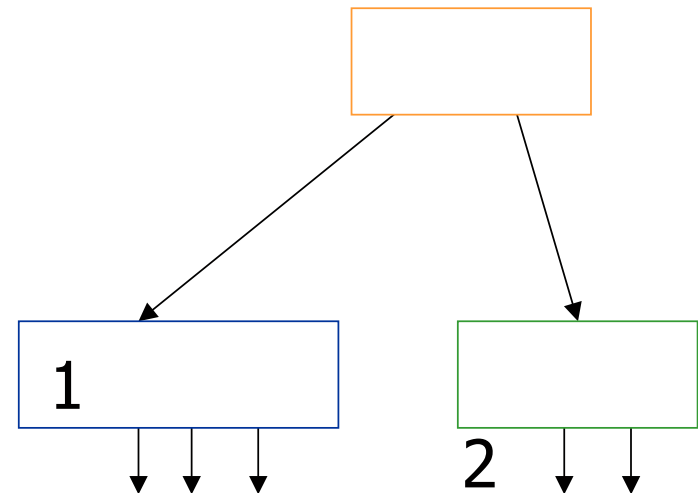
Arbre R appliqué aux images



Les feuilles de l'index contiennent les vecteurs de caractéristiques des images

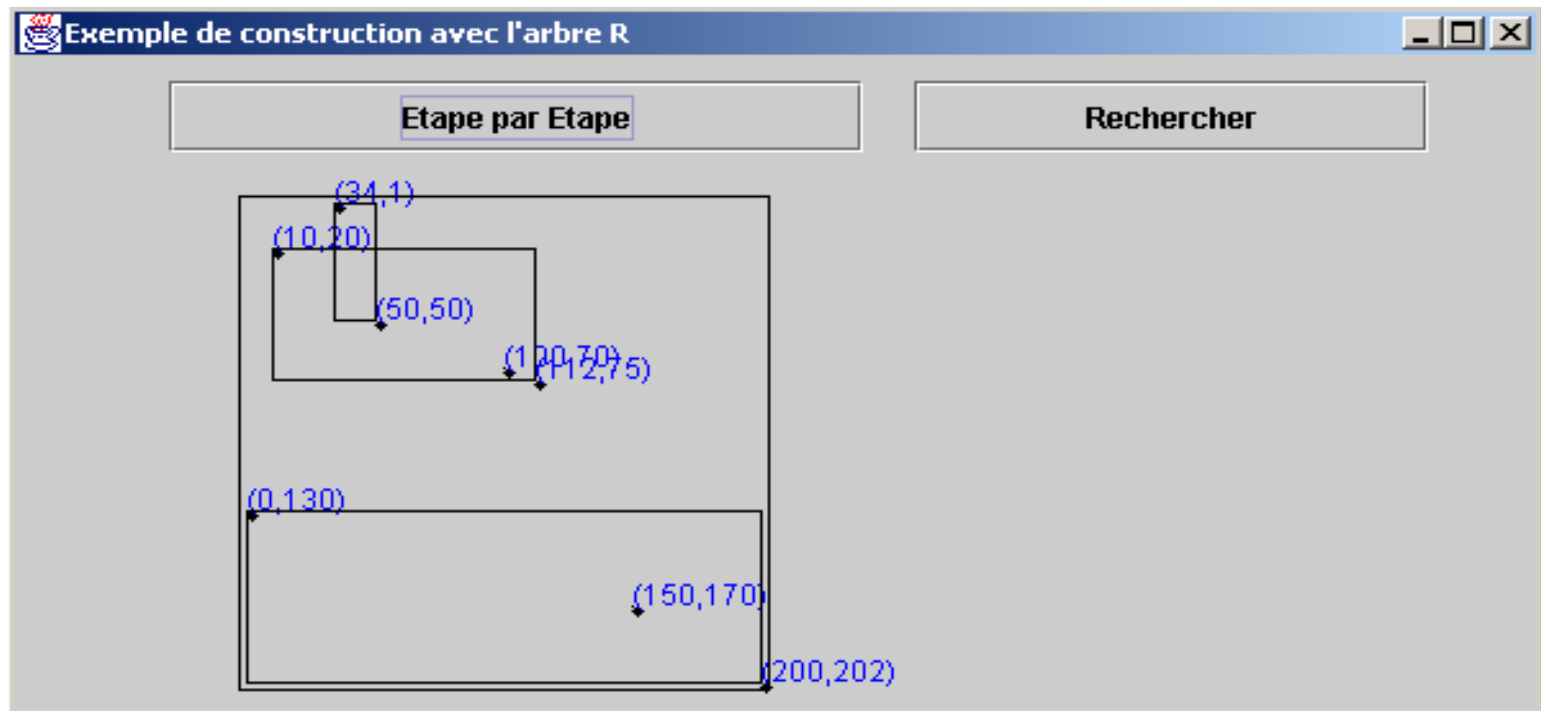


Capacité du rectangle = 4

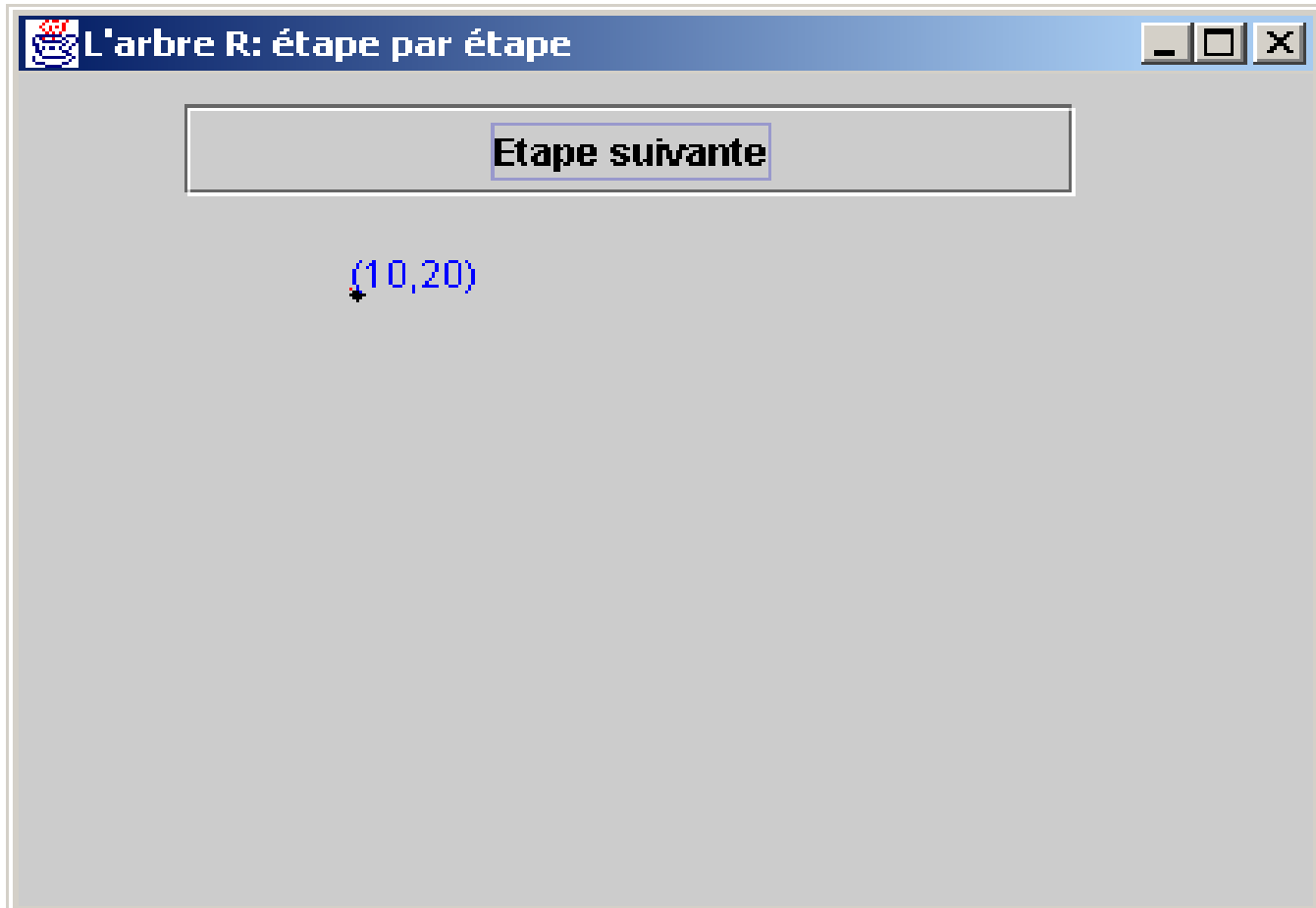


Exemple de construction de l'arbre R [Naï02]

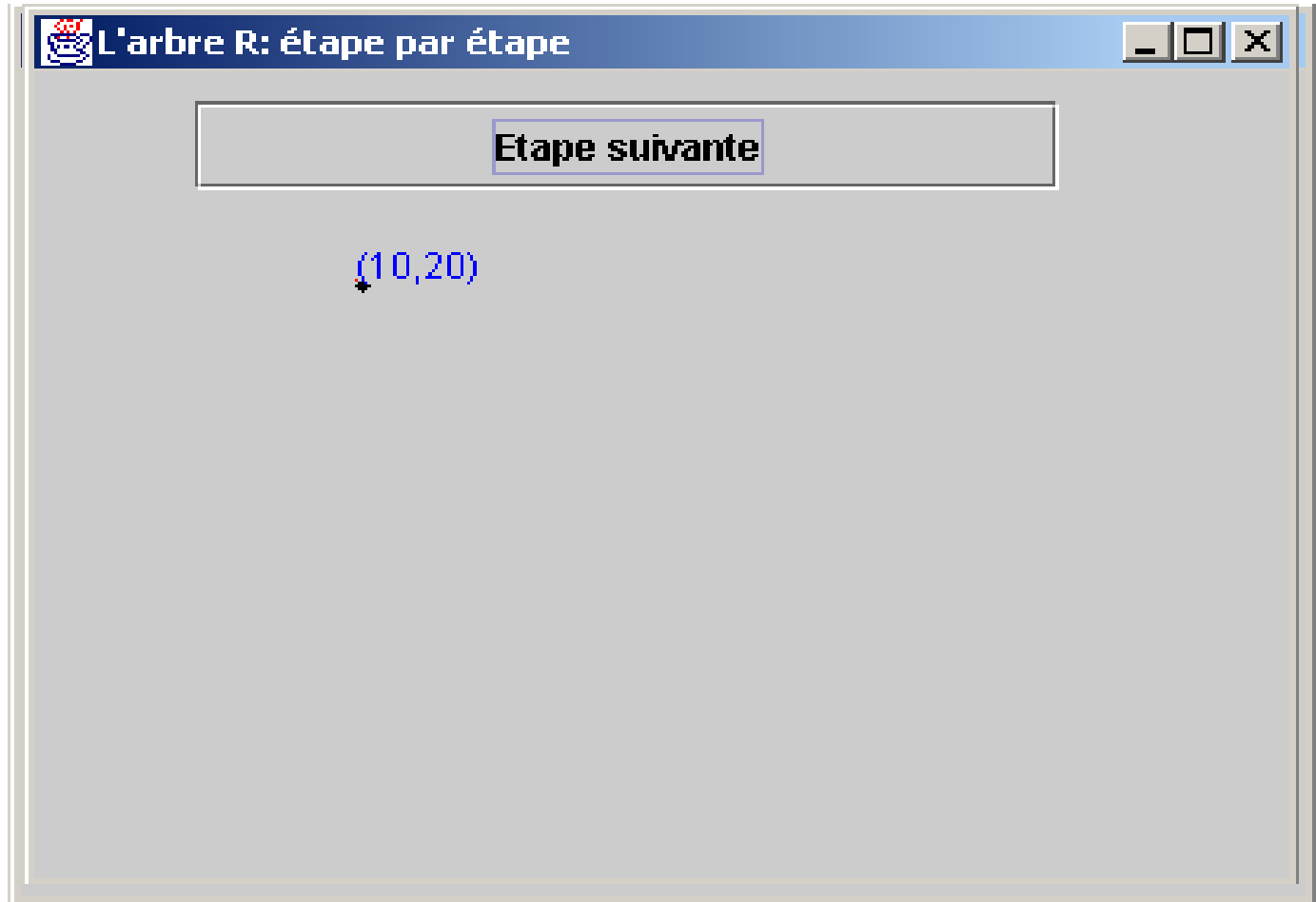
Prototype réalisé par Y. Naïja dans le cadre de son stage de DEA en 2002



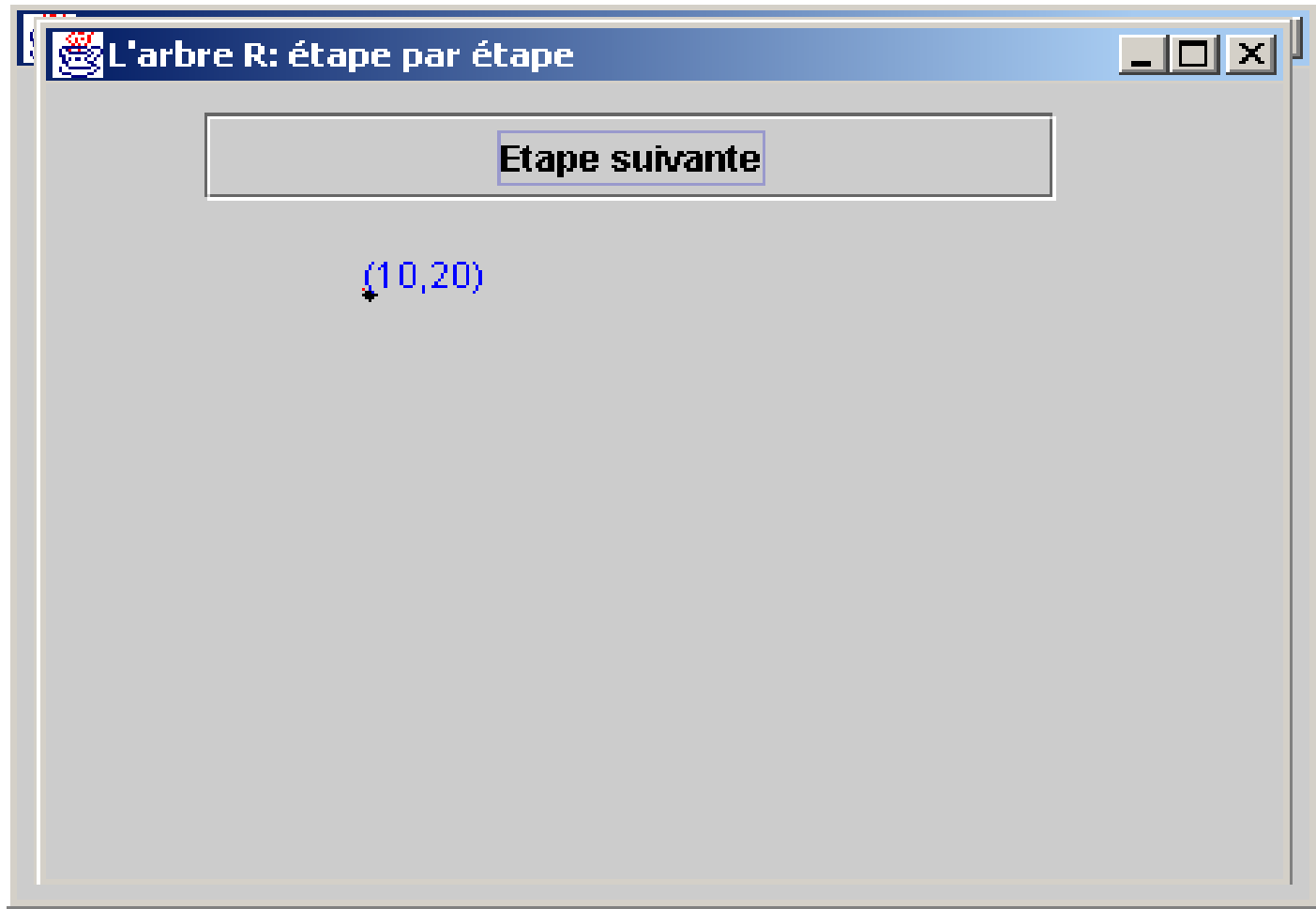
Etapes de construction d'un arbre R [Naï02]



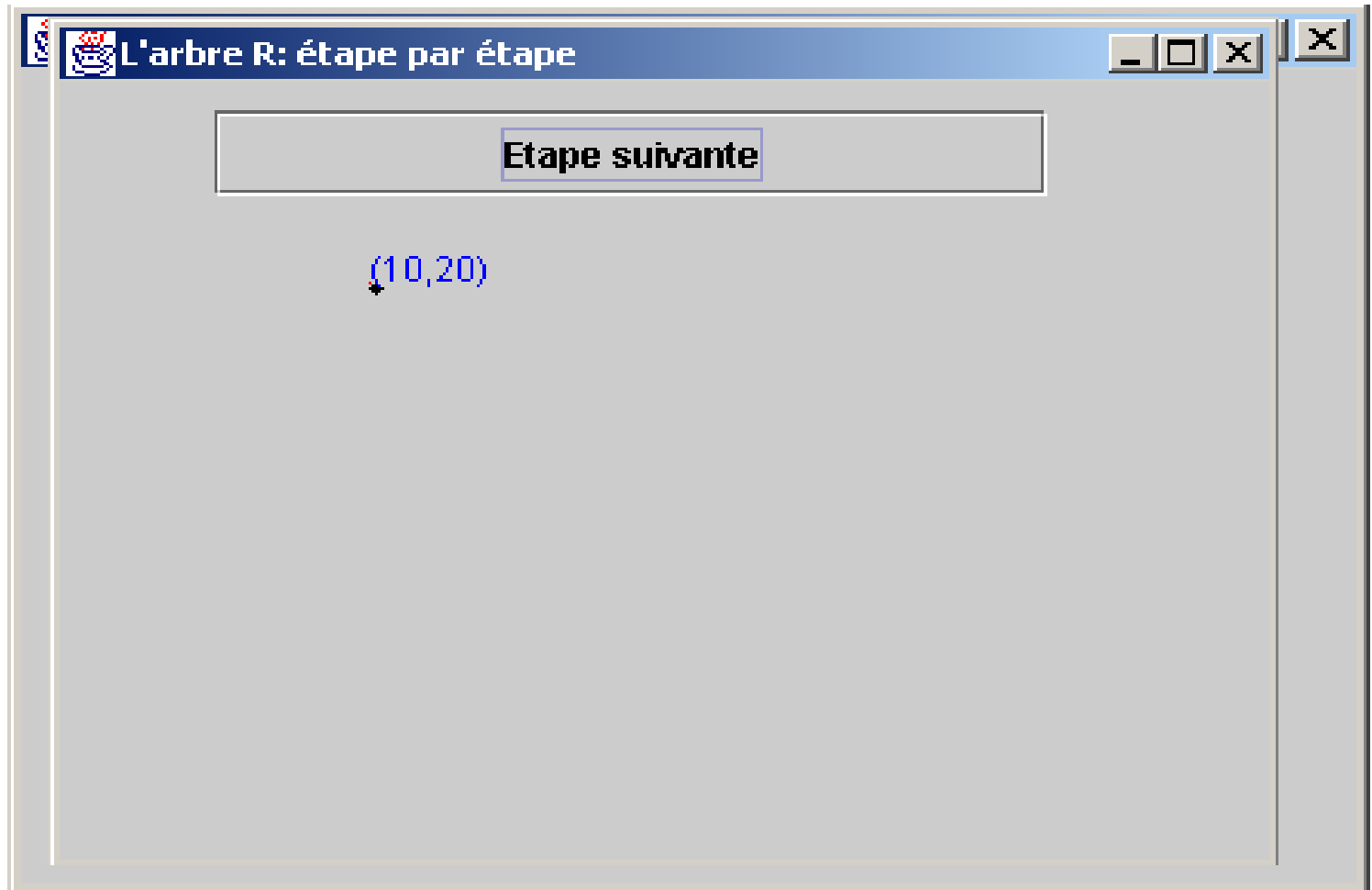
Etapes de construction d'un arbre R [Nai02]



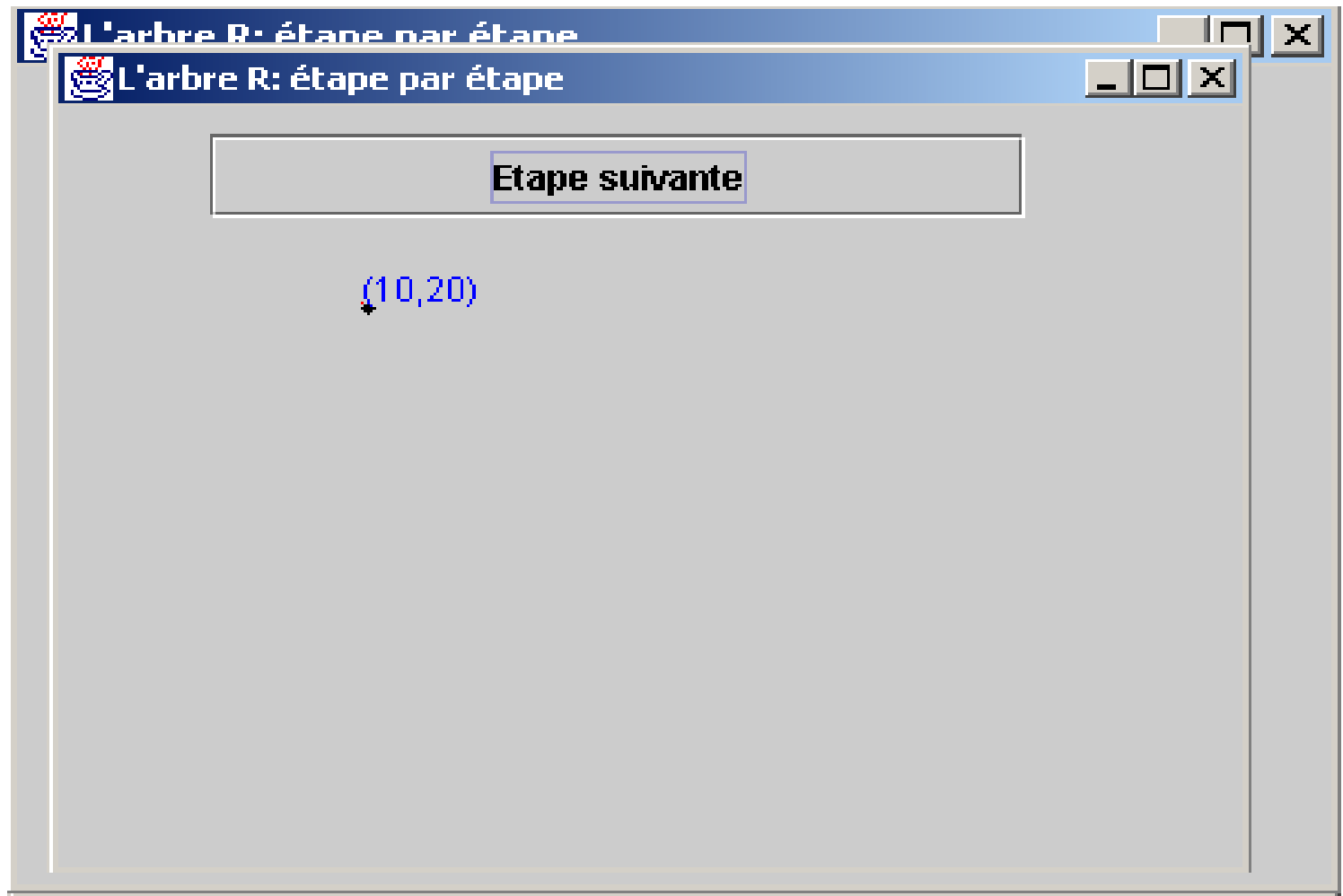
Etapes de construction d'un arbre R [Nai02]



Etapes de construction d'un arbre R [Nai02]



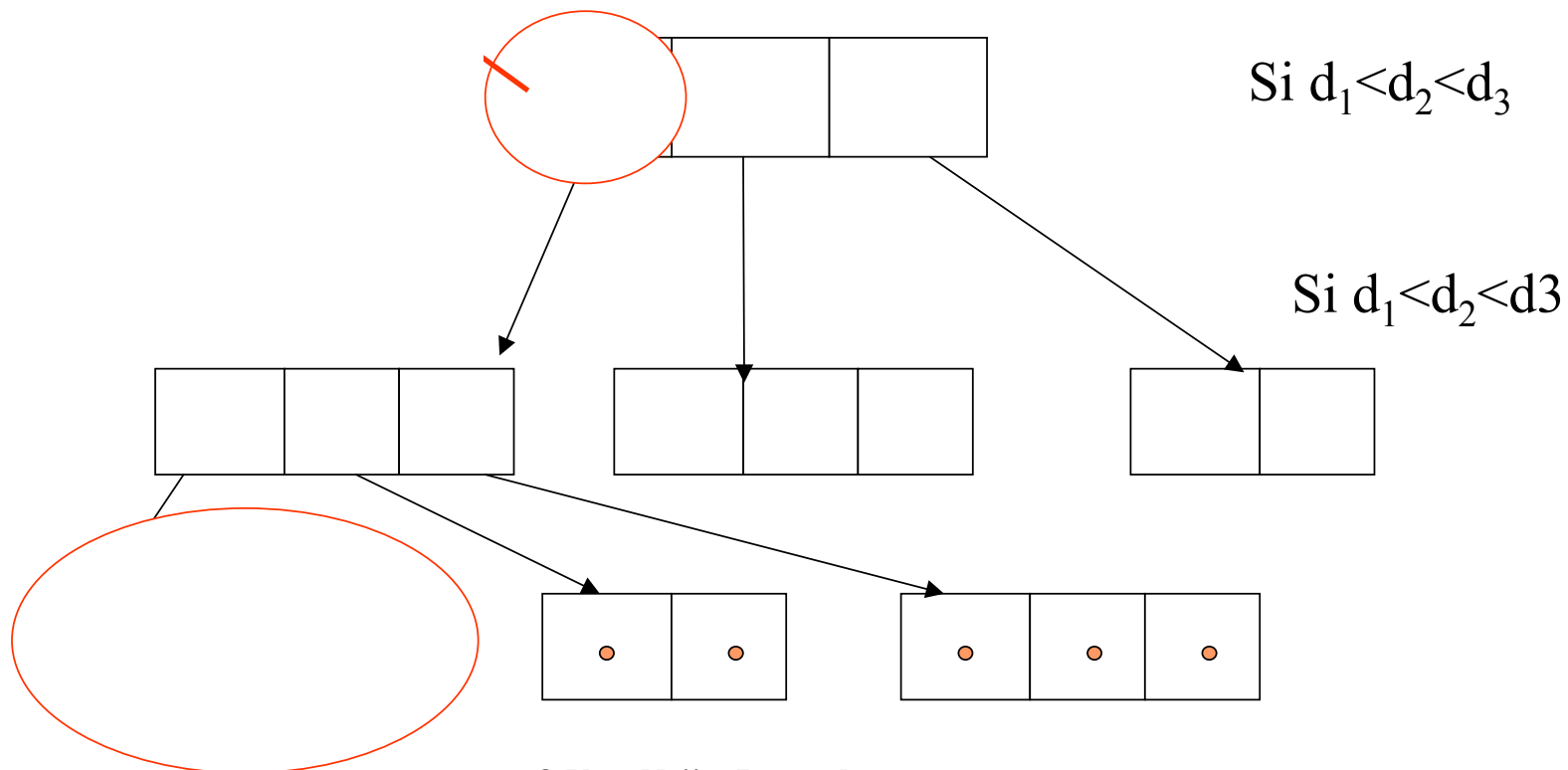
Etapes de construction d'un arbre R [Nai02]



Recherche dans l'arbre R



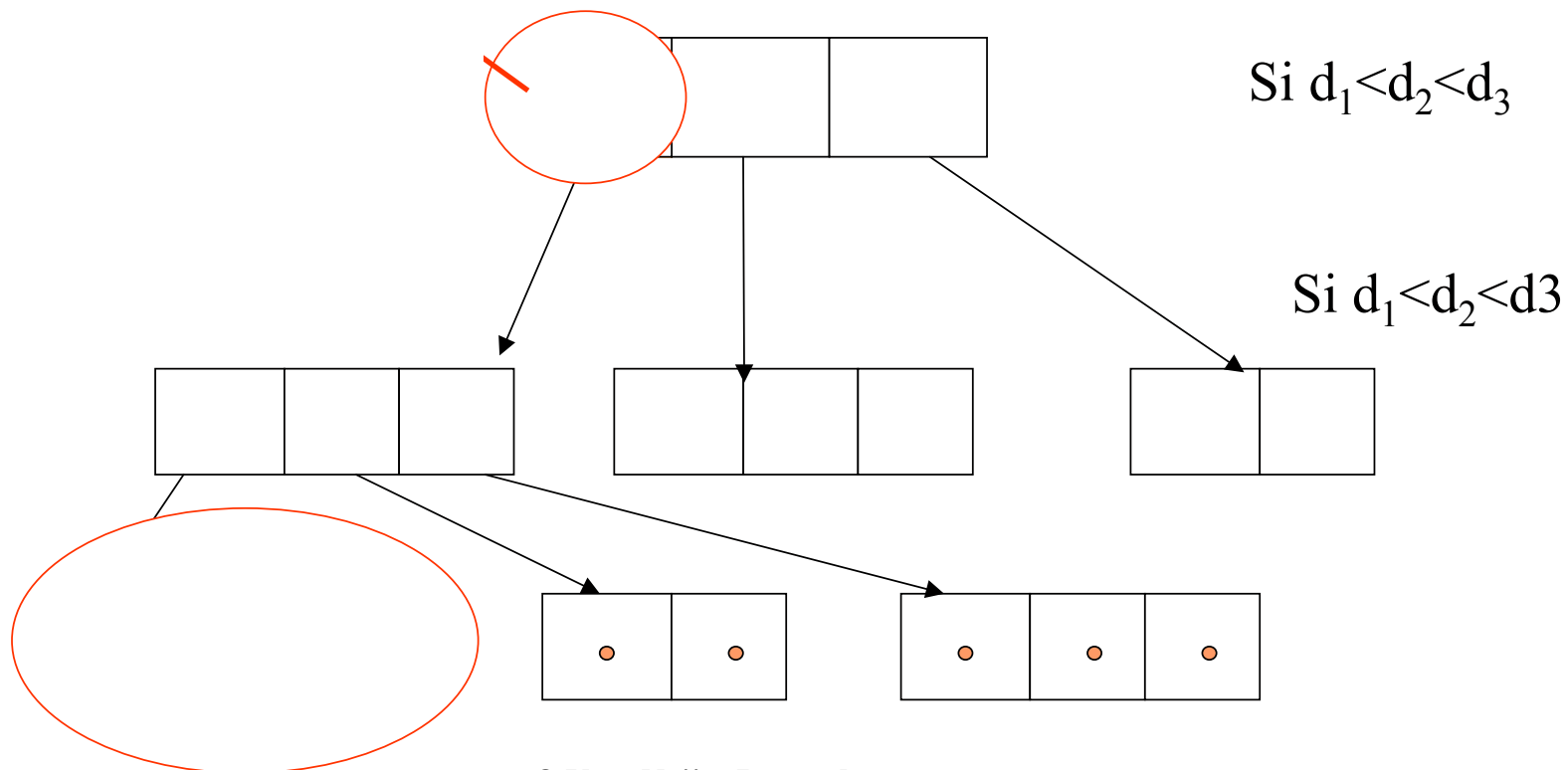
P ($a_1, a_2, a_3, \dots, a_n$) Point représentant l'image requête



Recherche dans l'arbre R



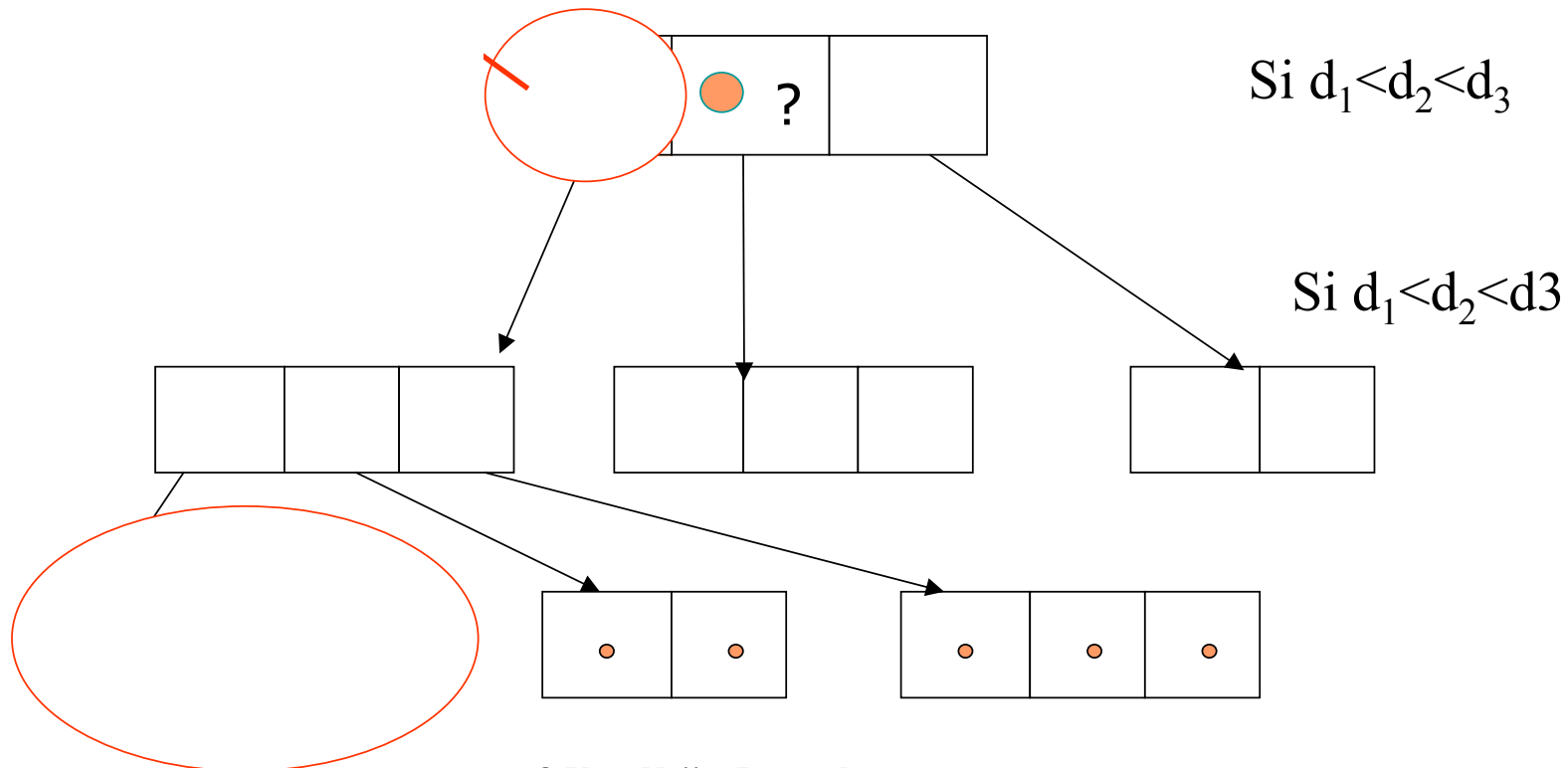
P ($a_1, a_2, a_3, \dots, a_n$) Point représentant l'image requête



Recherche dans l'arbre R



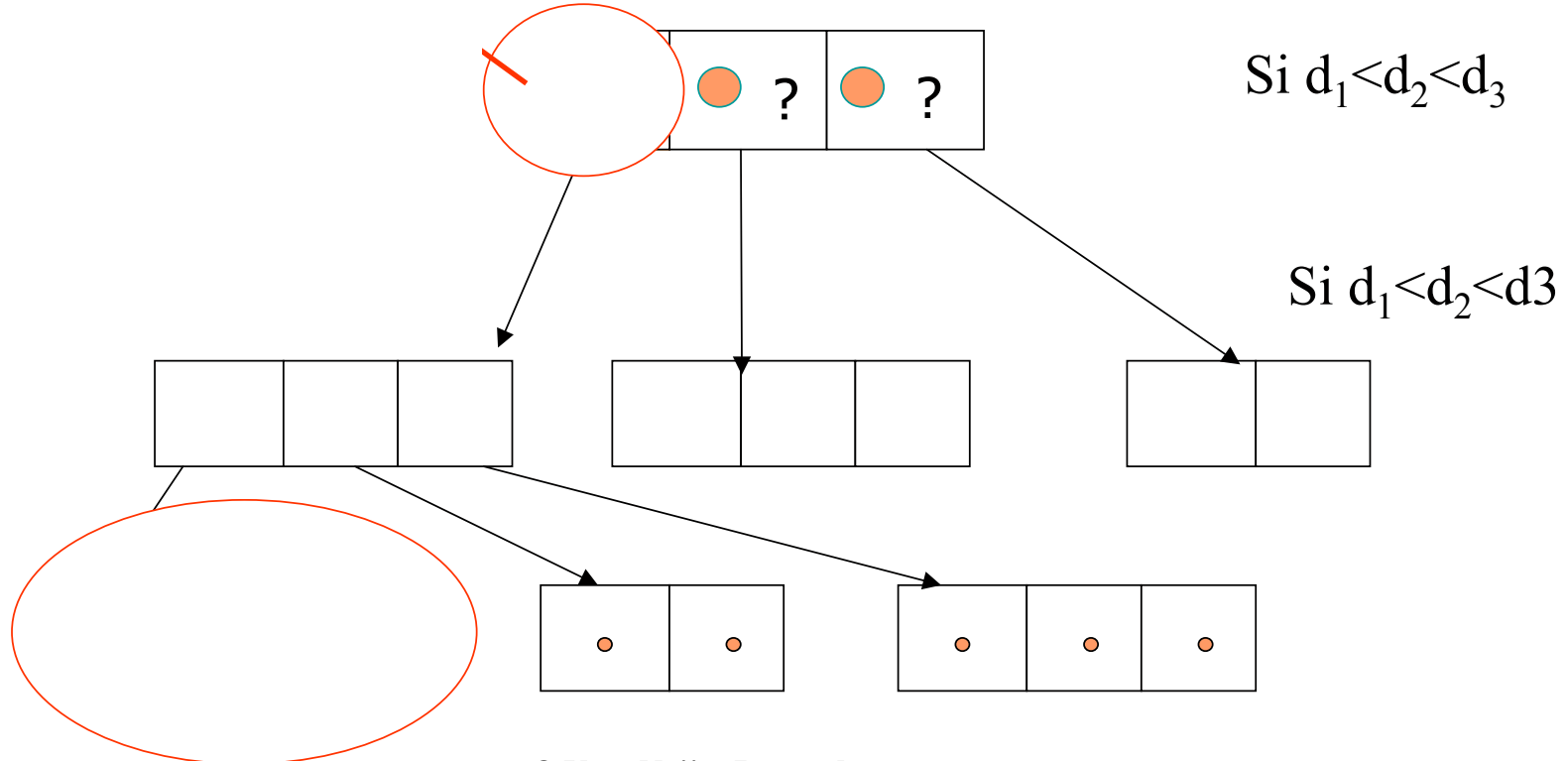
P ($a_1, a_2, a_3, \dots, a_n$) Point représentant l'image requête



Recherche dans l'arbre R



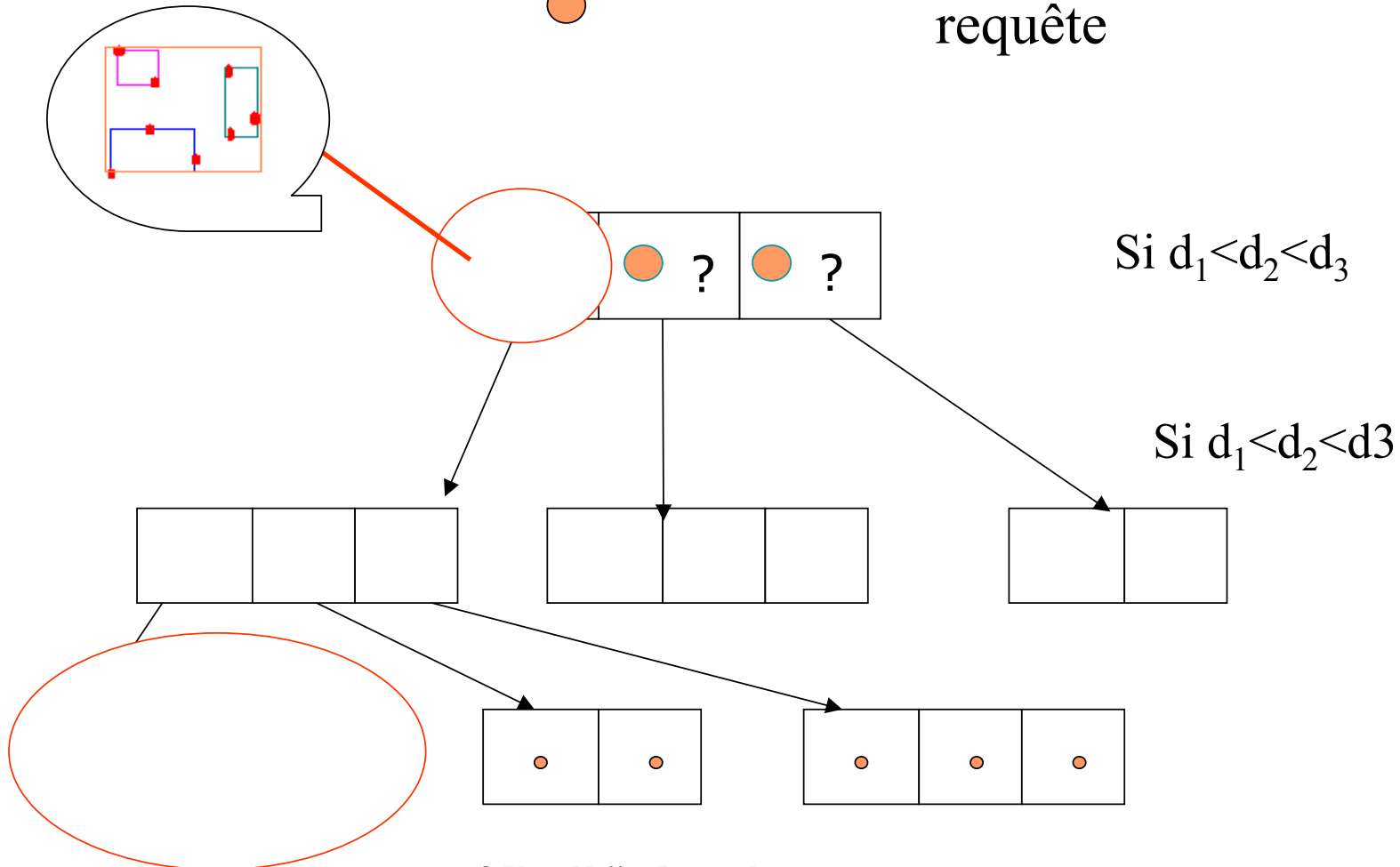
P ($a_1, a_2, a_3, \dots, a_n$) Point représentant l'image requête

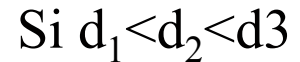


Recherche dans l'arbre R



P ($a_1, a_2, a_3, \dots, a_n$) Point représentant l'image requête

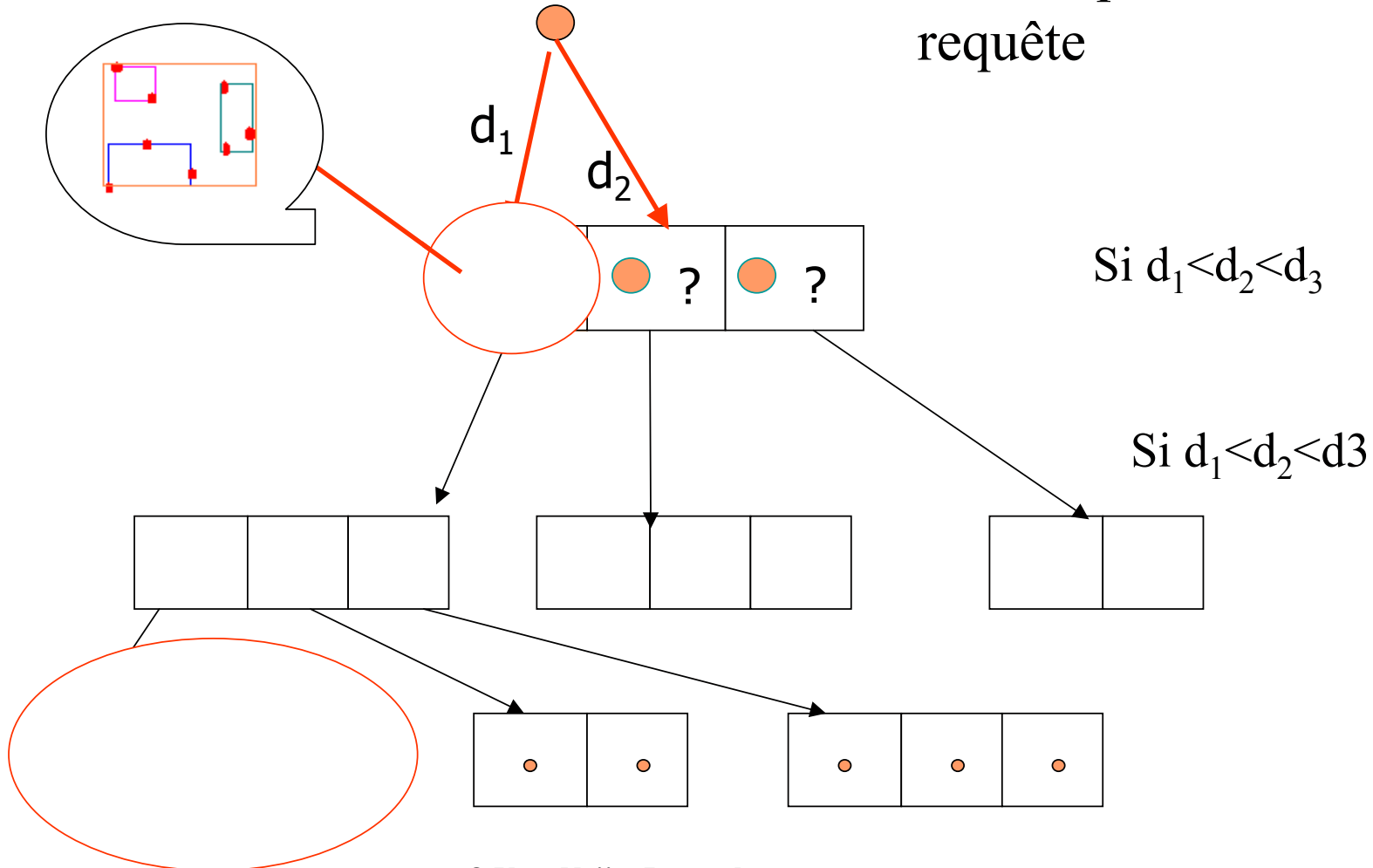


$$d_1$$


Recherche dans l'arbre R



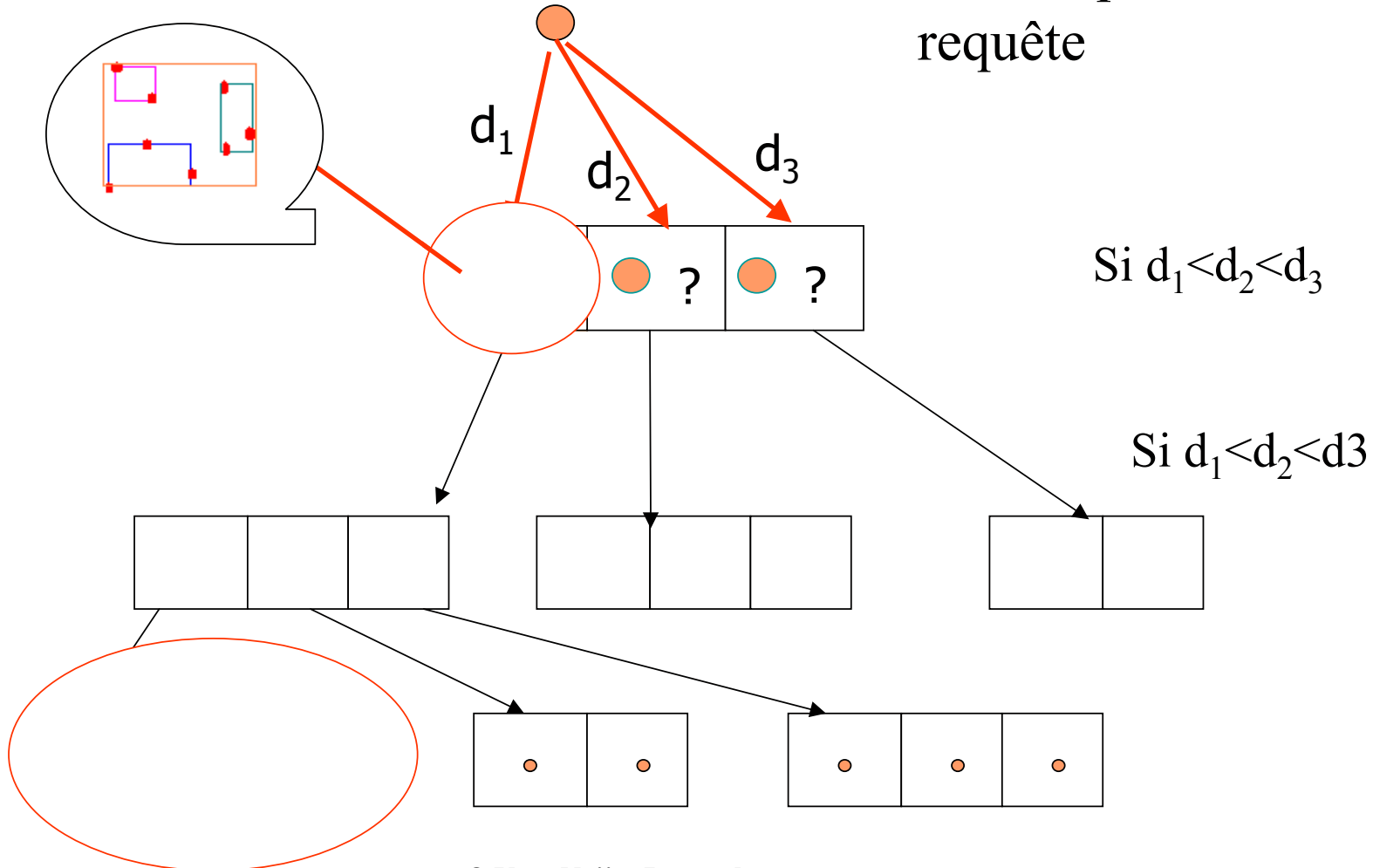
P ($a_1, a_2, a_3, \dots, a_n$) Point représentant l'image requête



Recherche dans l'arbre R



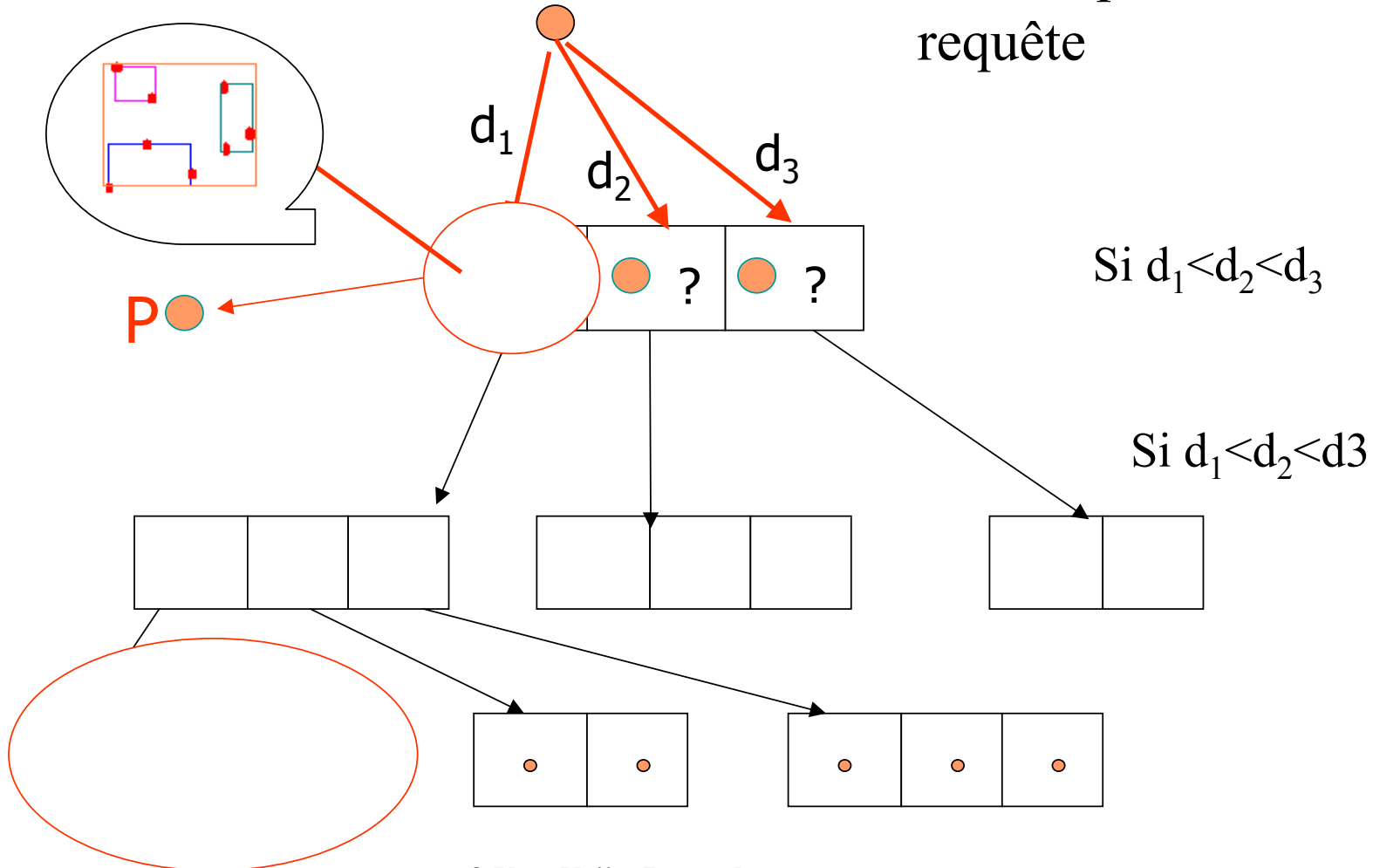
P ($a_1, a_2, a_3, \dots, a_n$) Point représentant l'image requête



Recherche dans l'arbre R



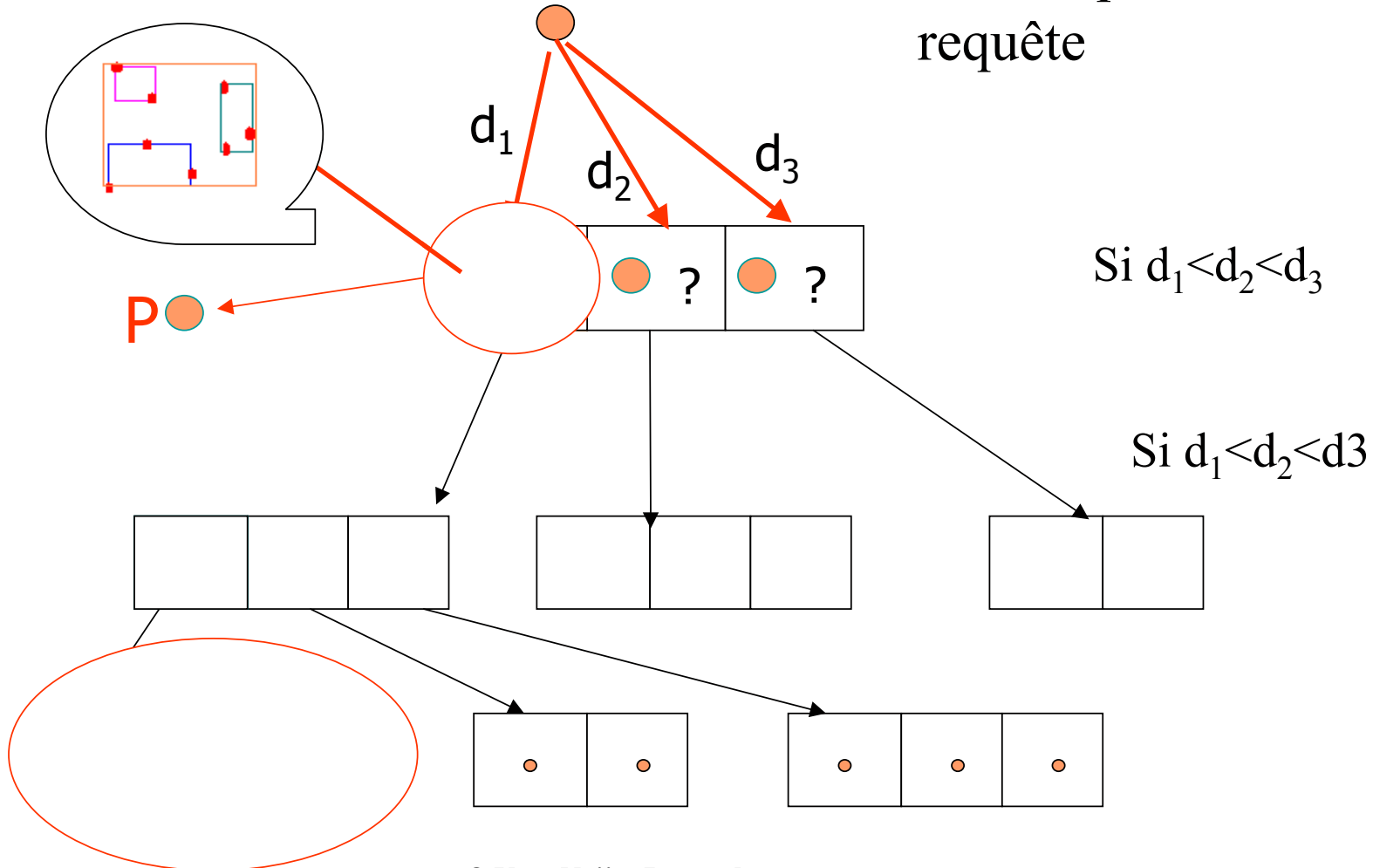
P ($a_1, a_2, a_3, \dots, a_n$) Point représentant l'image requête



Recherche dans l'arbre R



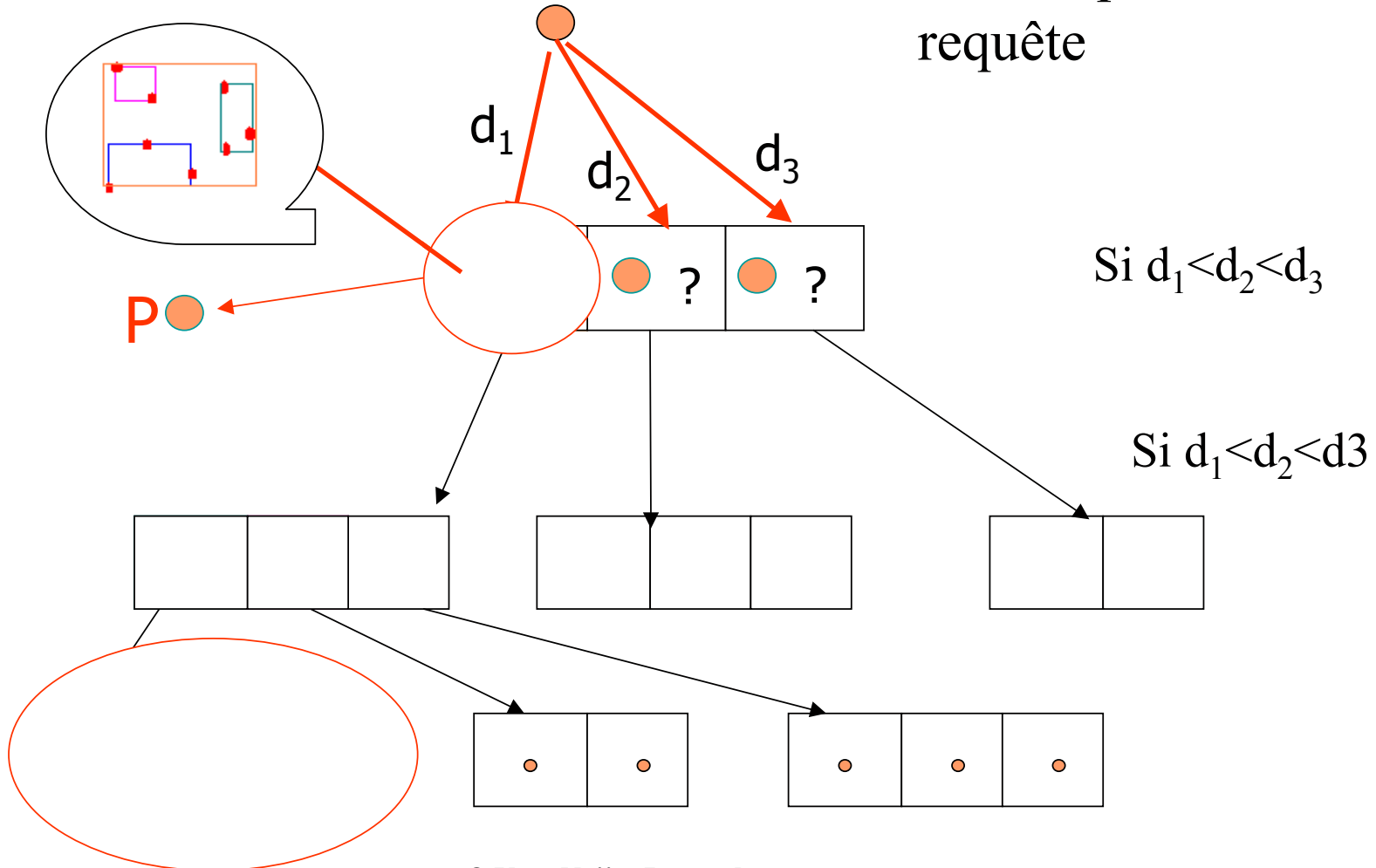
P ($a_1, a_2, a_3, \dots, a_n$) Point représentant l'image requête



Recherche dans l'arbre R



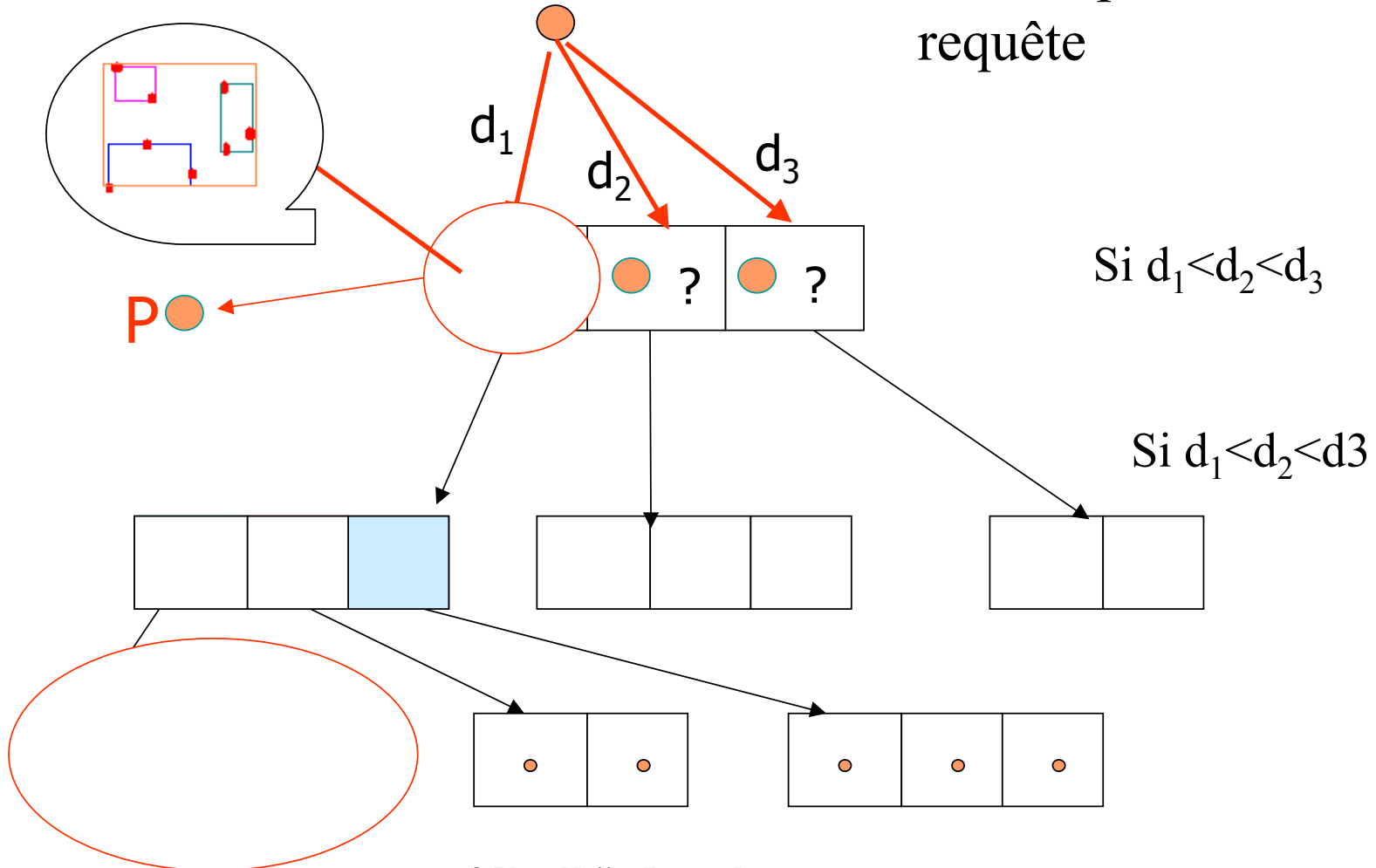
P ($a_1, a_2, a_3, \dots, a_n$) Point représentant l'image requête



Recherche dans l'arbre R



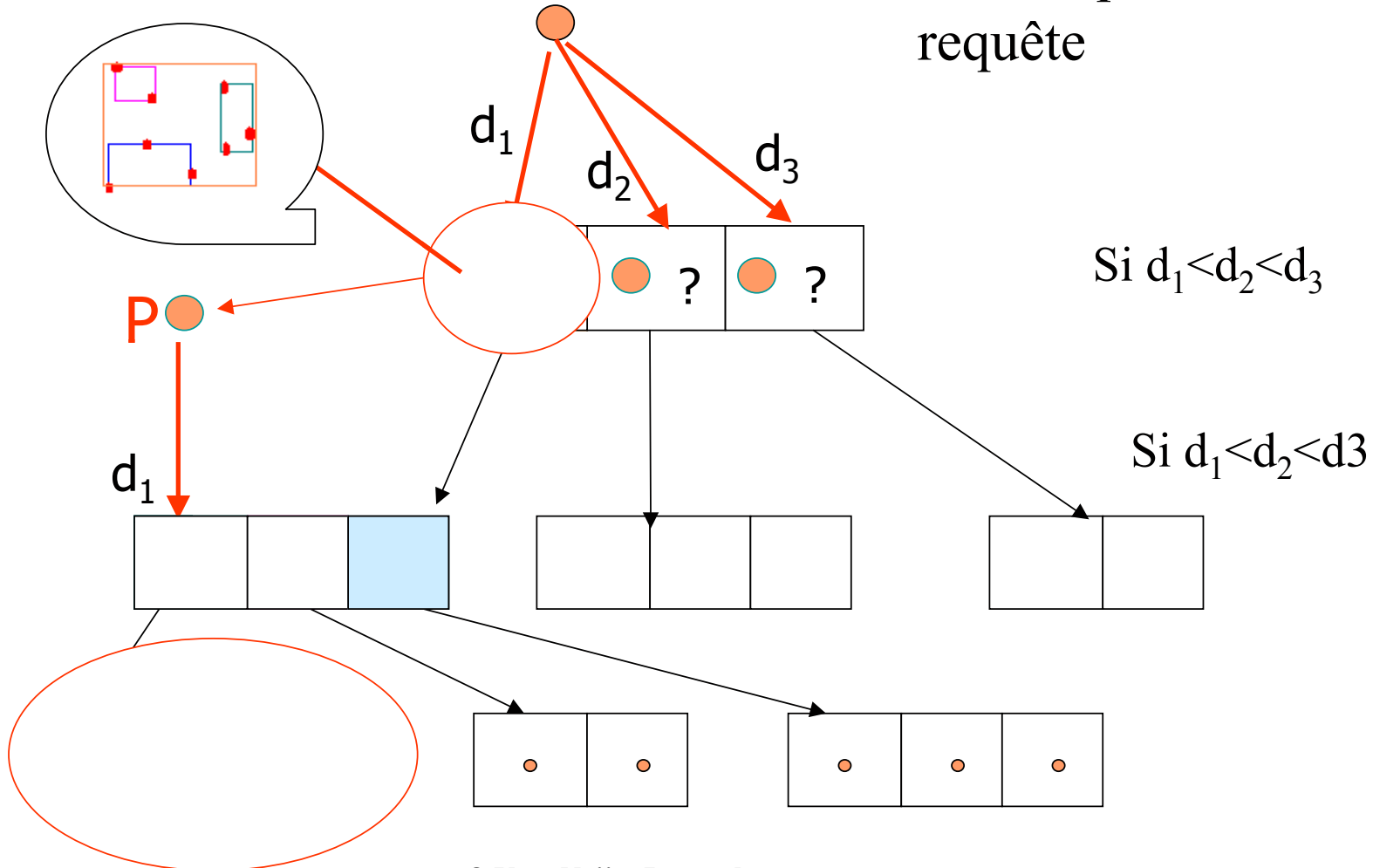
P ($a_1, a_2, a_3, \dots, a_n$) Point représentant l'image requête



Recherche dans l'arbre R



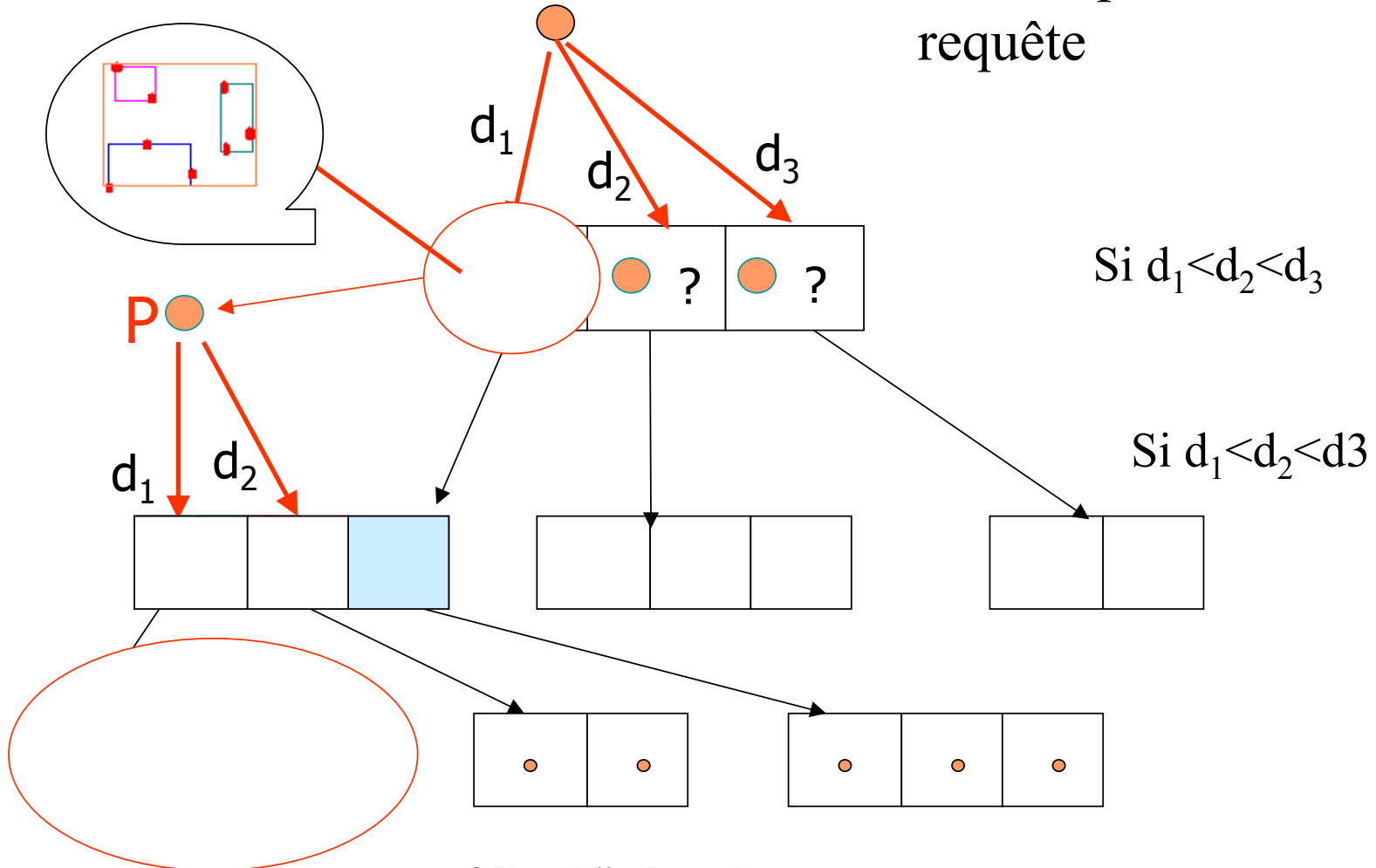
P ($a_1, a_2, a_3, \dots, a_n$) Point représentant l'image requête



Recherche dans l'arbre R



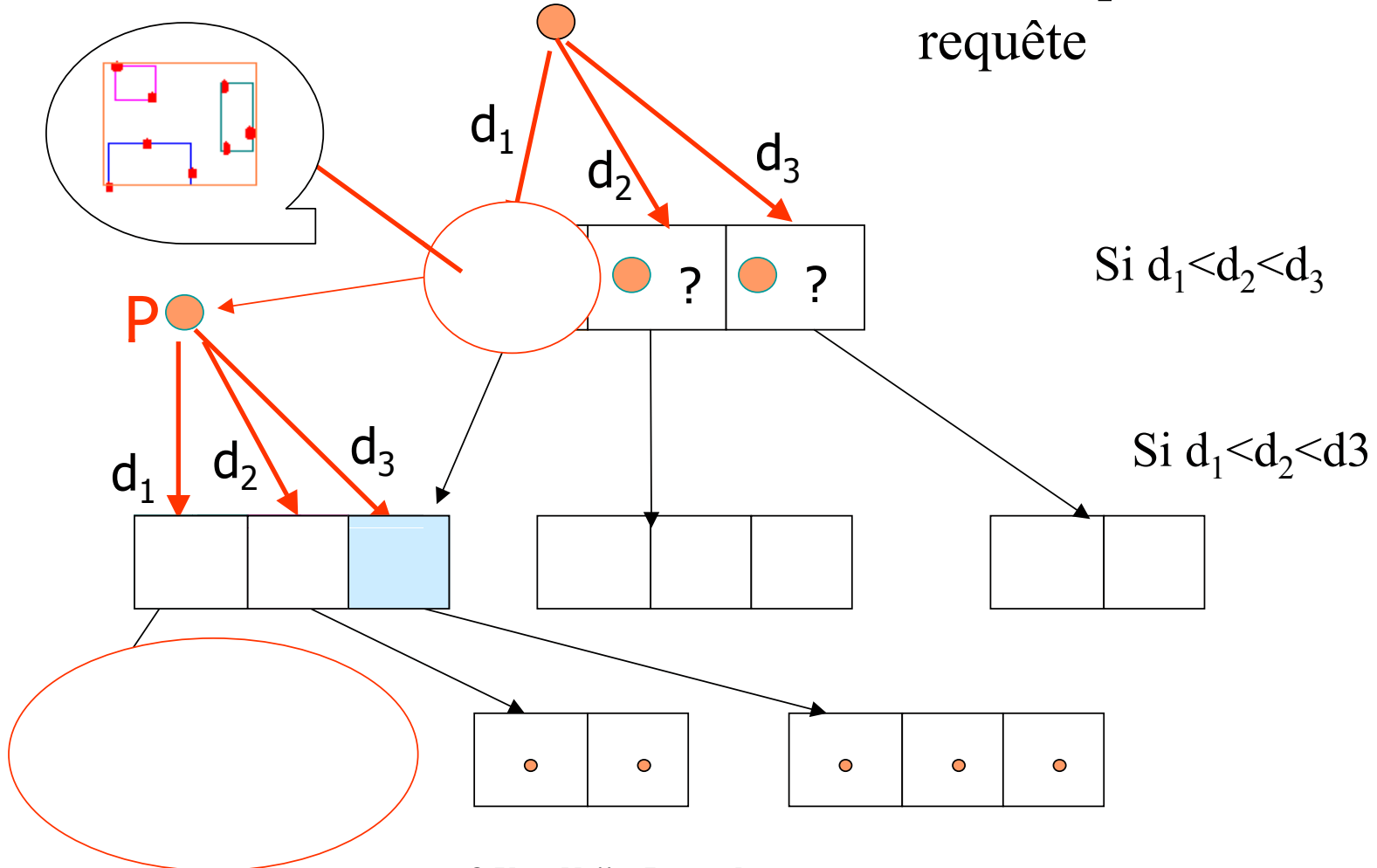
P ($a_1, a_2, a_3, \dots, a_n$) Point représentant l'image requête



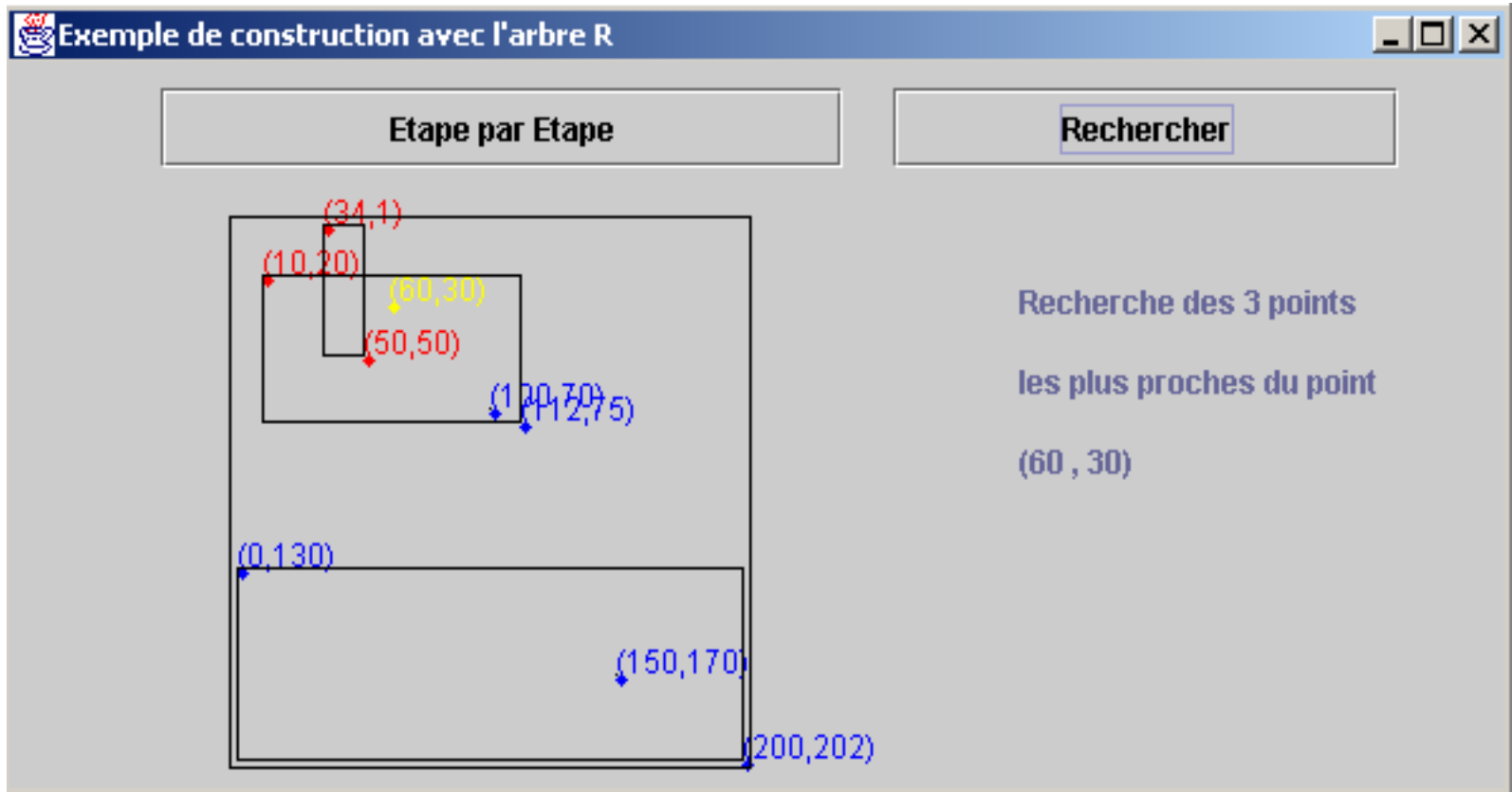
Recherche dans l'arbre R



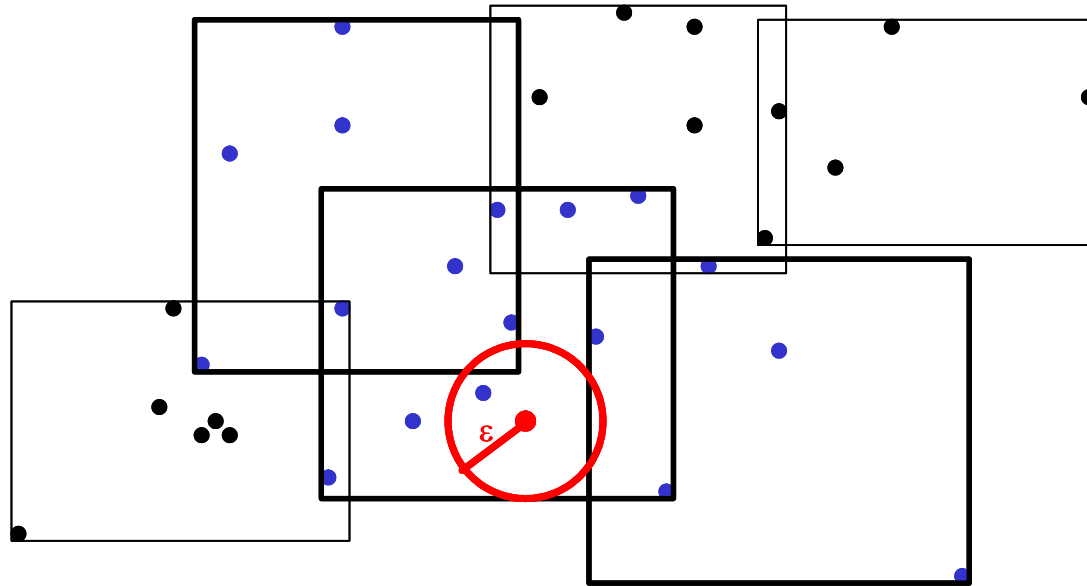
P ($a_1, a_2, a_3, \dots, a_n$) Point représentant l'image requête



Recherche dans l'arbre R [Naï02]



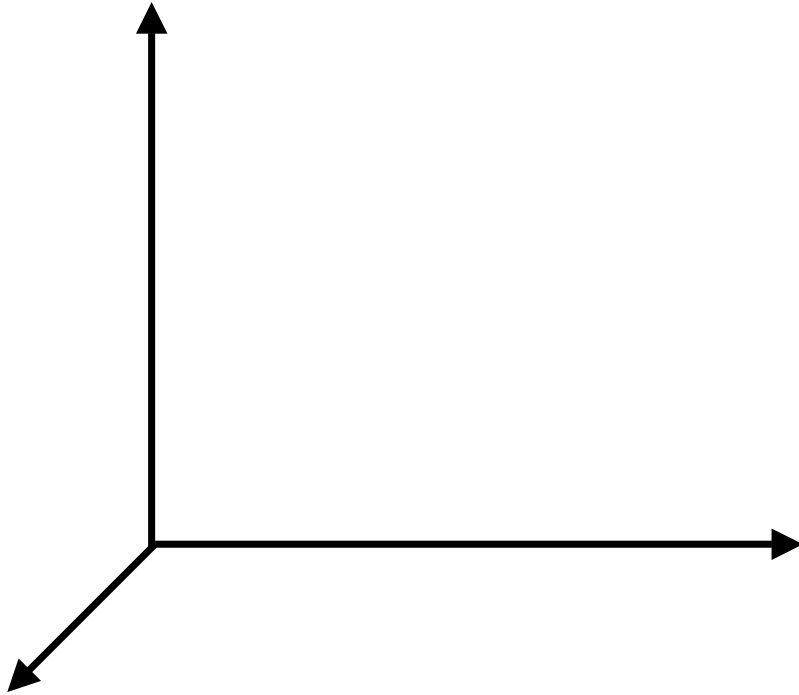
Recherche dans l'arbre R



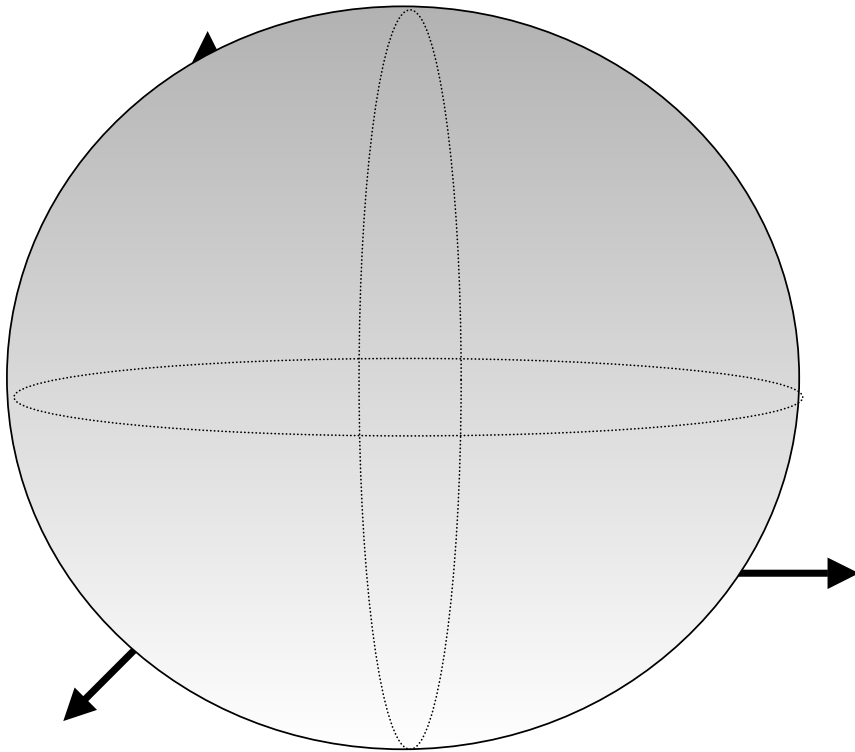
Recherche dans un arbre R

- Problème majeur de l'arbre R : parcours inutile parfois d'un grand nombre de nœuds pour retrouver la donnée recherchée [OA04,OLD04]
- Arbres dérivés de l'arbre R pour éviter le chevauchement de rectangles englobants : arbre R+ [SRF87] et arbre R* [BKS+90]
- R-tree plutôt orienté pour les requêtes d'intervalles

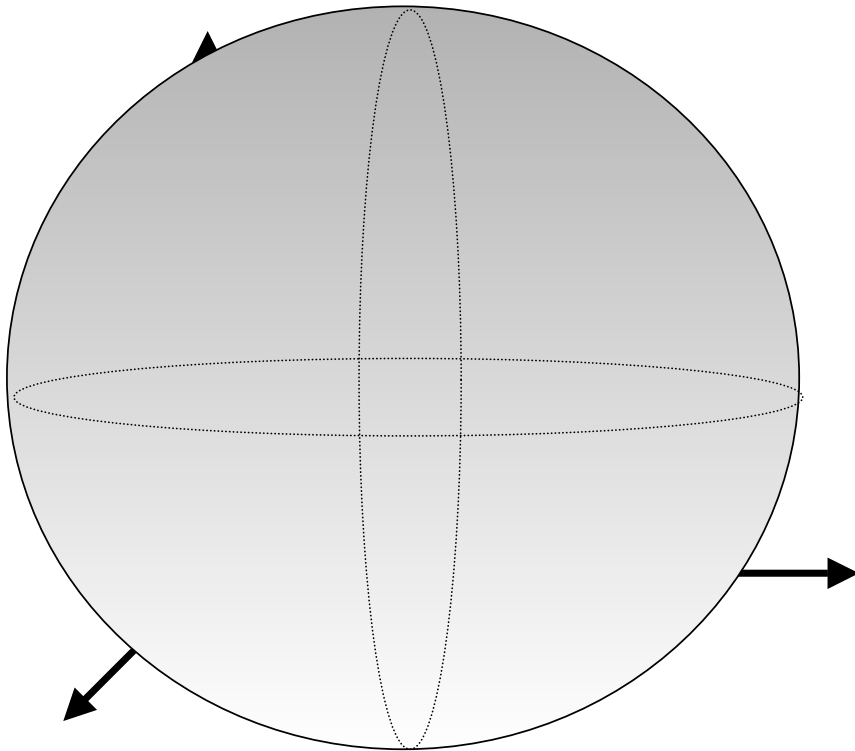
Structure d'index : arbre SS



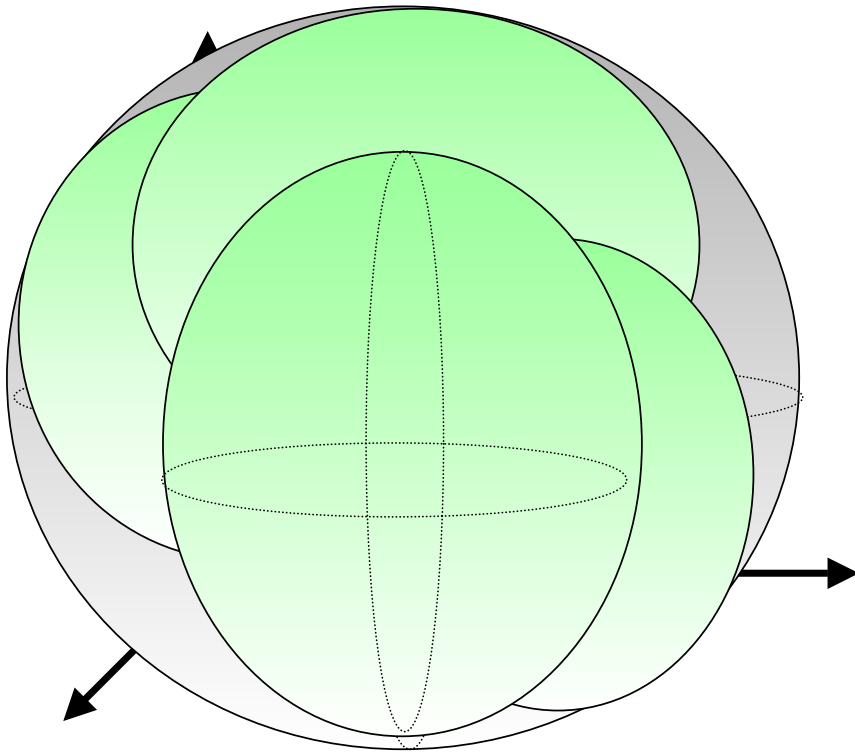
Structure d'index : arbre SS



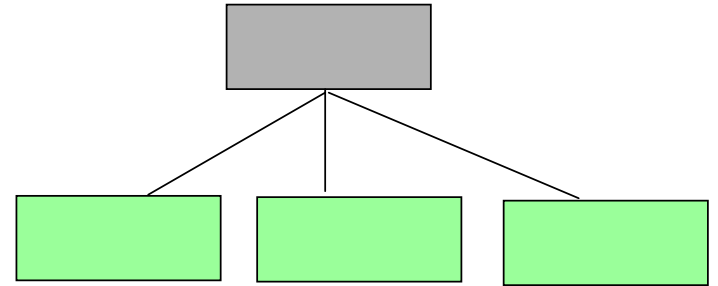
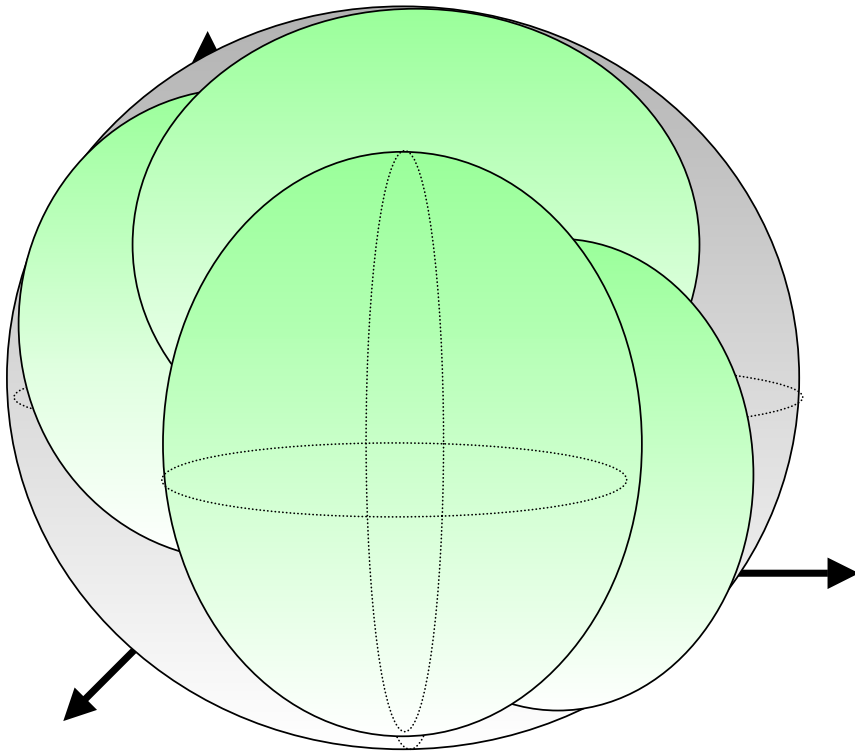
Structure d'index : arbre SS



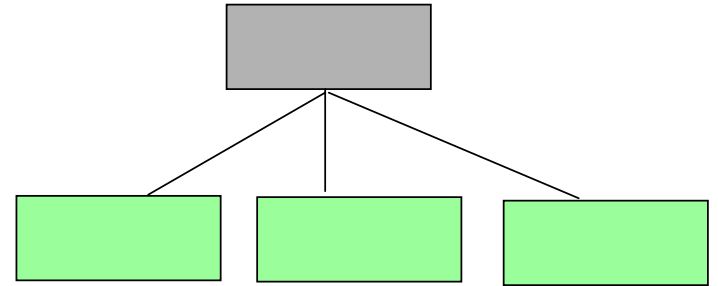
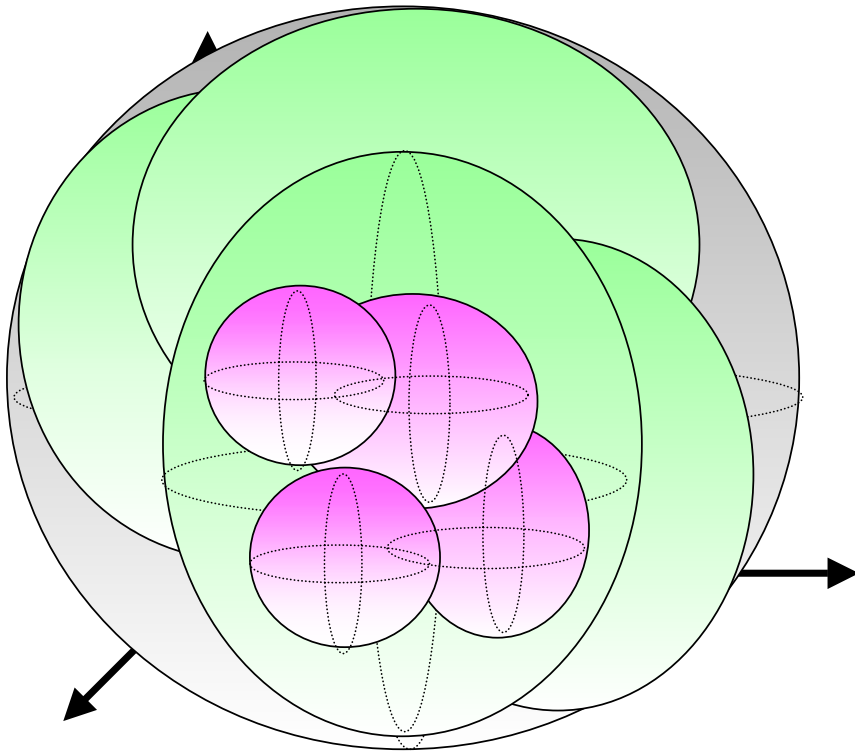
Structure d'index : arbre SS



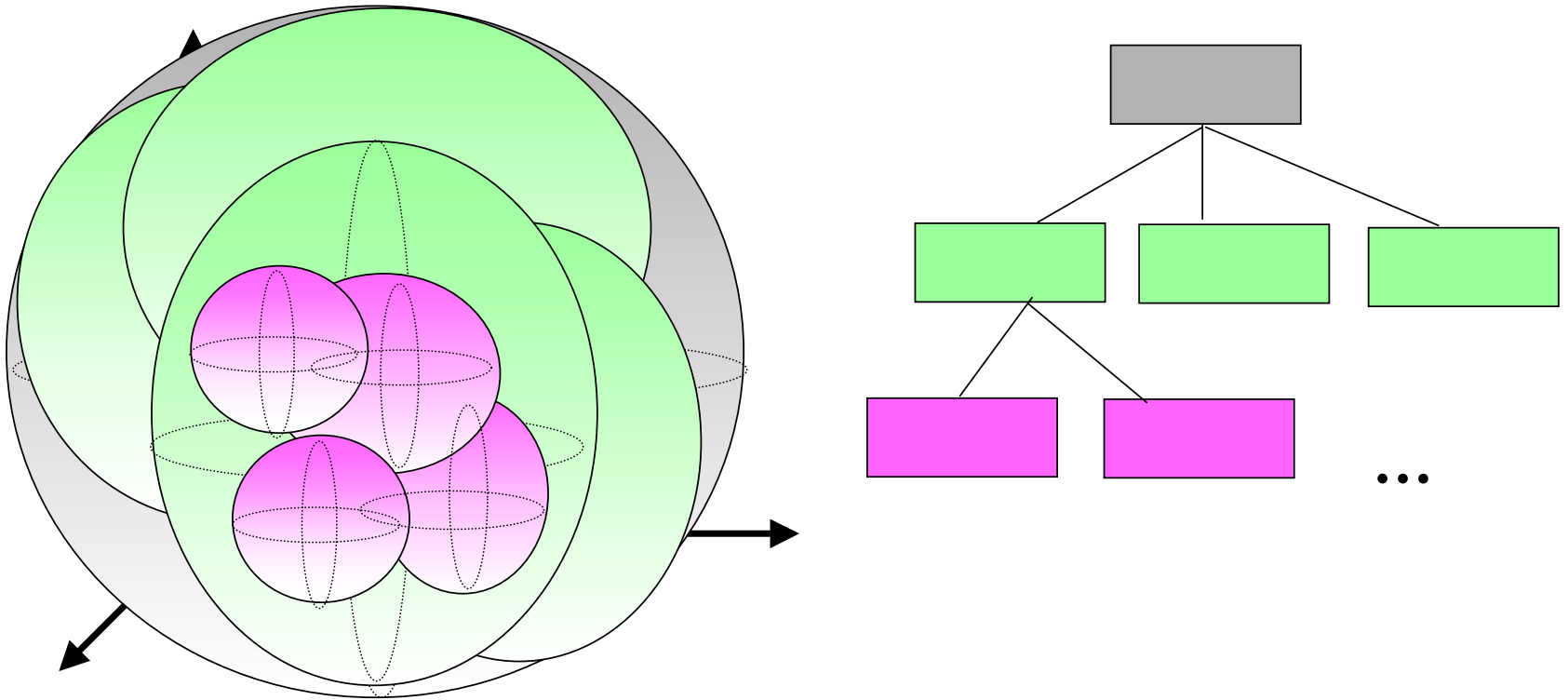
Structure d'index : arbre SS



Structure d'index : arbre SS

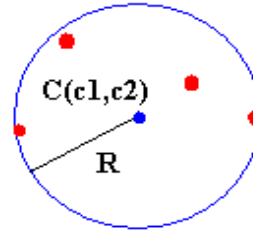


Structure d'index : arbre SS

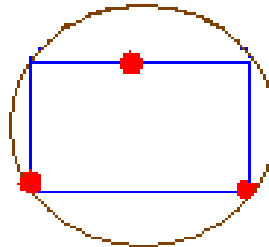


Arbre SR [KS97]

- Basé sur les rectangles et les sphères minimum englobantes



- Avantage des sphères / Inconvénient des rectangles :
Recherche plus appropriée pour la similarité
- Avantage des rectangles / Inconvénient des sphères :
Volume du rectangle plus petit



SR-Tree

“The SR-tree: An Indexing Structure for High-Dimensional Nearest Neighbor Queries.”

Norio Katayama

Multimedia Info. Research Div.
National Institute of Informatics

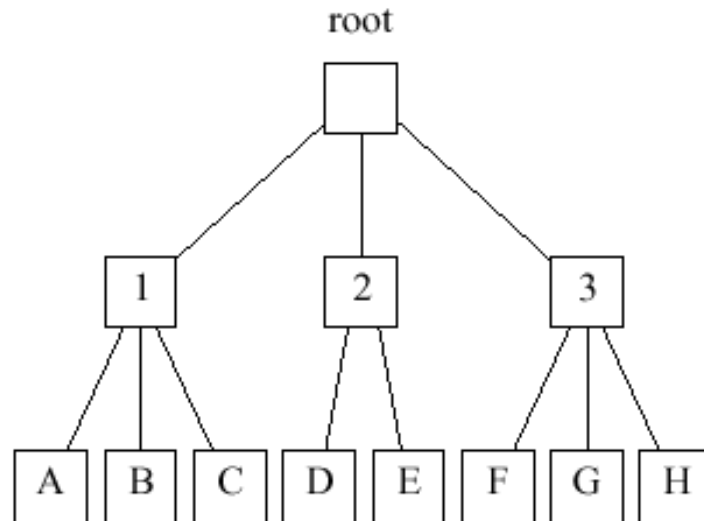
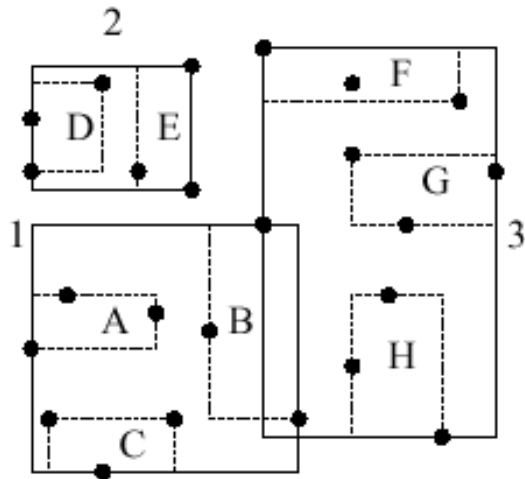
Shin'ichi Satoh

Software Research Div.
National Institute of Informatics

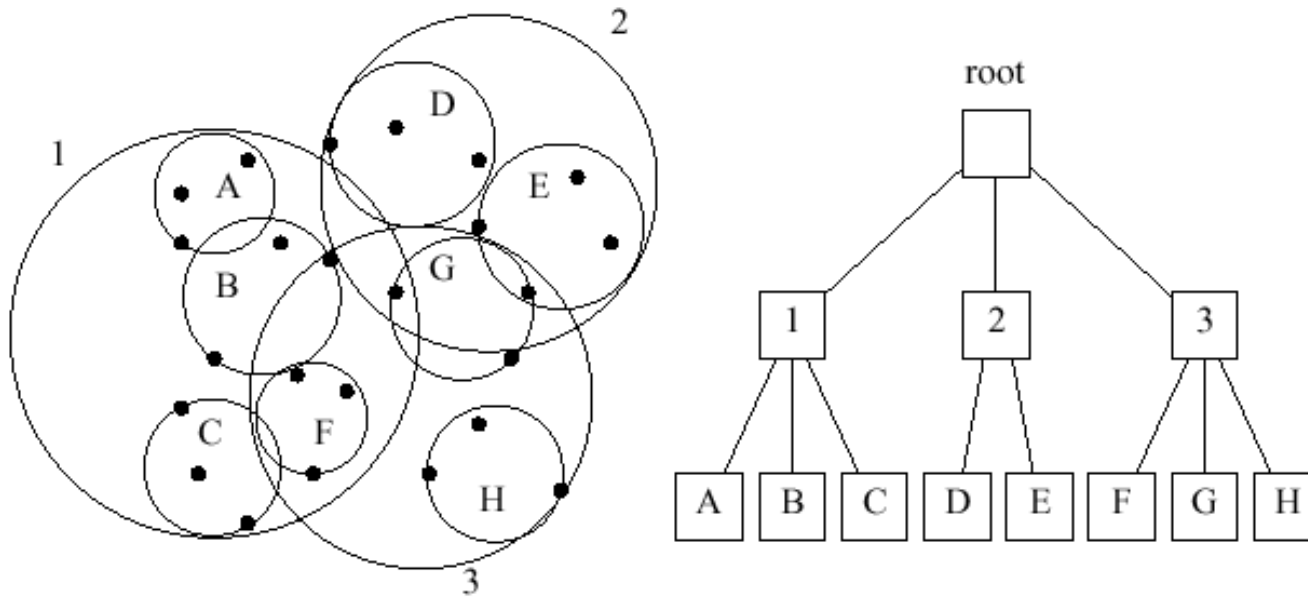
SR-Tree : Introduction

- SR-tree stands for “Sphere/Rectangle-Tree”
- SR-Tree is an extension of the R^* -tree and the SS-tree
- A region of the SR-tree is specified by the intersection of a bounding sphere and a bounding rectangle

The R*-Tree Structure



The SS-Tree Structure



Definitions (SR-Tree)

- The diameter of a bounded region means
 - ◆ diameter of a bounding sphere for the SS-Tree
 - ◆ diagonal of a bounding rectangle for the R*-Tree

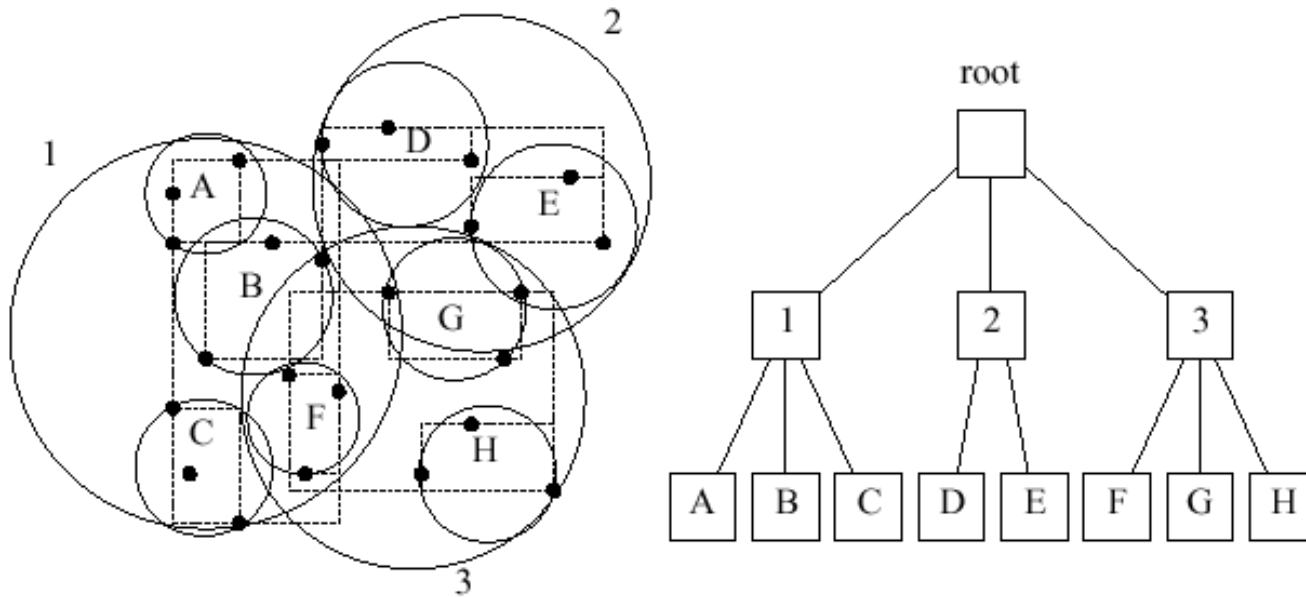
Properties

- Bounding rectangles divides points into smaller volume regions. But tends to have longer diameters than bounding spheres, especially in high-dimensional space.
- Bounding spheres divides points into short-diameter regions. But tends to have larger volumes than bounding rectangles.

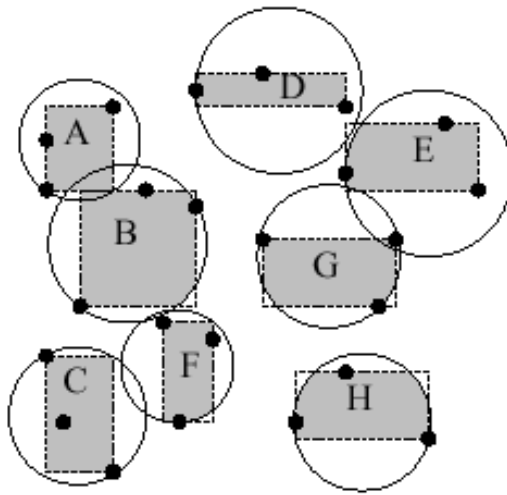
Properties

- SR-Tree combined the use of bounding sphere and bounding rectangle, as the properties are complementary to each other.

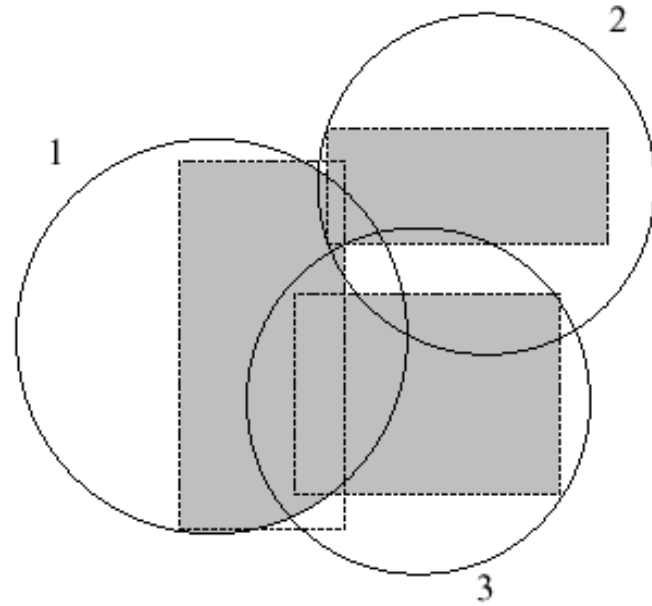
The SR-Tree Structure



Bounded regions

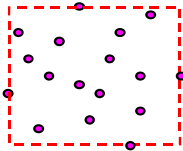
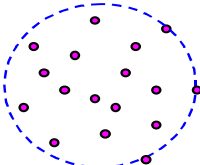
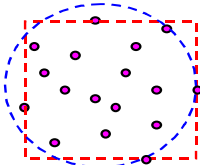


(a) Leaf level



(b) Node level

Arbres R, SS et SR

	R*-Tree	SS-Tree	SR-Tree
Region Shape			
Tree Construction Strategy	Minimizing Volume and Overlap	Minimizing Diameter	Minimizing Diameter
Diameter	4	1	1
Volume	60	10^9	1
Range Search	Very Good	Fair	Good
Nearest Neighbor Search	Fair	Good	Very Good

Malédiction/spirale dimensionnelle

curse of dimensionality [BGR99, OA04, OLD04] :

- Diminution des performances des index multidimensionnels proportionnellement au nombre de dimensions
- Accroissement des dimensions
 - ⇒ la distance d'un point donné au point le plus proche tend vers la distance au point le plus éloigné
 - ⇒ Faible occupation de l'espace
 - ⇒ En moyenne toutes les feuilles des index sont visitées
 - ⇒ Moins bonnes performances qu'une recherche séquentielle
- Observation de ce phénomène au delà de 10 à 15 dimensions
- Solutions : réduction des dimensions, VA-FILE ou arbres métriques

Réduction des dimensions

“Etant donné un ensemble de vecteurs dans un espace à N dimensions, trouver un ensemble de vecteurs correspondants dans un espace à K dimensions ($K \ll N$), qui maintienne autant que possible les distances entre les vecteurs que dans l’espace d’origine “ [OA04, OLD04].

Minimisation de : erreur moyenne =
$$\sqrt{\frac{\sum_{i,j} (d'_{ij} - d_{ij})^2}{\sum_{i,j} d_{ij}^2}}$$

Avec d_{ij} distance dans l’espace à N dimensions et d'_{ij} distance dans l’espace à K dimensions.

⇒ Problèmes de fausses alarmes et faux rejets

⇒ Utilisation de la réduction des dimensions pour le filtrage

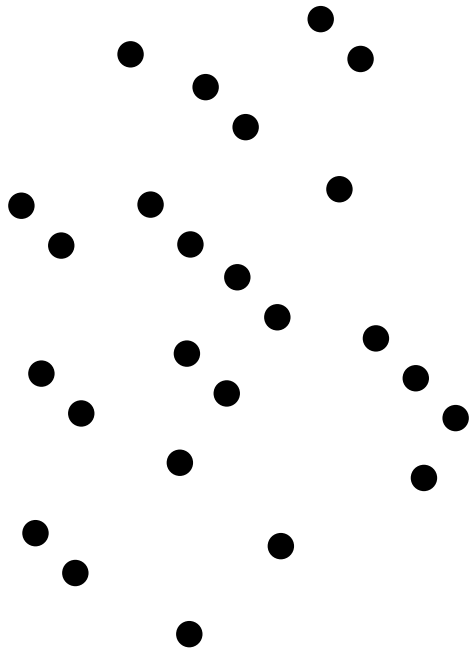
VA-FILE [WSB98,OLD04]

- **Découpage de l'espace en une grille d'hypercubes ou cellules**
- **Approximation de chaque vecteur par une séquence binaire identifiant la cellule le contenant**
- **Stockage en mémoire d'une approximation compacte de l'espace à N dimensions (compression de 10 à 15%)**
- **Calcul de la distance min et max d'un point par rapport à sa cellule $\Rightarrow$ bornes inf et sup de la distance avec le point requête**
- **Utilisation du VA-FILE pour le filtrage**
- **Calcul réel des distances avec les points résultat du filtrage**

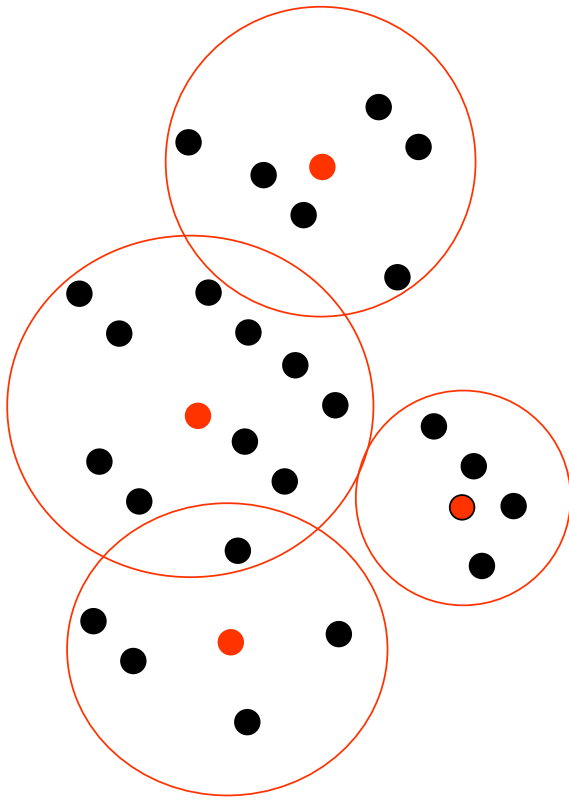
Arbres Métriques

- Indexation des points multidimensionnels par rapport à leur distance relative
- Plus de problème de *curse of dimensionality* (spirale/malédiction dimensionnelle)
- Utilisation de points pivots/points clés (appartenant ou non à la base d'images)

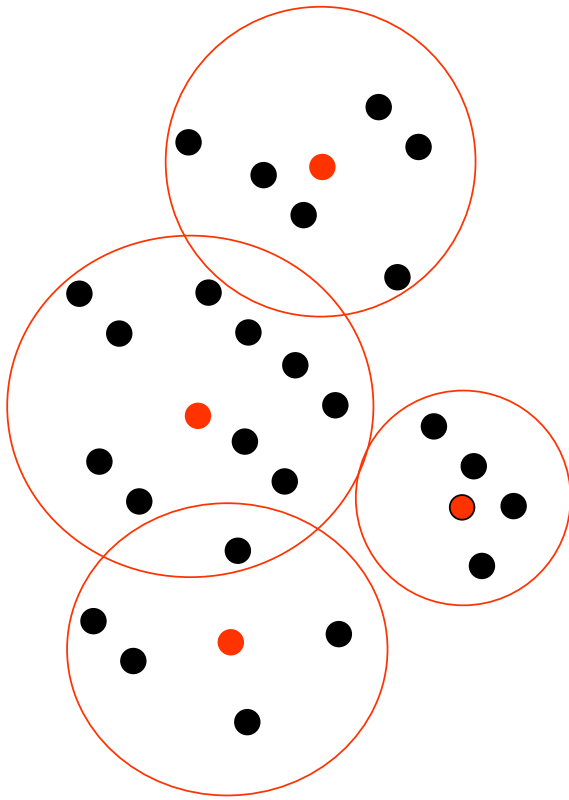
Structure d'index : arbre M



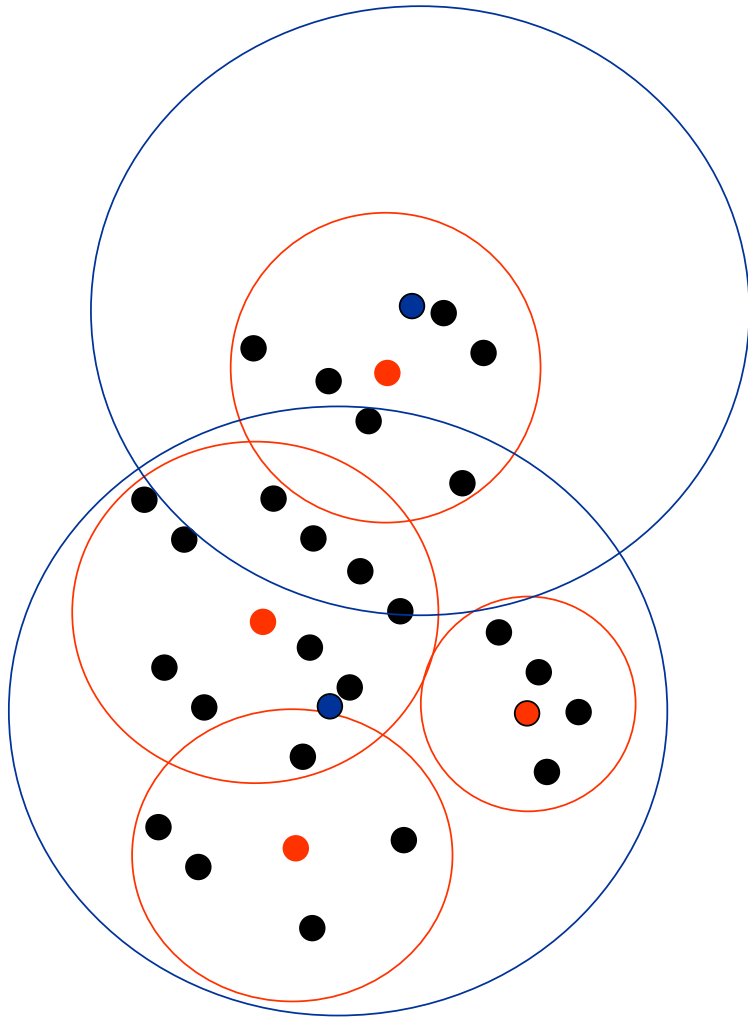
Structure d'index : arbre M



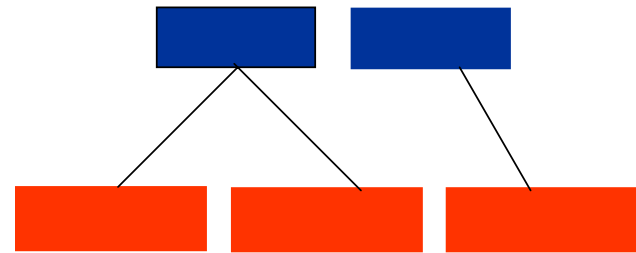
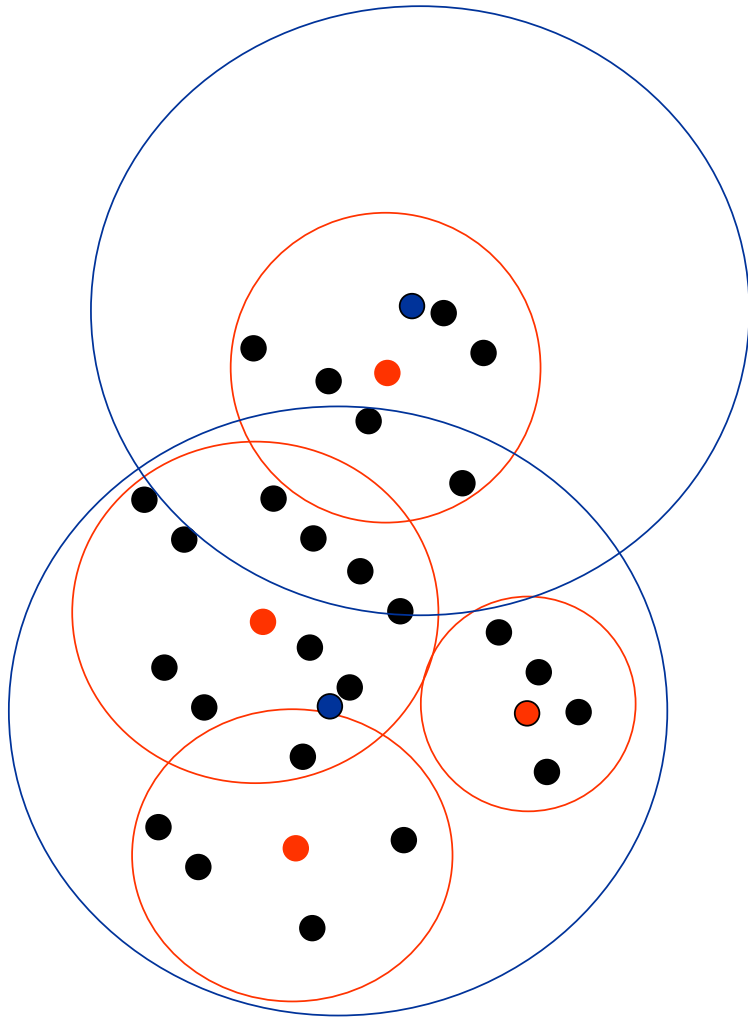
Structure d'index : arbre M



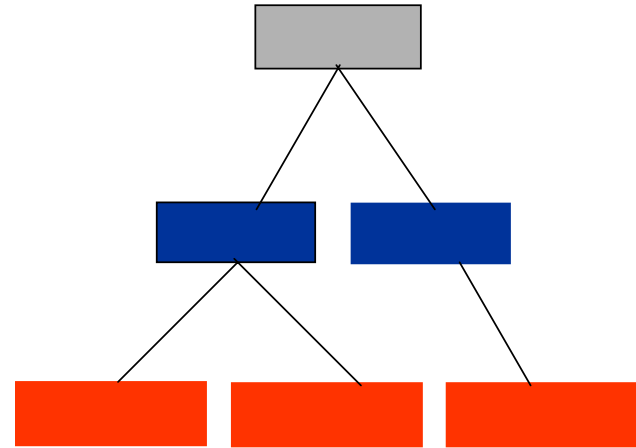
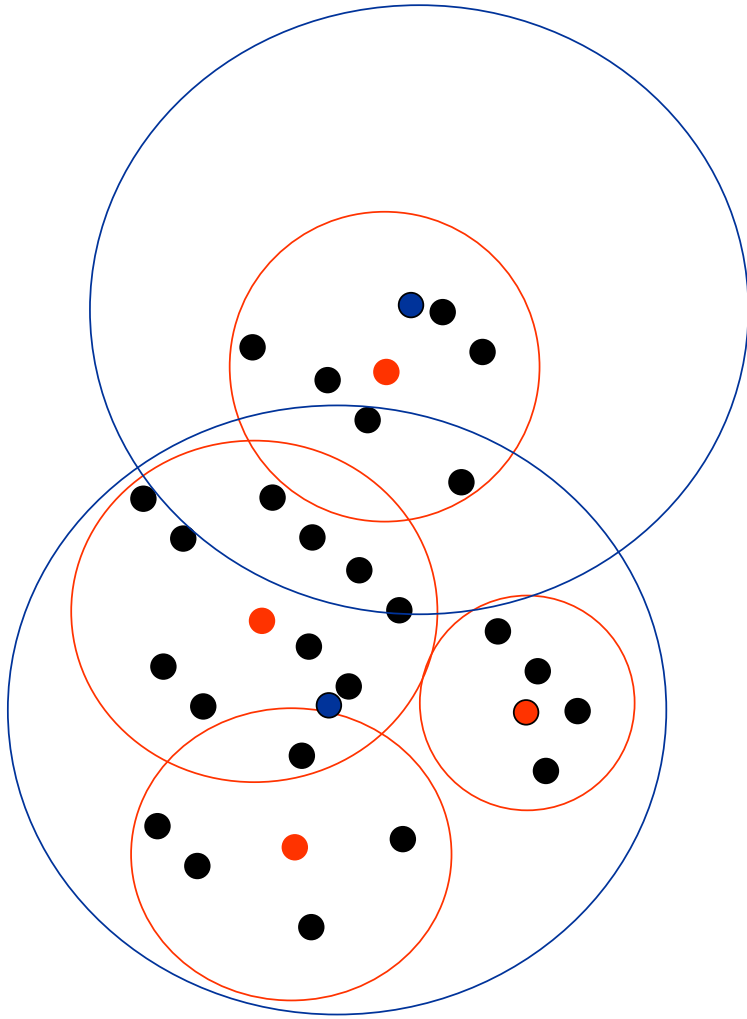
Structure d'index : arbre M



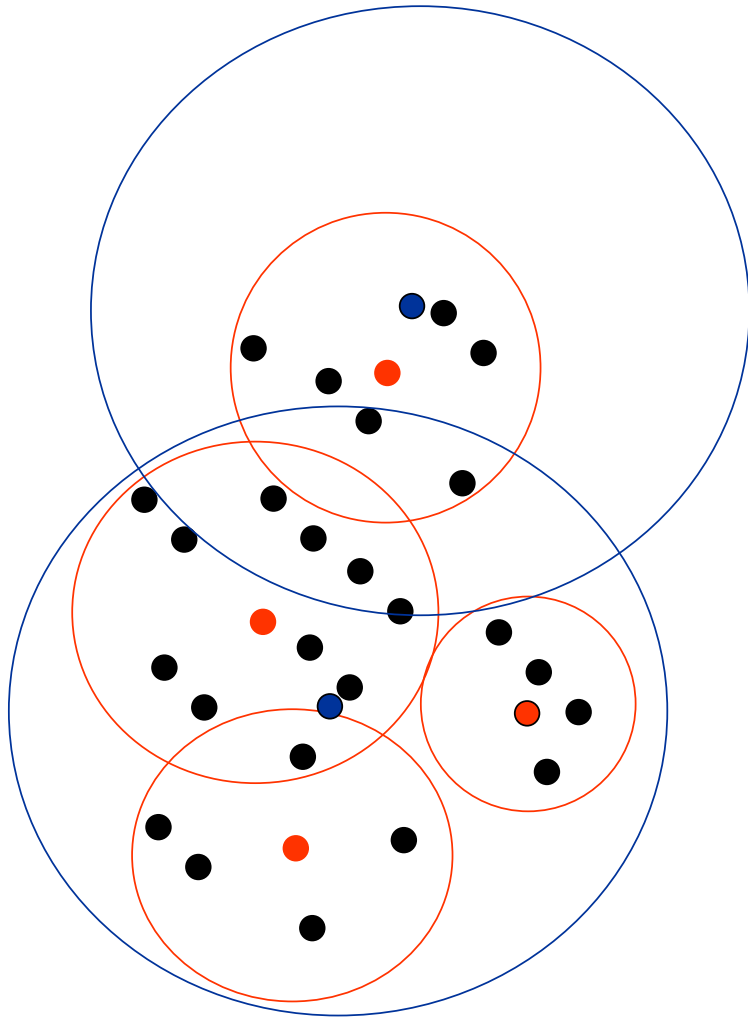
Structure d'index : arbre M



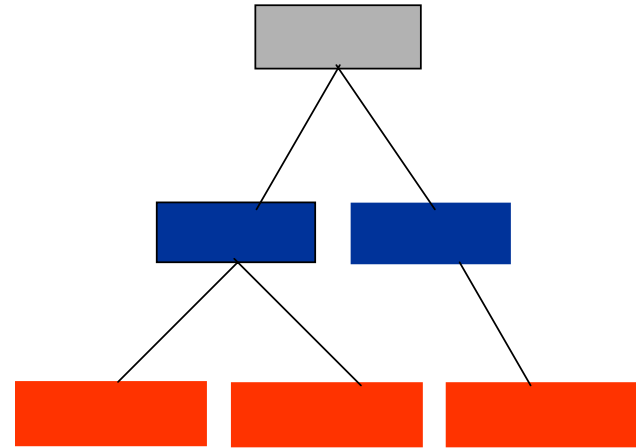
Structure d'index : arbre M



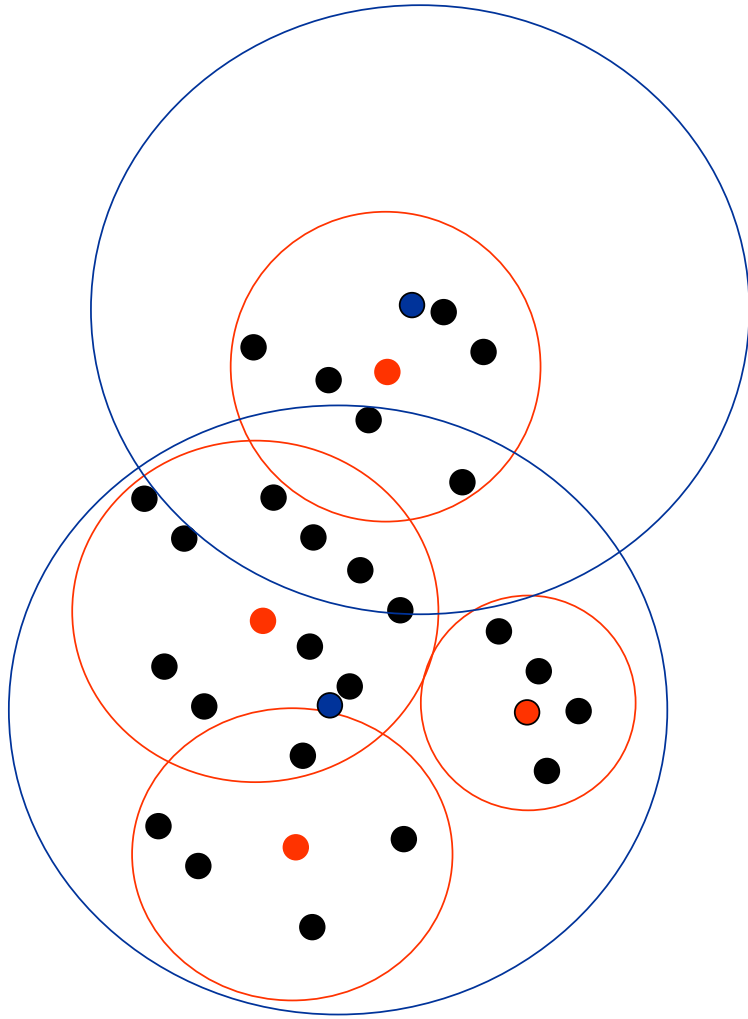
Structure d'index : arbre M



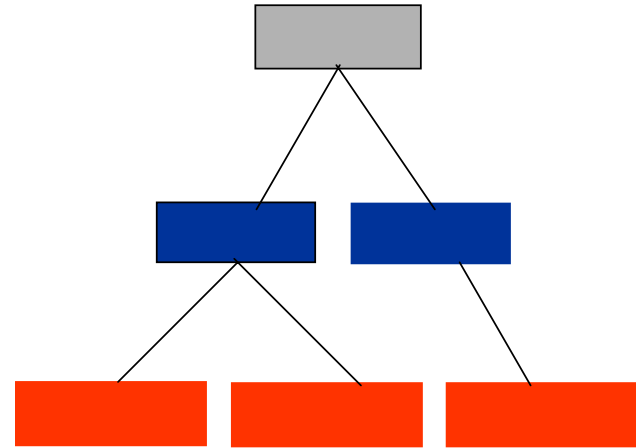
Noeud = (O, pointeur vers la racine, diamètre, distance entre O et son noeud père)



Structure d'index : arbre M

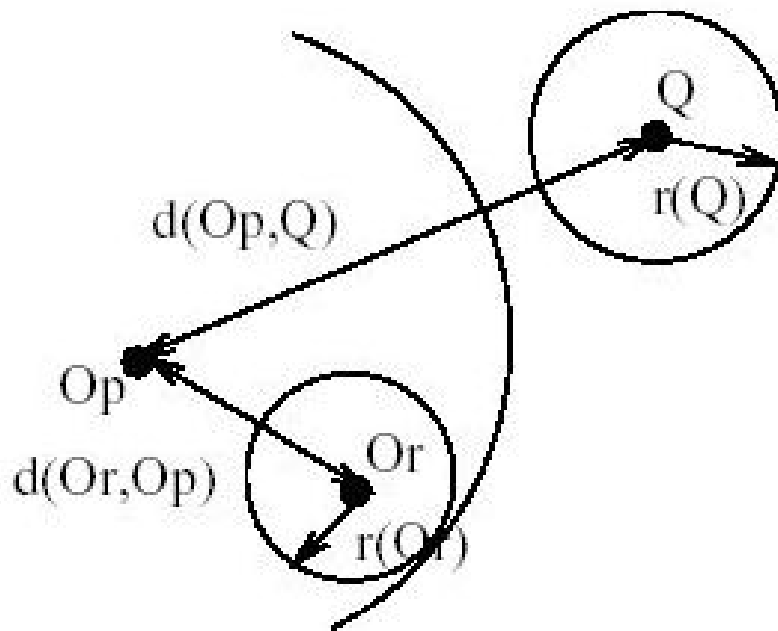


Noeud = (O, pointeur vers la racine, diamètre, distance entre O et son noeud père)

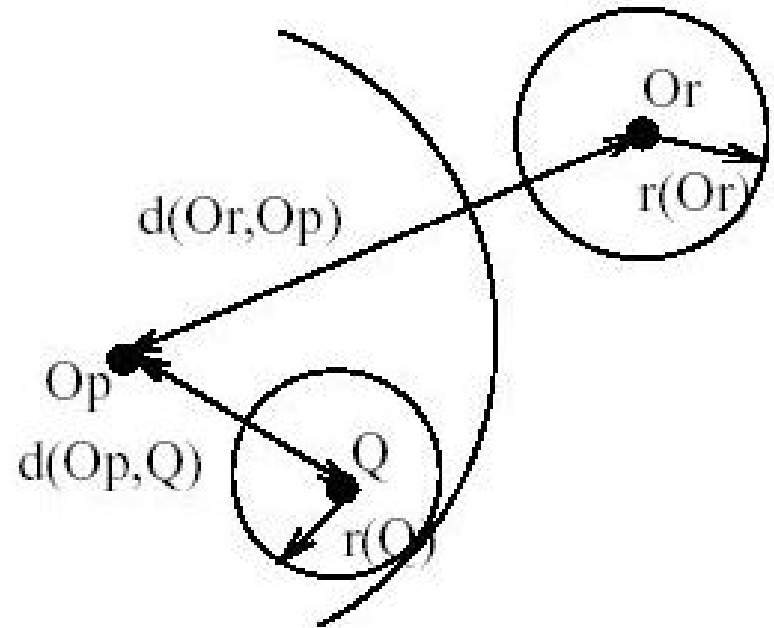


~~~ Arbre B

# Arbre M [CPZ97]



a)



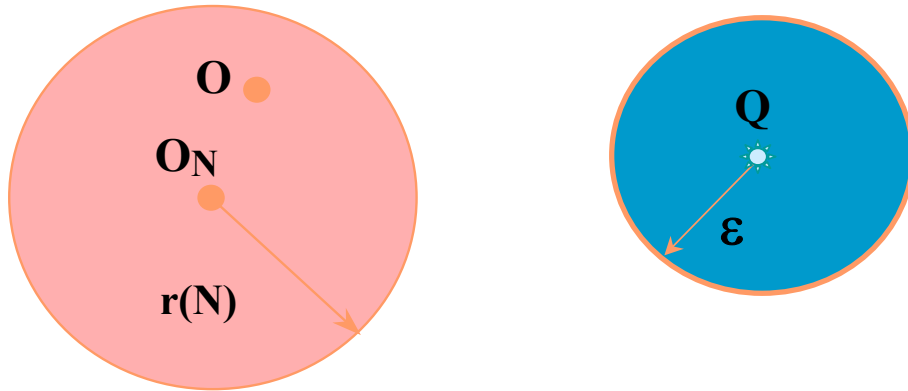
b)

# M-tree: An Efficient Access Method for Similarity Search

- Paolo Ciaccia,
  - Marco Patella,
  - Pavel Zezula
- 
- Present by Qi Zhong(qzhong@ics.uci.edu)

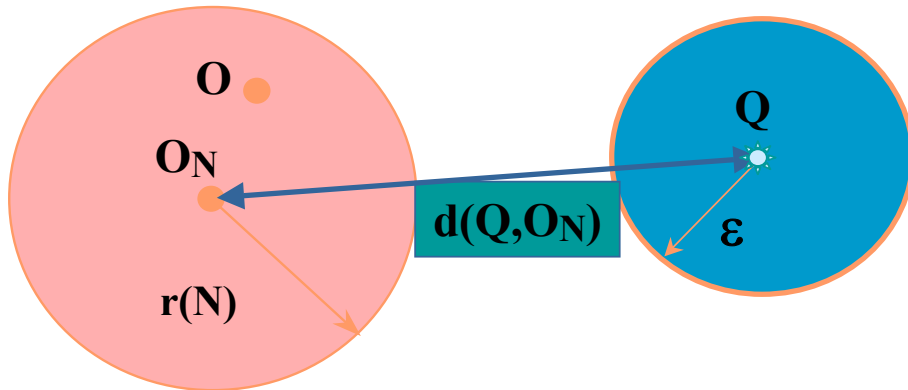
# Searching the M-tree

- Range query:  $d(O, Q) < \epsilon$



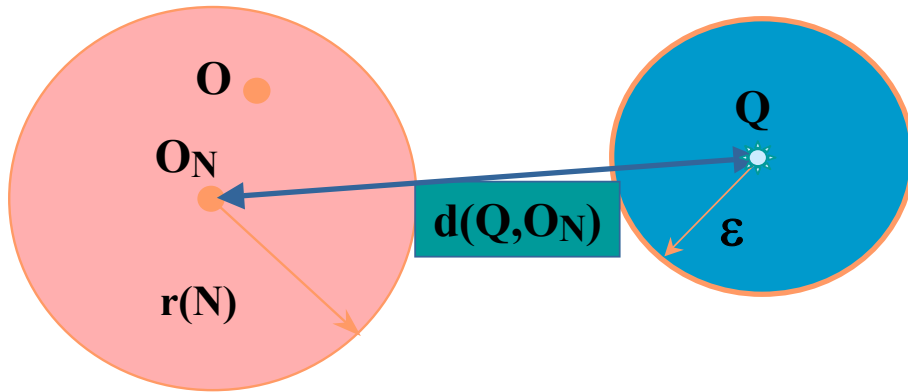
# Searching the M-tree

- Range query:  $d(O, Q) < \epsilon$



# Searching the M-tree

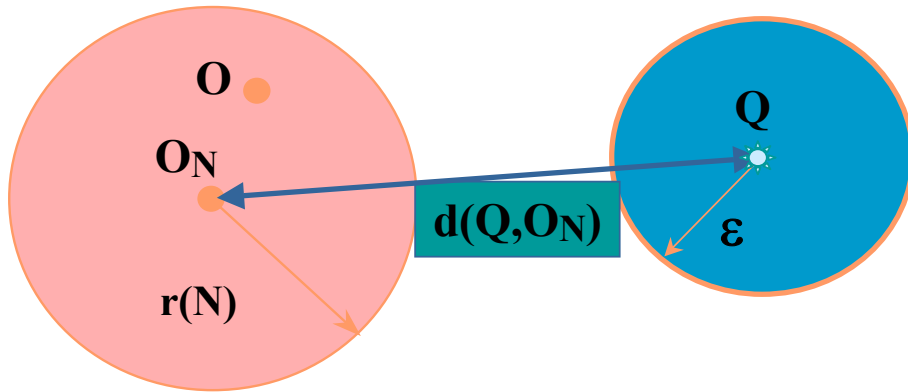
- Range query:  $d(O, Q) < \epsilon$



Prune node N iff  $d(Q, O_N) > r(N) + \epsilon$

# Searching the M-tree

■ Range query:  $d(O, Q) < \epsilon$

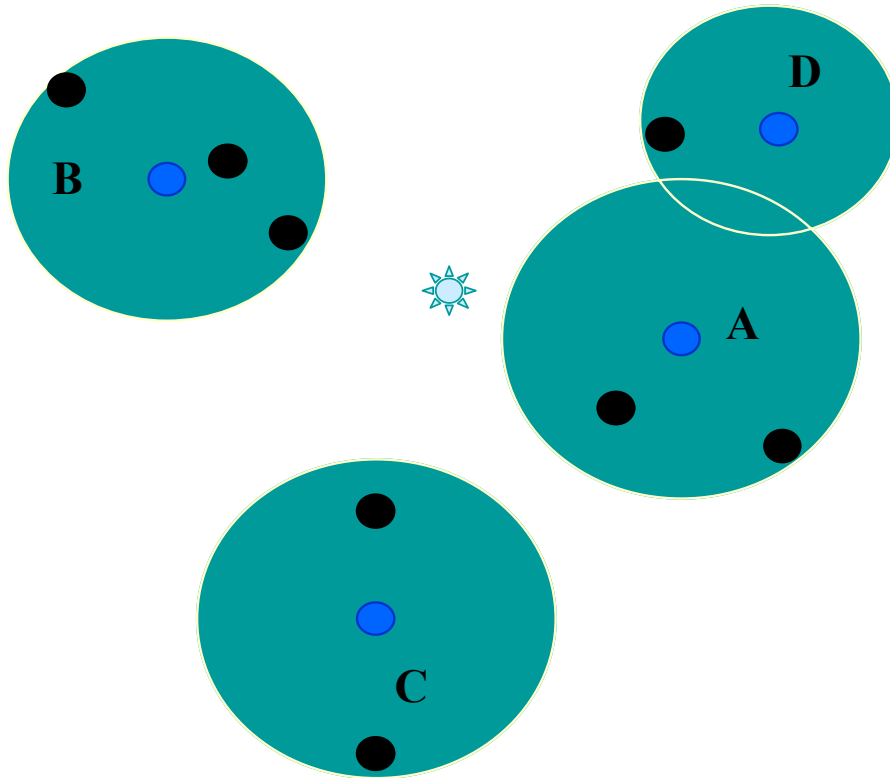


$$\begin{aligned} d(Q, O) &\geq d(Q, O_N) - d(O, O_N) \\ &\geq d(Q, O_N) - r(N) \\ &> r(N) + \epsilon - r(N) \\ &= \epsilon \end{aligned}$$

Prune node N iff  $d(Q, O_N) > r(N) + \epsilon$

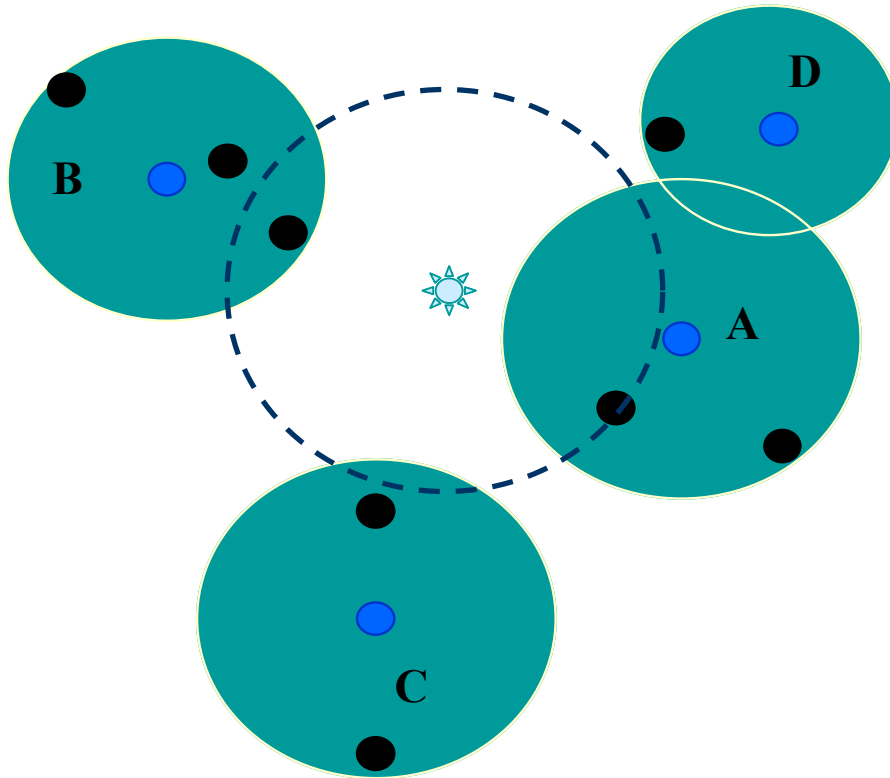
# Nearest Neighbor(s) Search

- **Priority queue: read the node with minimum  $d_{\text{MIN}}(Q, O_r)$** 
  - ◆ If  $O_r$  is not leaf node,  $d_{\text{MIN}}(Q, O_r) = d(Q, O_r) - r(O_r)$ ,  $d_{\text{MAX}}(Q, O_r) = d(Q, O_r) + r(O_r)$
  - ◆ If  $O_r$  is leaf node,  $d_{\text{min}} = d_{\text{max}} = d(Q, O_r)$
- **Prune  $O_i$  iff  $d_{\text{min}}(Q, O_i) > d_{\text{max}}(Q, O_r)$**



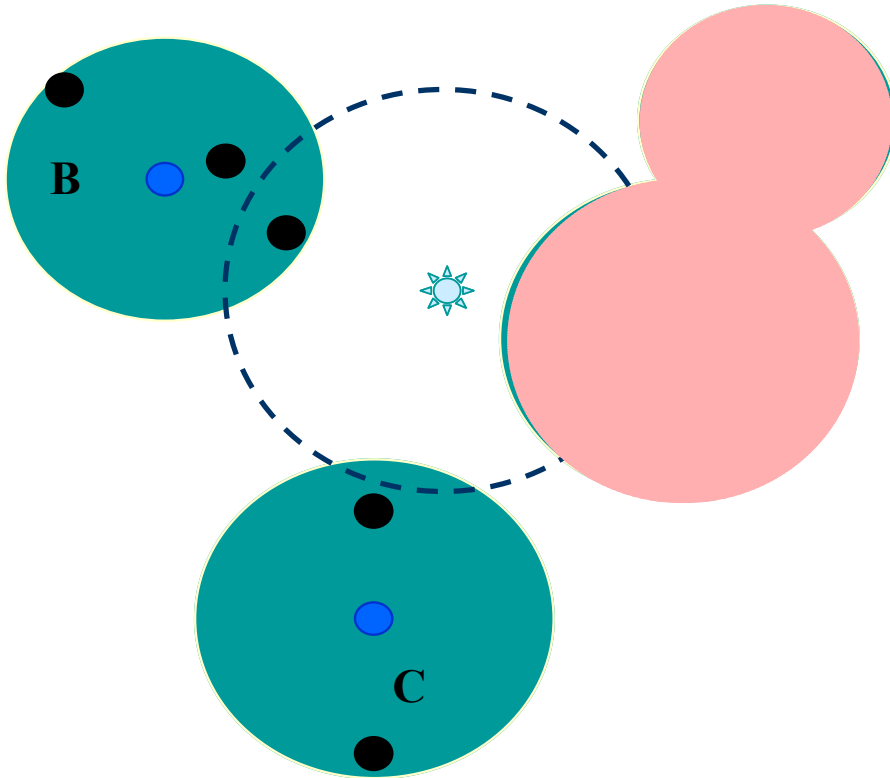
# Nearest Neighbor(s) Search

- **Priority queue:** read the node with minimum  $d_{\text{MIN}}(Q, O_r)$ 
  - ◆ If  $O_r$  is not leaf node,  $d_{\text{MIN}}(Q, O_r) = d(Q, O_r) - r(O_r)$ ,  $d_{\text{MAX}}(Q, O_r) = d(Q, O_r) + r(O_r)$
  - ◆ If  $O_r$  is leaf node,  $d_{\text{min}} = d_{\text{max}} = d(Q, O_r)$
- **Prune**  $O_i$  iff  $d_{\text{min}}(Q, O_i) > d_{\text{max}}(Q, O_r)$



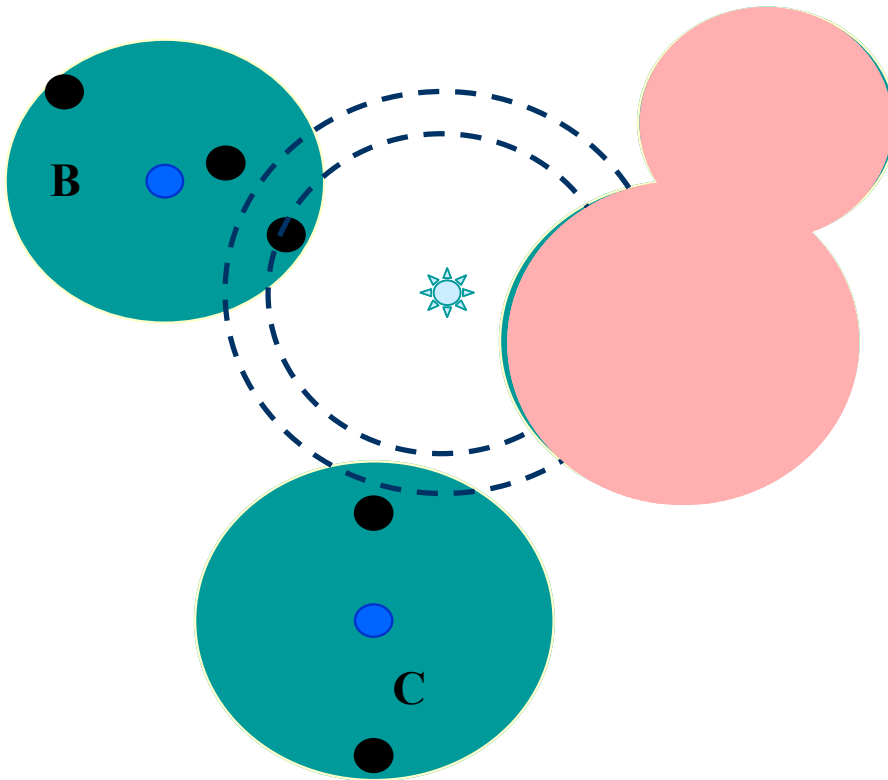
# Nearest Neighbor(s) Search

- **Priority queue: read the node with minimum  $d_{\min}(Q, O_r)$** 
  - ◆ If  $O_r$  is not leaf node,  $d_{\min}(Q, O_r) = d(Q, O_r) - r(O_r)$ ,  $d_{\max}(Q, O_r) = d(Q, O_r) + r(O_r)$
  - ◆ If  $O_r$  is leaf node,  $d_{\min} = d_{\max} = d(Q, O_r)$
- **Prune  $O_i$  iff  $d_{\min}(Q, O_i) > d_{\max}(Q, O_r)$**



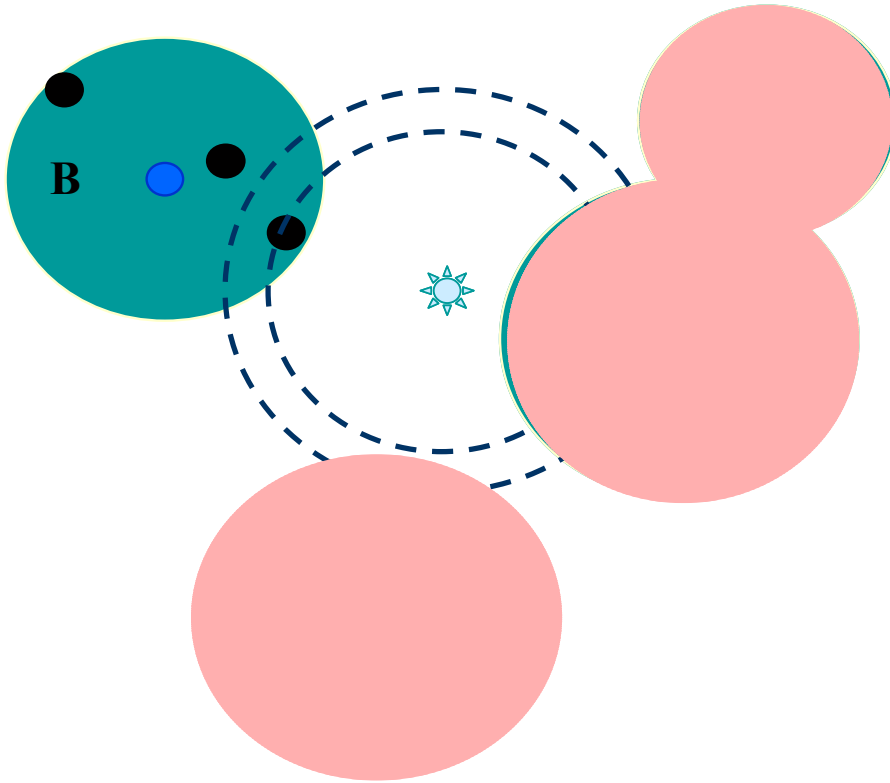
# Nearest Neighbor(s) Search

- **Priority queue: read the node with minimum  $d_{\min}(Q, O_r)$** 
  - ◆ If  $O_r$  is not leaf node,  $d_{\min}(Q, O_r) = d(Q, O_r) - r(O_r)$ ,  $d_{\max}(Q, O_r) = d(Q, O_r) + r(O_r)$
  - ◆ If  $O_r$  is leaf node,  $d_{\min} = d_{\max} = d(Q, O_r)$
- **Prune  $O_i$  iff  $d_{\min}(Q, O_i) > d_{\max}(Q, O_r)$**



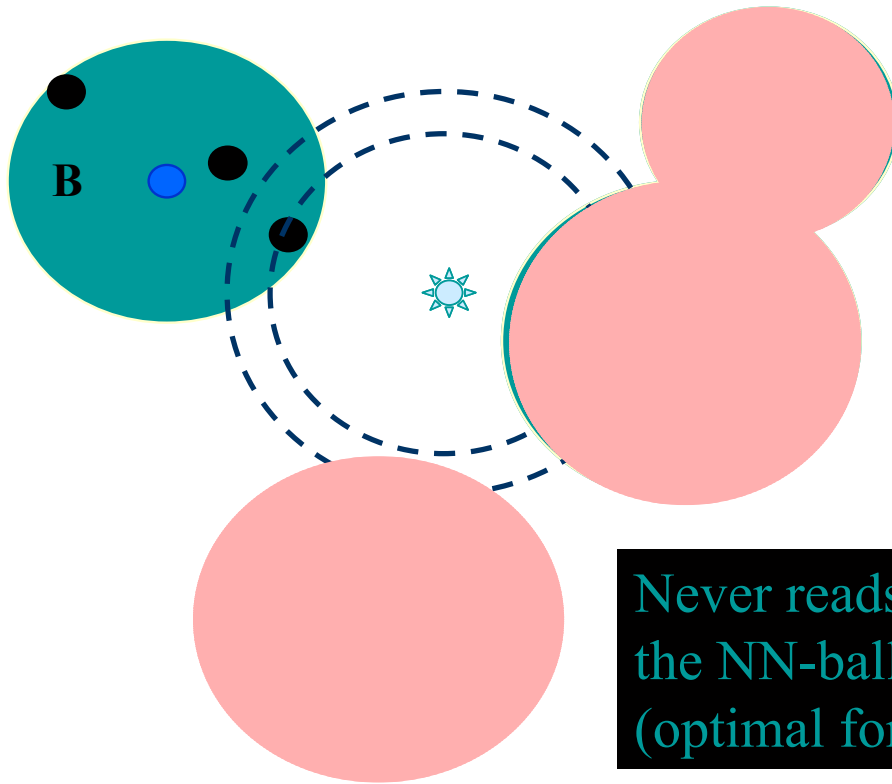
# Nearest Neighbor(s) Search

- **Priority queue: read the node with minimum  $d_{\text{MIN}}(Q, O_r)$** 
  - ◆ If  $O_r$  is not leaf node,  $d_{\text{MIN}}(Q, O_r) = d(Q, O_r) - r(O_r)$ ,  $d_{\text{MAX}}(Q, O_r) = d(Q, O_r) + r(O_r)$
  - ◆ If  $O_r$  is leaf node,  $d_{\text{min}} = d_{\text{max}} = d(Q, O_r)$
- **Prune  $O_i$  iff  $d_{\text{min}}(Q, O_i) > d_{\text{max}}(Q, O_r)$**



# Nearest Neighbor(s) Search

- **Priority queue: read the node with minimum  $d_{\text{MIN}}(Q, O_r)$** 
  - ◆ If  $O_r$  is not leaf node,  $d_{\text{MIN}}(Q, O_r) = d(Q, O_r) - r(O_r)$ ,  $d_{\text{MAX}}(Q, O_r) = d(Q, O_r) + r(O_r)$
  - ◆ If  $O_r$  is leaf node,  $d_{\text{min}} = d_{\text{max}} = d(Q, O_r)$
- **Prune  $O_i$  iff  $d_{\text{min}}(Q, O_i) > d_{\text{max}}(Q, O_r)$**



Never reads nodes that do not intersect  
the NN-ball  
(optimal for metric spaces)

# Structures multidimensionnelles : quelques références

- B. Salzberg and V.J. Tsotras. *A Comparison of Access for Time Evolving Data*. ACM Computing Surveys, 31(2):158--221, 1999.  
<ftp://ftp.ccs.neu.edu/pub/people/salzberg/tempsurvey.ps.gz>
- E. Chávez, G. Navarro, R.A. Baeza-Yates, and J.L. Marroquín. *Searching in metric spaces*. ACM Computing Surveys, 33(3):273--321, 2001.
- V. Castelli. *Image Databases : Search and Retrieval of Digital Imagery, chapter 14 - Multidimensional Indexing Structures for Content-Based Retrieval*, pages 373--433. Wiley Inter-Science, 2002. V. Castelli and L.D. Bergman (Eds) - ISBN: 0-471-32116-8
- S-A. Berrani, L. Amsaleg et P. Gros. *Recherche par similarité dans les bases de données multidimensionnelles : panorama des techniques d'indexation*. ISI 7/2002 Bases de données multimédia, pp. 9-44, 2002

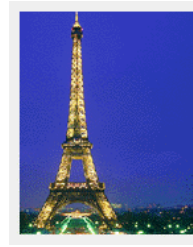
# Récapitulatif

**Qu'est-ce qu'une base d'images ?**

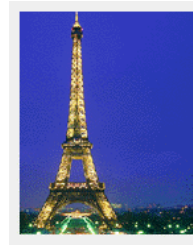
- **Un ensemble d'images**
- **Des descripteurs extraits par traitement d'images**
- **Des annotations/mot-clés/méta-données définis par les utilisateurs**
- **Des structures d'index propres aux descripteurs ou aux méta-données**
- **Des mécanismes de recherche et de comparaison d'images**
- **Des mécanismes d'interrogation et de visualisation du résultat des requêtes.**

# Recherche d'images basée sur le contenu

# Recherche d'images basée sur le contenu

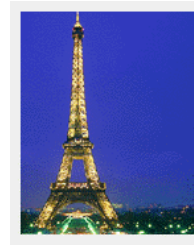


# Recherche d'images basée sur le contenu



- **Extraction des caractéristiques de l'image** : couleur, texture, forme, contraintes spatiales, régions

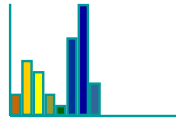
# Recherche d'images basée sur le contenu



forme



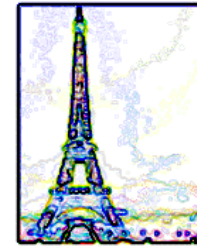
texture



couleur



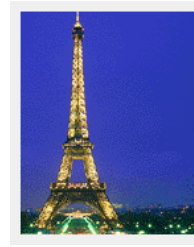
contraintes  
spatiales



régions

- **Extraction des caractéristiques de l'image** : couleur, texture, forme, contraintes spatiales, régions

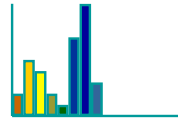
# Recherche d'images basée sur le contenu



forme



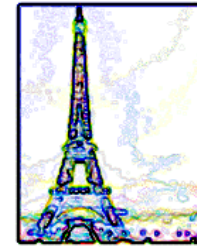
texture



couleur



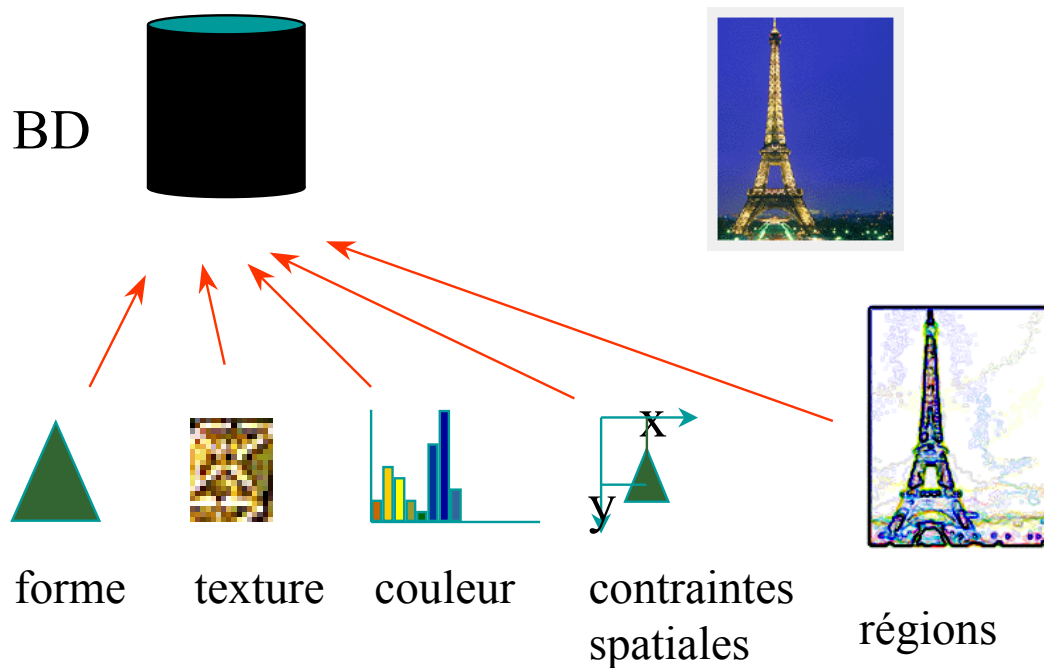
contraintes  
spatiales



régions

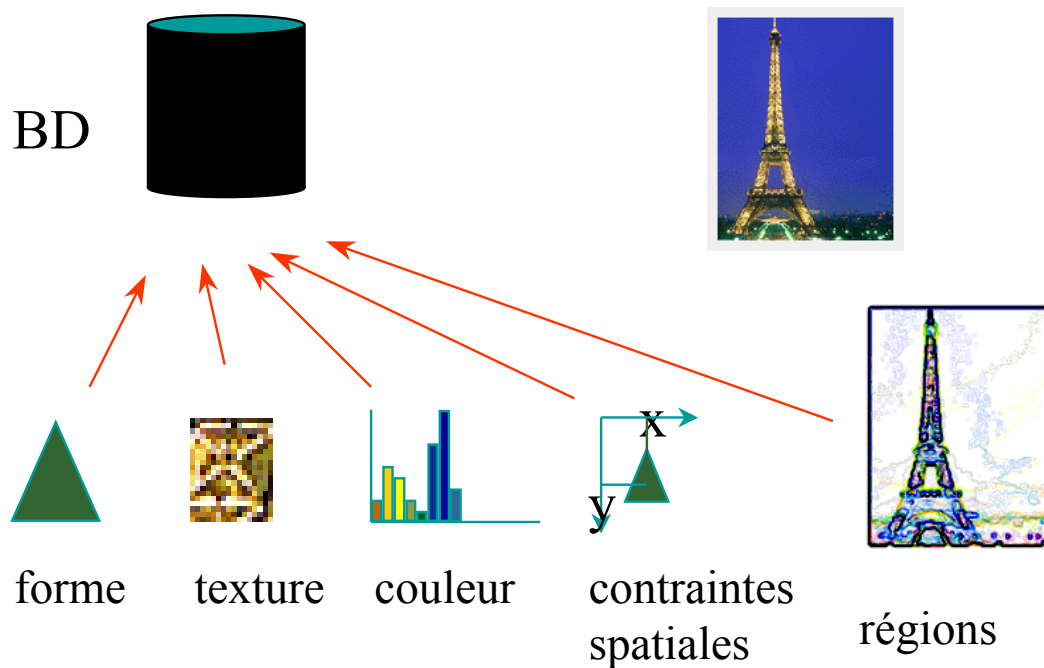
- **Extraction des caractéristiques de l'image** : couleur, texture, forme, contraintes spatiales, régions
- **Représentation de ces caractéristiques** en vue de leur stockage dans la base de données

# Recherche d'images basée sur le contenu



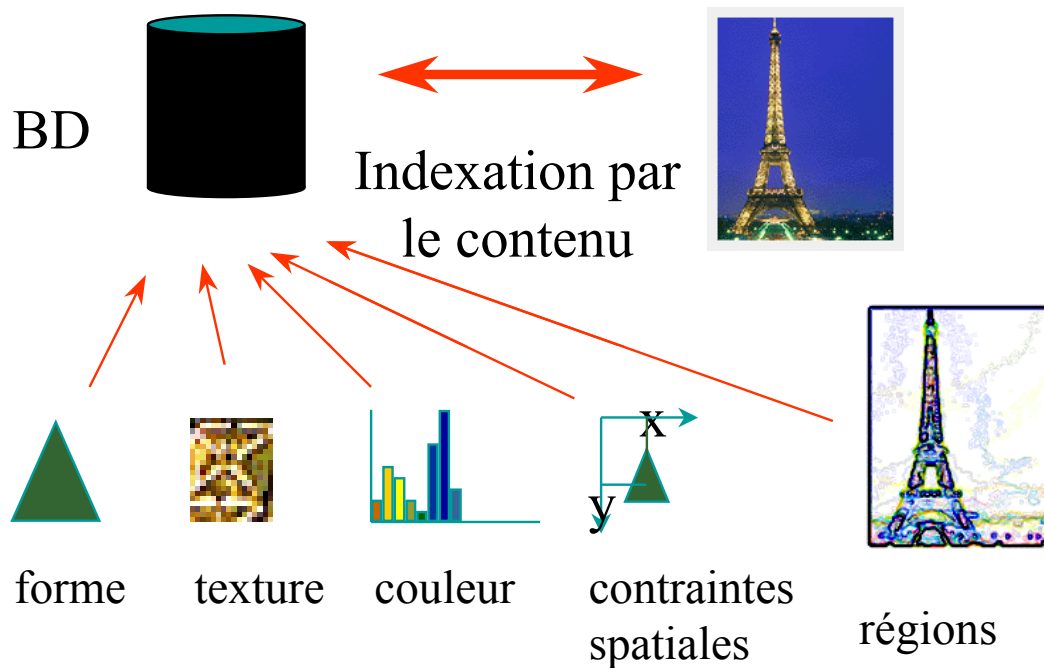
- **Extraction des caractéristiques de l'image** : couleur, texture, forme, contraintes spatiales, régions
- **Représentation de ces caractéristiques** en vue de leur stockage dans la base de données

# Recherche d'images basée sur le contenu



- **Extraction des caractéristiques de l'image** : couleur, texture, forme, contraintes spatiales, régions
- **Représentation de ces caractéristiques** en vue de leur stockage dans la base de données
- **Recherche sur ces informations extraites**

# Recherche d'images basée sur le contenu



- **Extraction des caractéristiques de l'image** : couleur, texture, forme, contraintes spatiales, régions
- **Représentation de ces caractéristiques** en vue de leur stockage dans la base de données
- **Recherche sur ces informations extraites**

# Recherche d'images

- **Construction d'index :**
  - ◆ A partir des descripteurs (points à  $n$  dimensions dans un espace de caractéristiques)
  - ◆ A partir de fonctions de distance  $d$  (Minkowsky)
- **A partir d'une image requête (de la base ou non) :**
  - ◆ Trouver, à l'aide de l'index, les images les plus proches de l'image requête en utilisant  $d$
  - ◆ Afficher les images de manière ordonnées (des plus similaires au moins similaires)

# Structure d'index : arbres R, SS, SR ...

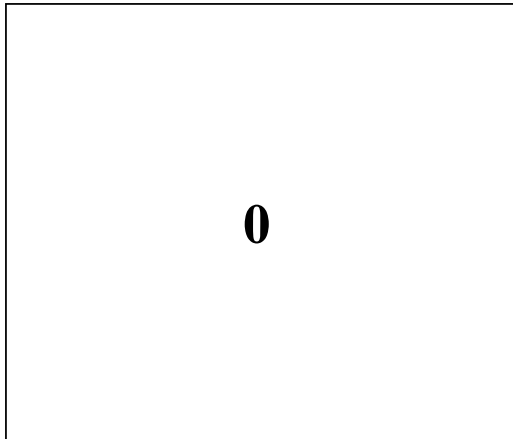
- Placement des images dans leur espace de caractéristiques
  - Découpage de l'espace multidimensionnel des caractéristiques
  - Problèmes de malédiction de la dimension
- ⇒ **Placement des images par rapport à leur distance à des points pivots : arbre M, Slim-tree, arbre A**

# Recherche et indexation d'images similaires à l'aide d'un arbre quaternaire

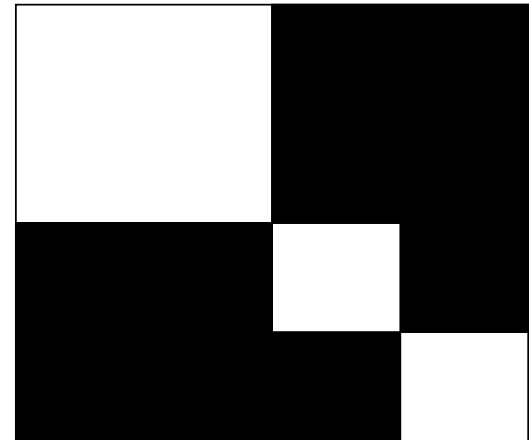
- Définition d'un arbre quaternaire
- Similarité d'images représentées par des arbres quaternaires
- Indexation des images représentées par des arbres quaternaires

# Image organisée en arbre quaternaire

Identification des quadrants

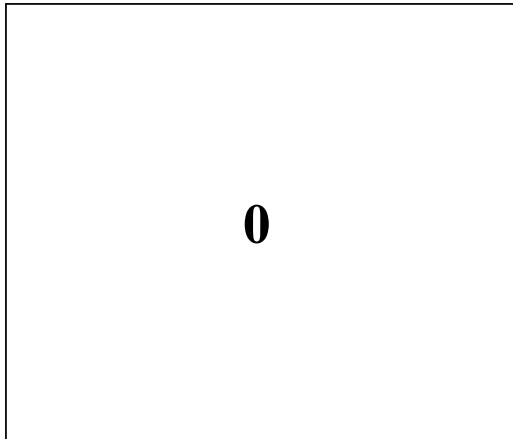


Image



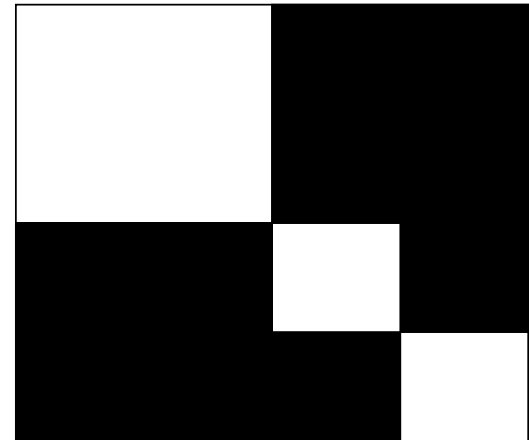
# Image organisée en arbre quaternaire

Identification des quadrants



Arbre quaternaire  
représentant l'image

Image



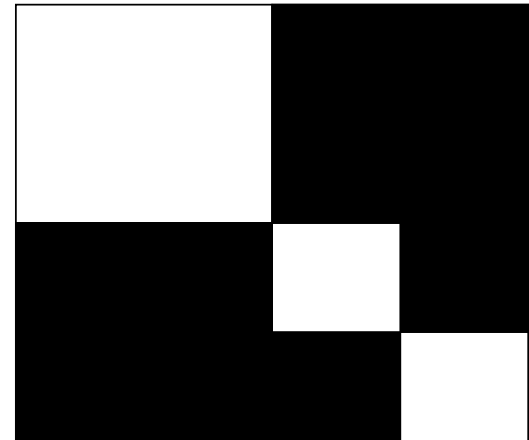
# Image organisée en arbre quaternaire

Identification des quadrants

|    |    |
|----|----|
| 00 | 01 |
| 02 | 03 |

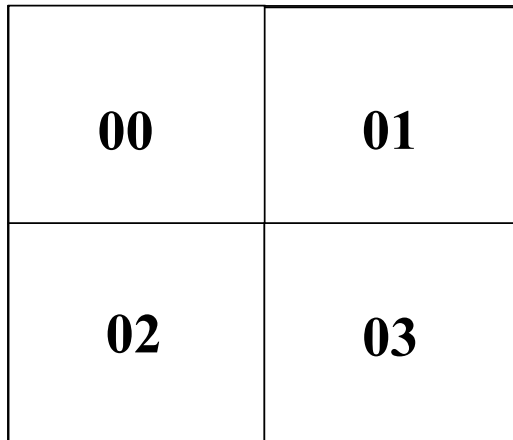
Arbre quaternaire  
représentant l'image

Image

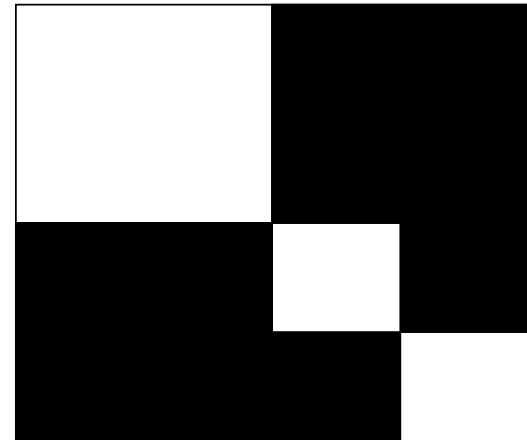


# Image organisée en arbre quaternaire

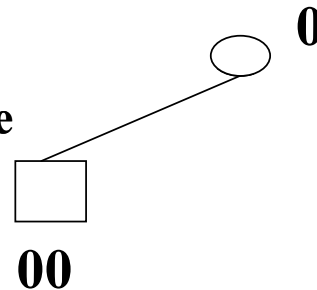
Identification des quadrants



Image

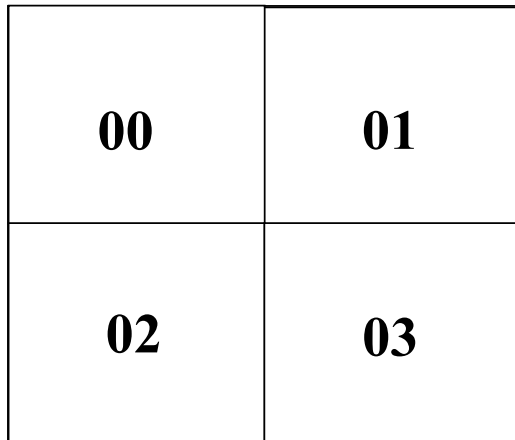


Arbre quaternaire  
représentant l'image

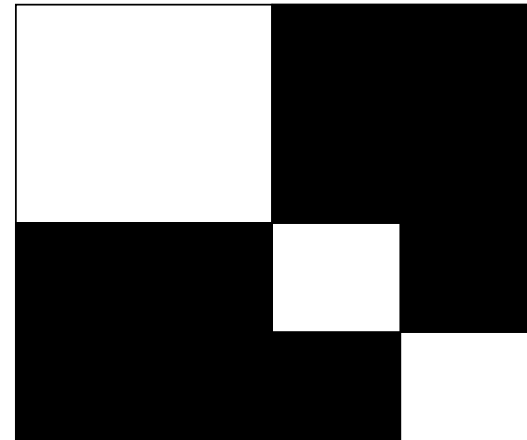


# Image organisée en arbre quaternaire

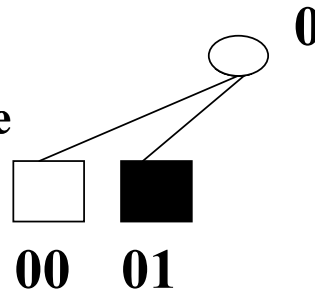
Identification des quadrants



Image

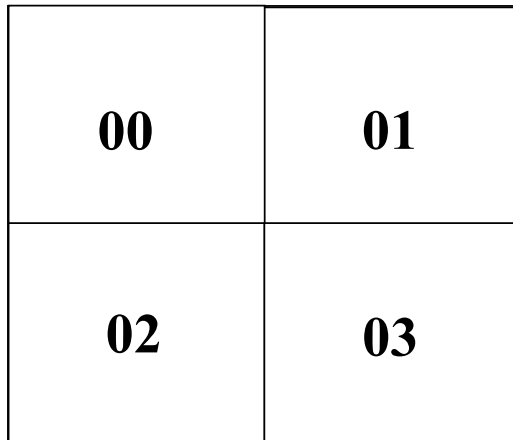


Arbre quaternaire  
représentant l'image

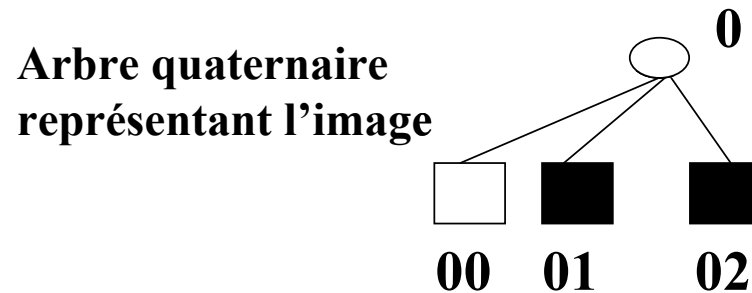
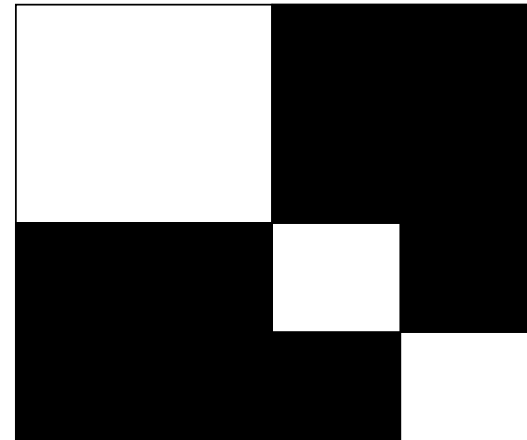


# Image organisée en arbre quaternaire

Identification des quadrants

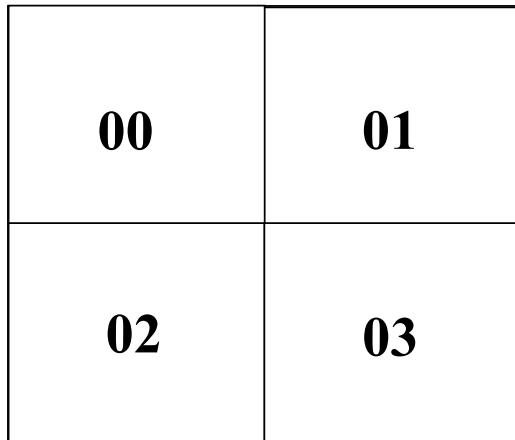


Image

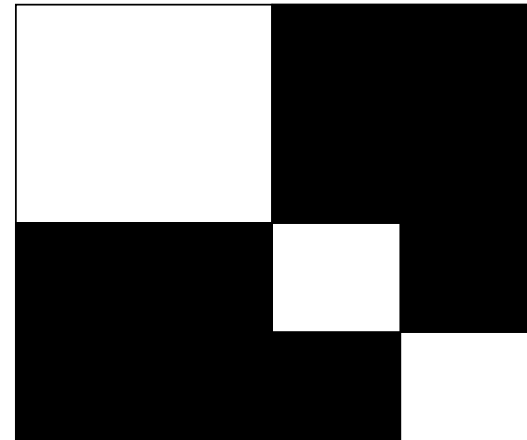


# Image organisée en arbre quaternaire

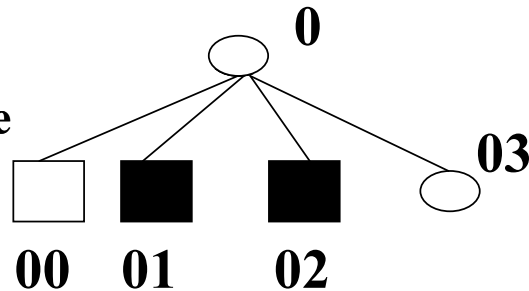
Identification des quadrants



Image

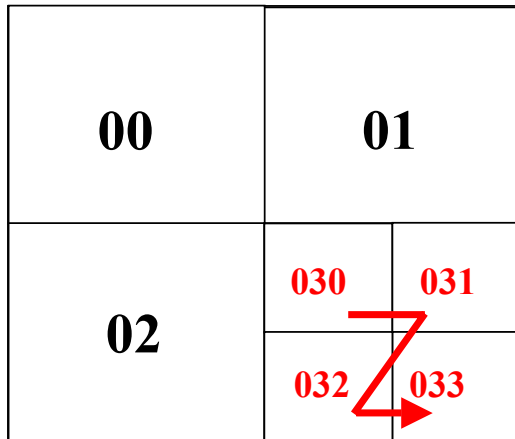


Arbre quaternaire  
représentant l'image

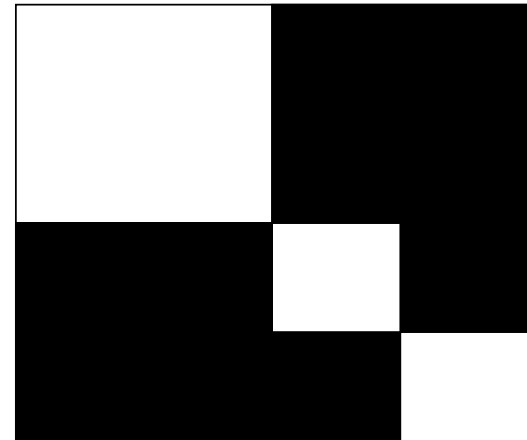


# Image organisée en arbre quaternaire

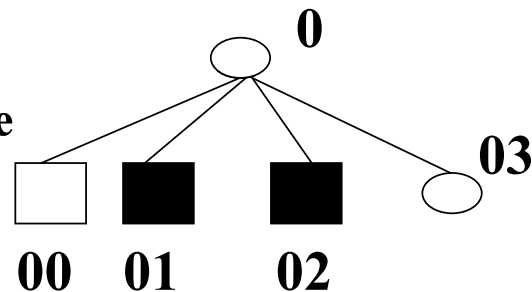
Identification des quadrants



Image

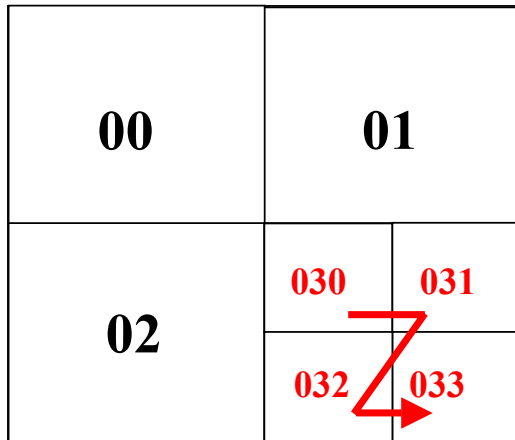


Arbre quaternaire  
représentant l'image

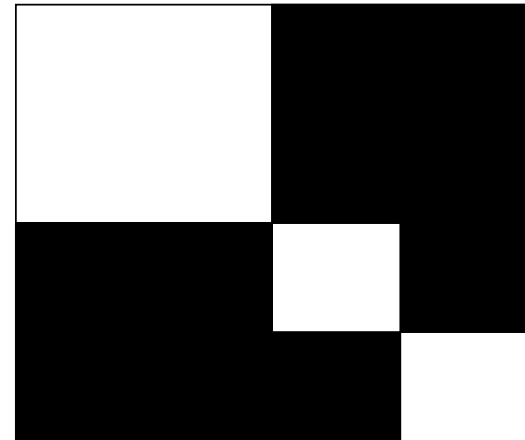


# Image organisée en arbre quaternaire

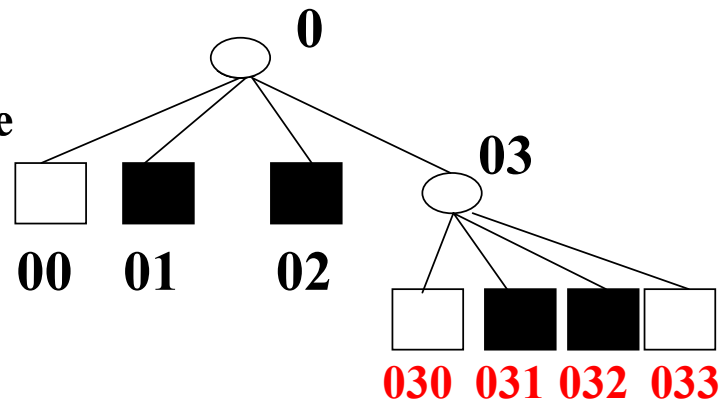
Identification des quadrants



Image



Arbre quaternaire  
représentant l'image

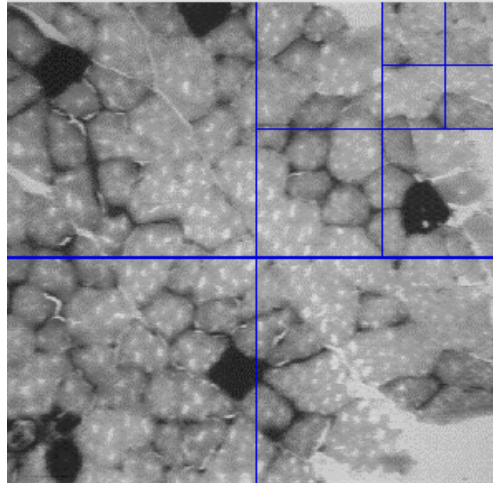


# Arbre Quaternaire

**Identification des quadrants**

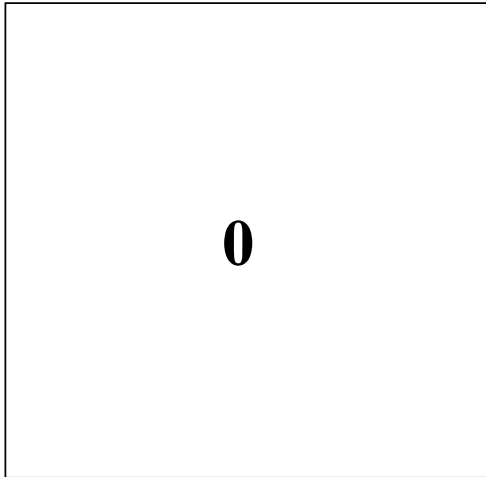
**Image**

**Arbre quaternaire représentant  
l'image**

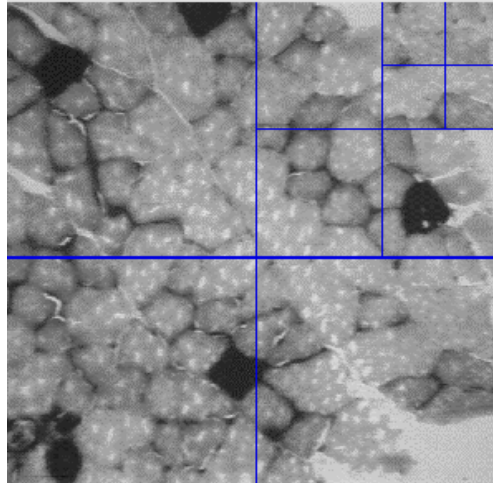


# Arbre Quaternaire

Identification des quadrants



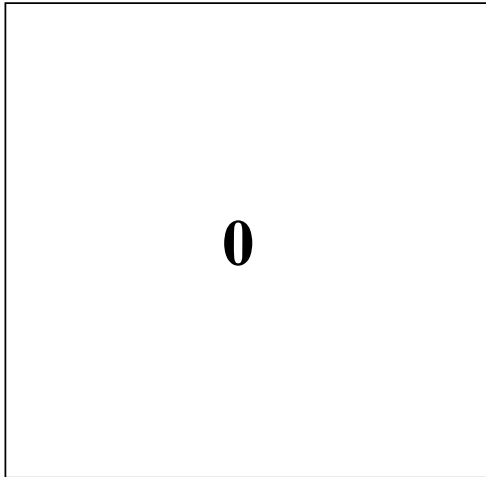
Image



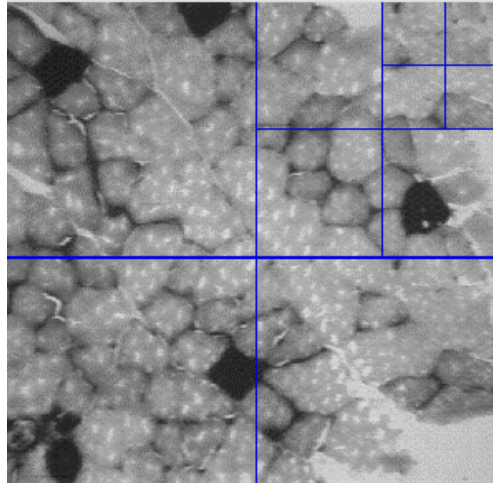
Arbre quaternaire représentant  
l'image

# Arbre Quaternaire

Identification des quadrants



Image



Arbre quaternaire représentant  
l'image

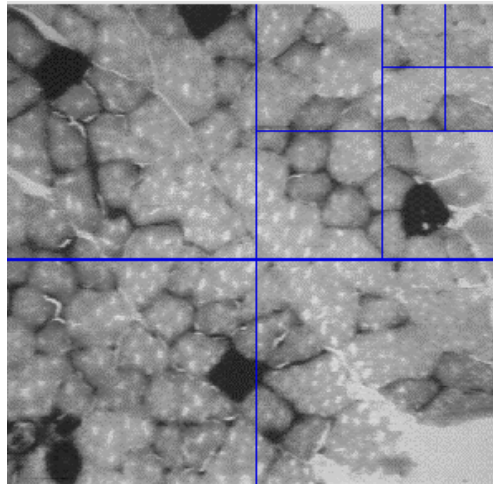


# Arbre Quaternaire

Identification des quadrants

|    |    |
|----|----|
| 00 | 01 |
| 02 | 03 |

Image

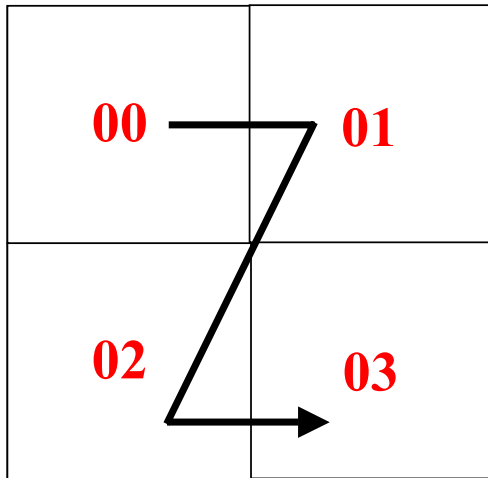


Arbre quaternaire représentant  
l'image

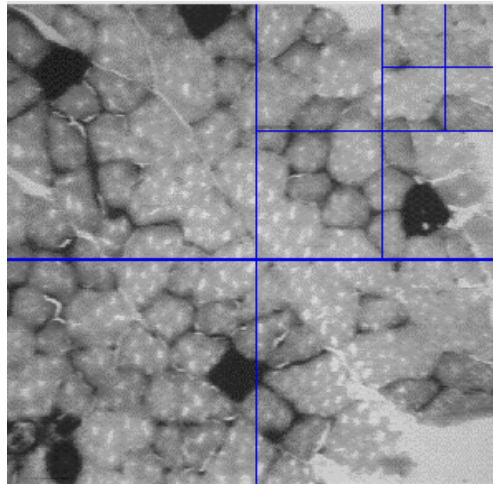
○ 0

# Arbre Quaternaire

Identification des quadrants



Image

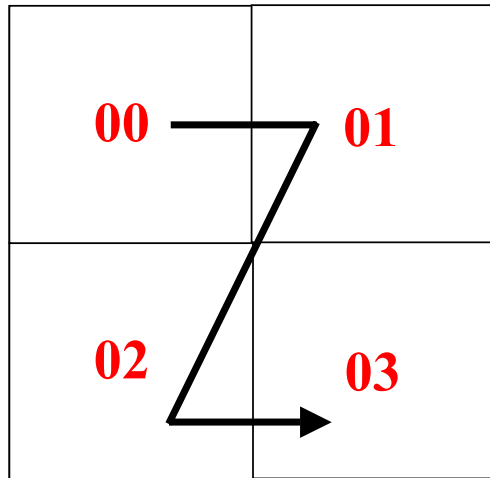


Arbre quaternaire représentant  
l'image

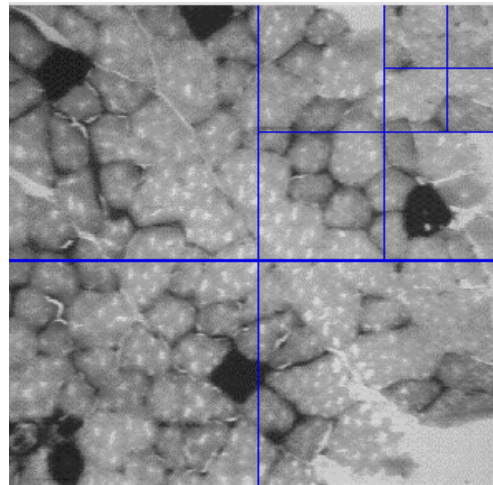
○ 0

# Arbre Quaternaire

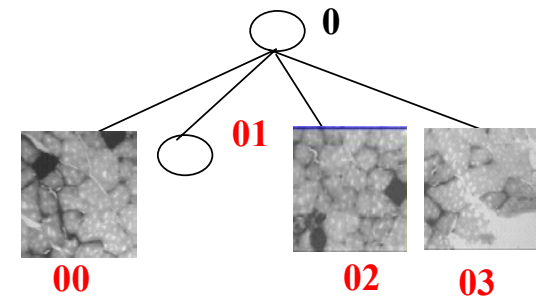
Identification des quadrants



Image

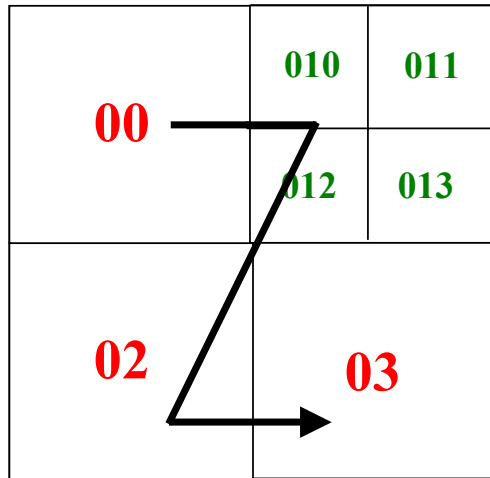


Arbre quaternaire représentant l'image

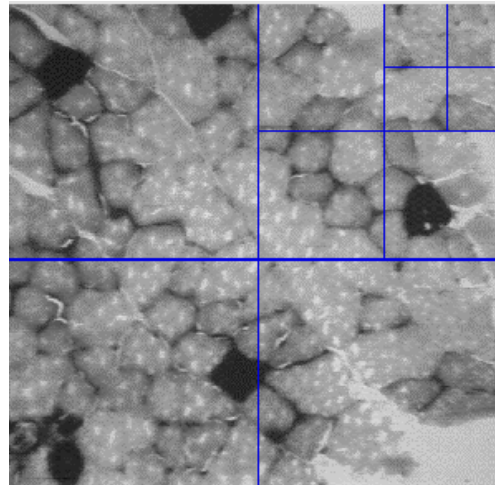


# Arbre Quaternaire

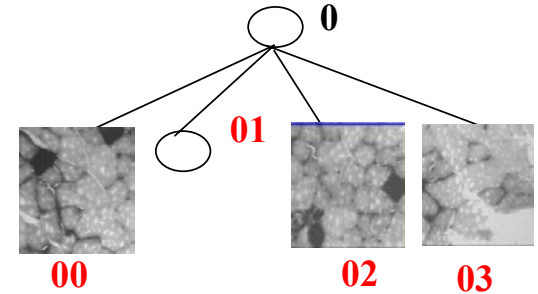
## Identification des quadrants



## Image

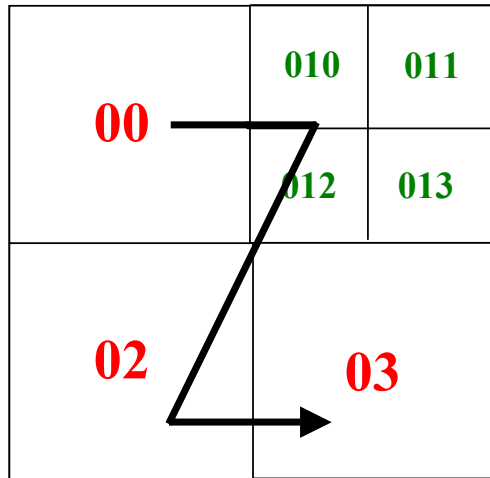


## Arbre quaternaire représentant l'image

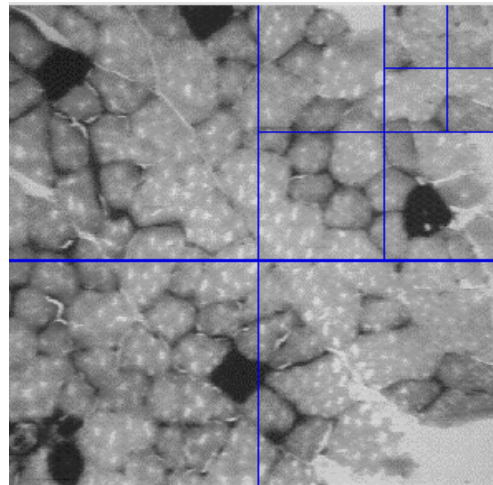


# Arbre Quaternaire

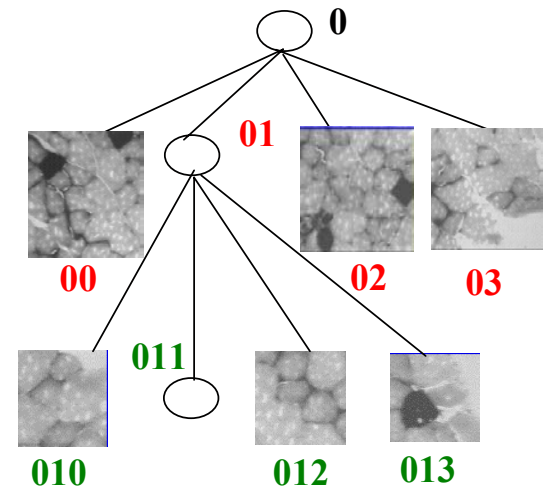
Identification des quadrants



Image

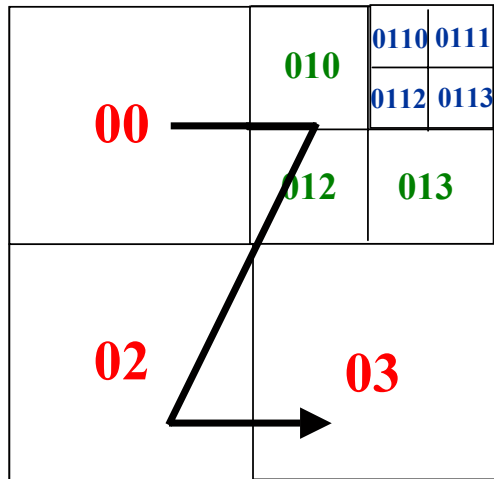


Arbre quaternaire représentant l'image

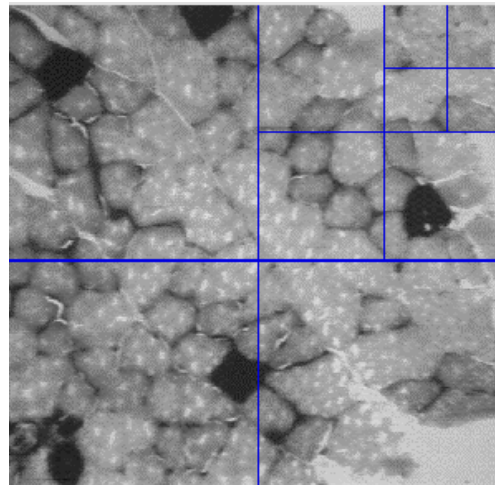


# Arbre Quaternaire

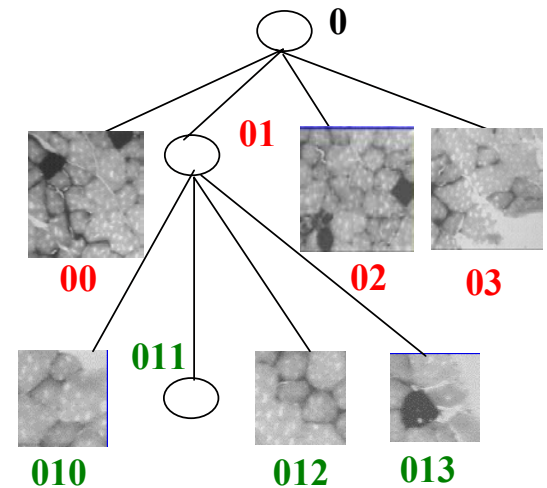
Identification des quadrants



Image

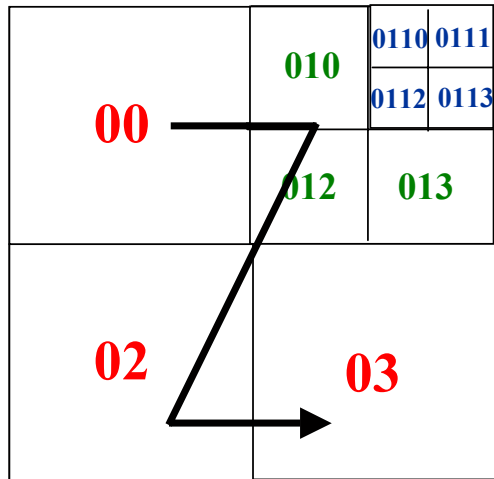


Arbre quaternaire représentant l'image

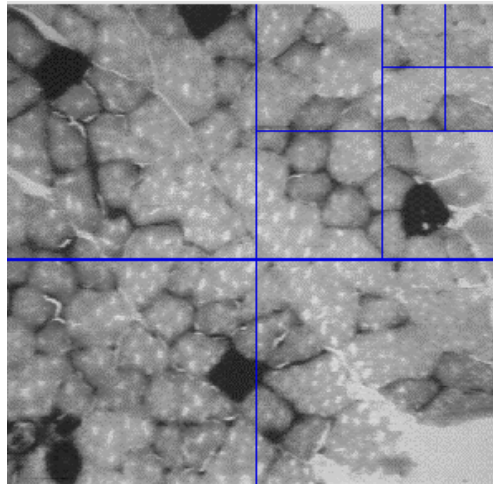


# Arbre Quaternaire

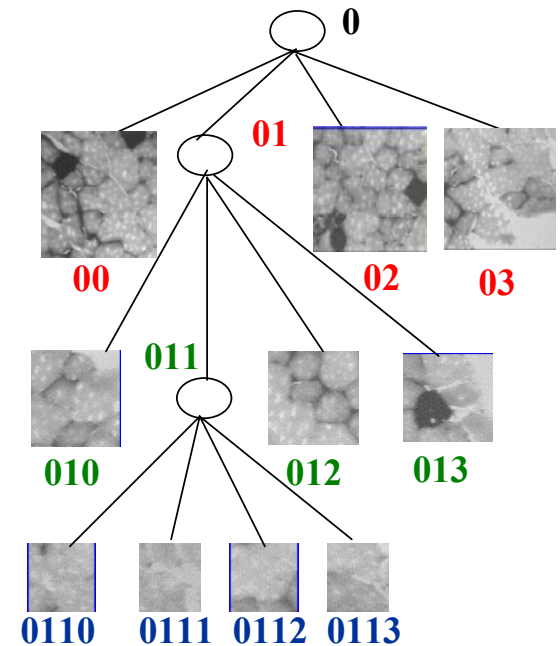
Identification des quadrants



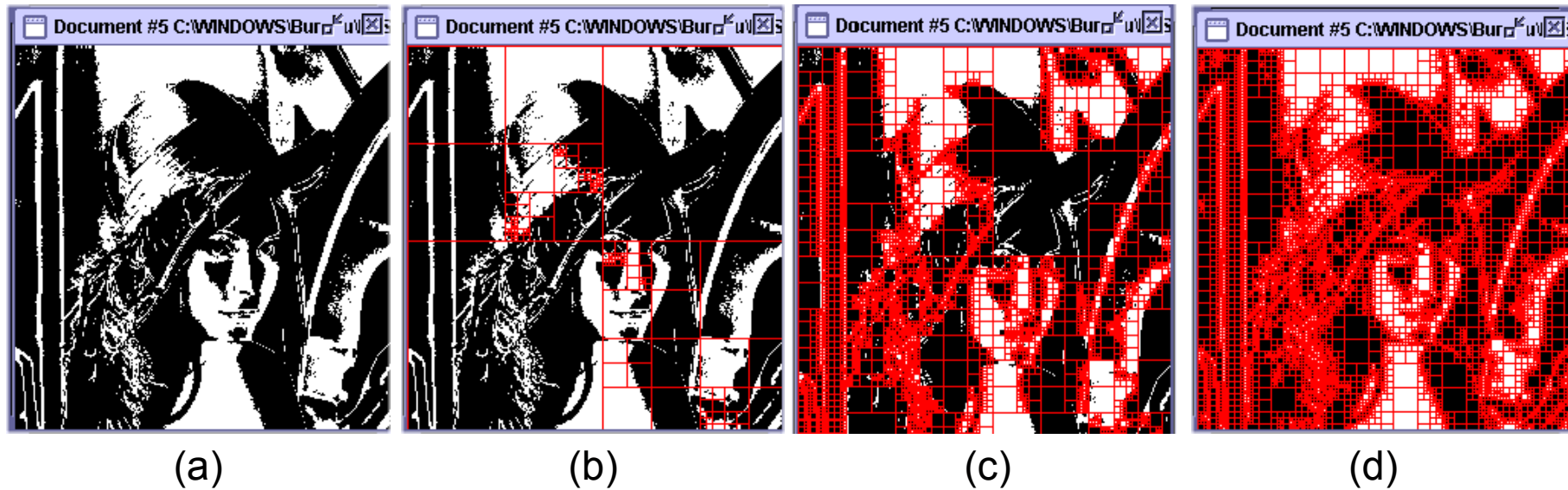
Image



Arbre quaternaire représentant l'image



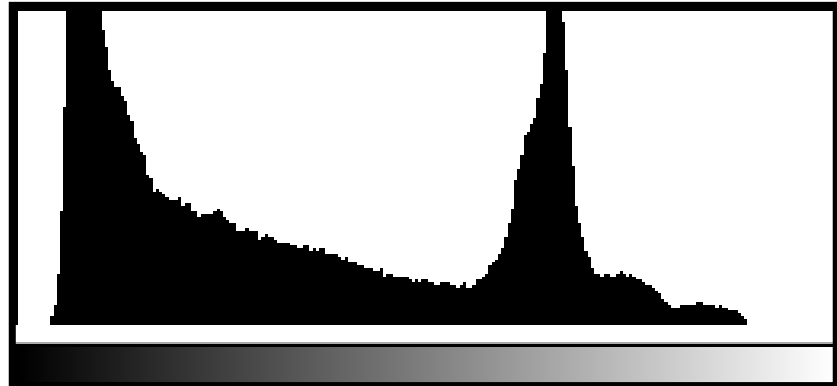
# Décomposition d'une image en arbre quaternaire en fonction de différents niveaux de résolution



(a) Image originale (b) homogénéité des couleurs de 70%, (c) 90% et (d) 100%

# Représentation multi-niveau

*Histogrammes en niveaux de gris*

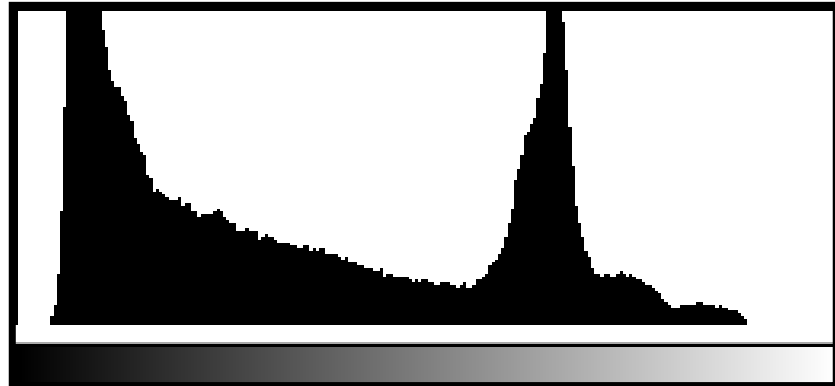


*Noir*

*Blanc*

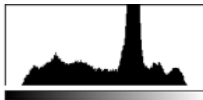
# Représentation multi-niveau

*Histogrammes en niveaux de gris*



*Noir*

*Blanc*



0



00



01



000



001



002

003

...

...

...



02

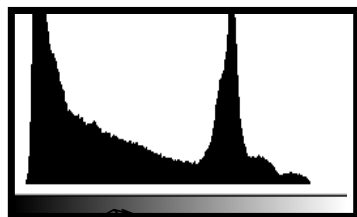


03



Utilisation d'un  
arbre quaternaire  
(*quadtree*)

0



00



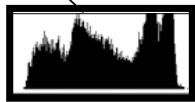
01



02



03



000

001

002

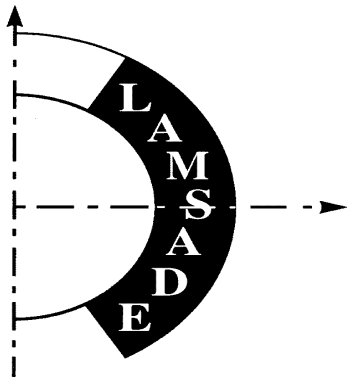
003

...

...

...

# A generalized distance between hierarchically partitioned images\*



Maude Manouvrier<sup>1</sup>

Marta Rukoz<sup>2</sup>

Geneviève Jomier<sup>1</sup>



*\* Supported in part by CDCH in Venezuela*

- 1. LAMSADE - Paris-Dauphine University - France*
- 2. CCPD - Universidad Central de Venezuela - Caracas*

# Outline

- **Background**
  - **Content-Based Image Retrieval**
  - **Recursive image partition**
  - **Multi-Level Feature Vector**
- **$\Delta$ -distance**
  - **Multi-level filtering for global image retrieval**
  - **Pattern and sub-image searches**
  - **Particular cases**
- **Conclusion and future work**

# Background (1/4)

## Content-Based Image Retrieval

- **Image visual features extracted and represented as vectors**
- **Representation of each image by a point in a multidimensional space**
- **Image similarity defined as distance between points**
- **Use of index structure to speed up image searches**

# Background (1/4)

## Content-Based Image Retrieval

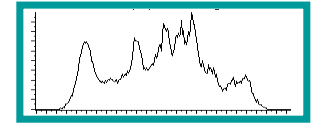


- **Image visual features extracted and represented as vectors**
- **Representation of each image by a point in a multidimensional space**
- **Image similarity defined as distance between points**
- **Use of index structure to speed up image searches**

# Background (1/4)

## Content-Based Image Retrieval

- **Image visual features extracted and represented as vectors**
- **Representation of each image by a point in a multidimensional space**
- **Image similarity defined as distance between points**
- **Use of index structure to speed up image searches**

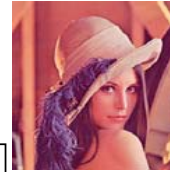


Color histogram

# Background (1/4)

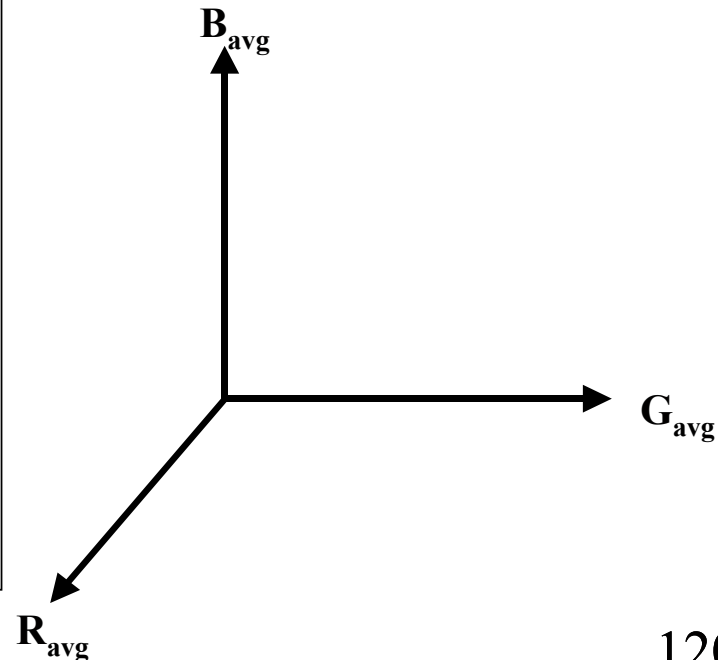
## Content-Based Image Retrieval

- Image visual features extracted and represented as vectors
- Representation of each image by a point in a multidimensional space
- Image similarity defined as distance between points
- Use of index structure to speed up image searches



(180, 98, 105)

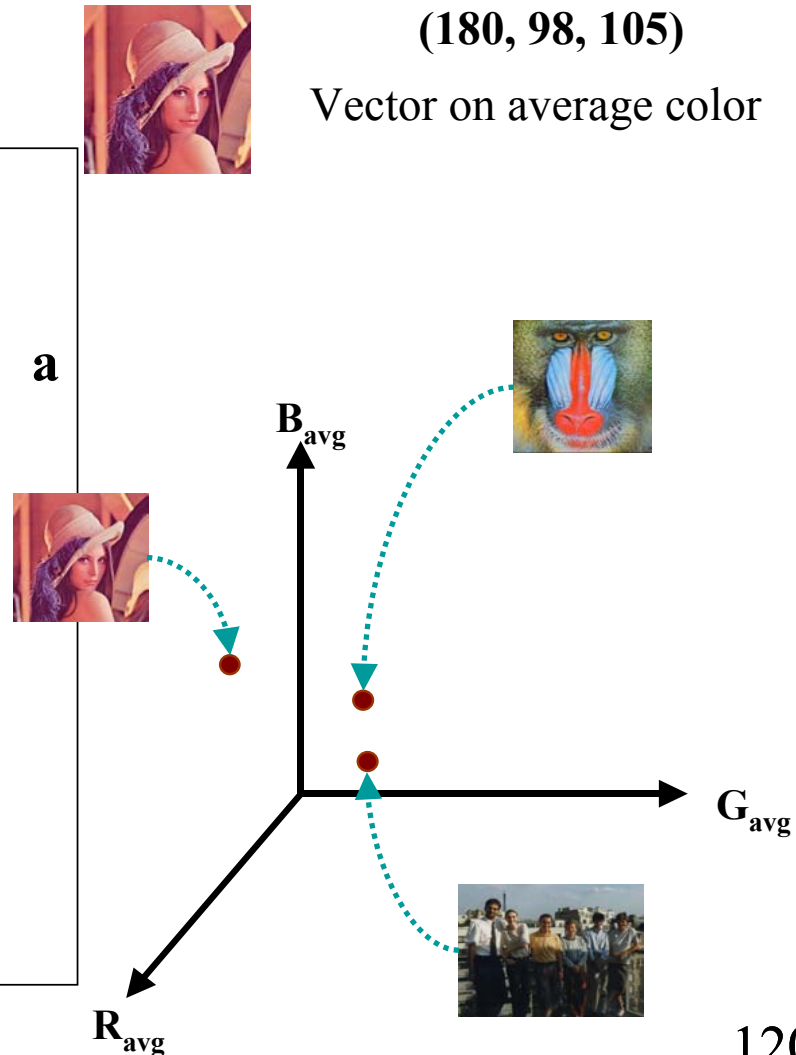
Vector on average color



# Background (1/4)

## Content-Based Image Retrieval

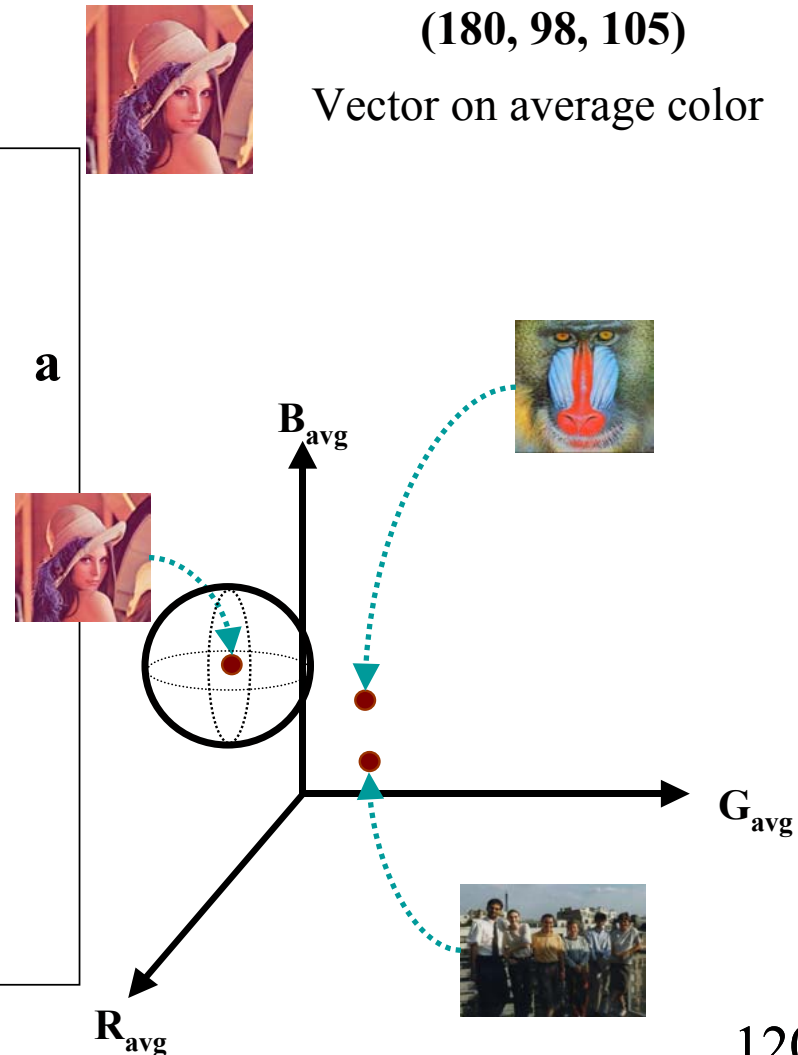
- Image visual features extracted and represented as vectors
- Representation of each image by a point in a multidimensional space
- Image similarity defined as distance between points
- Use of index structure to speed up image searches



# Background (1/4)

## Content-Based Image Retrieval

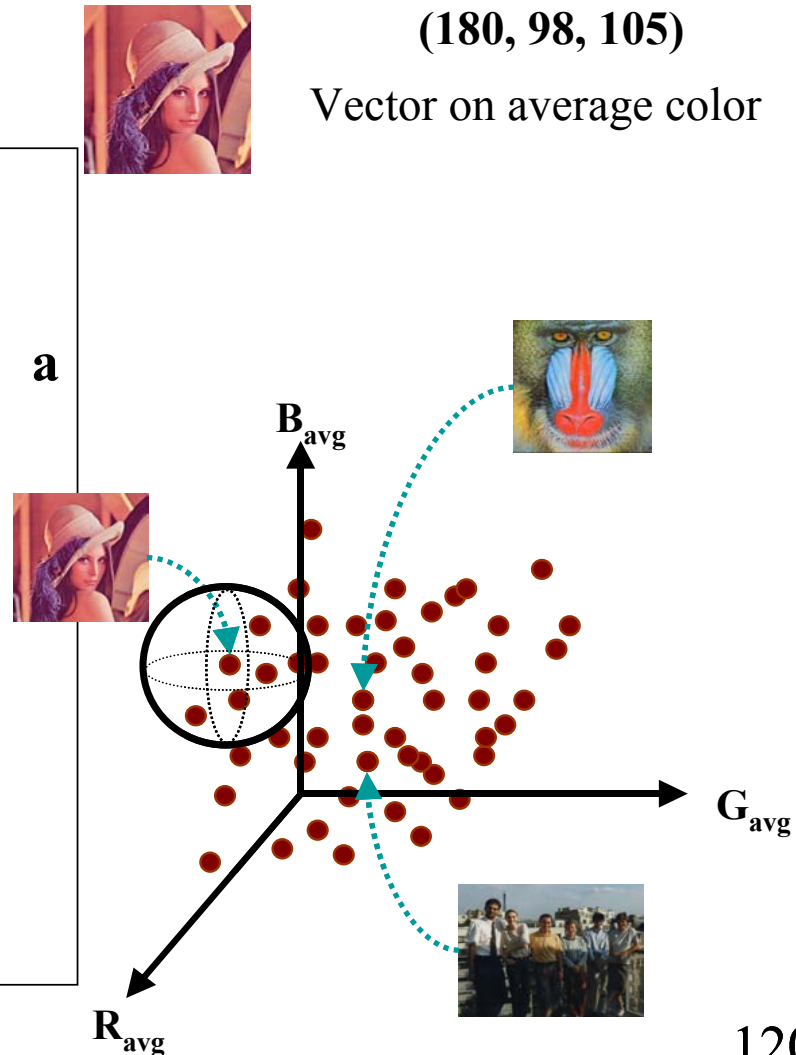
- Image visual features extracted and represented as vectors
- Representation of each image by a point in a multidimensional space
- Image similarity defined as distance between points
- Use of index structure to speed up image searches



# Background (1/4)

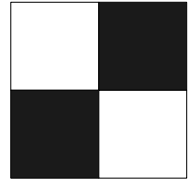
## Content-Based Image Retrieval

- Image visual features extracted and represented as vectors
- Representation of each image by a point in a multidimensional space
- Image similarity defined as distance between points
- Use of index structure to speed up image searches



# Background (2/4)

- Global similarity  $\nRightarrow$  Local similarity



# Background (2/4)

- Global similarity  $\nRightarrow$  Local similarity



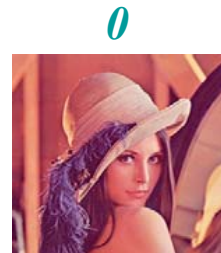
# Background (2/4)

- Global similarity  $\nrightarrow$  Local similarity
- Recursive image partition
  - o Quadtree decomposition



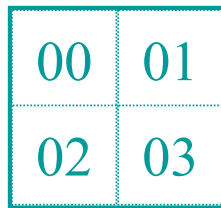
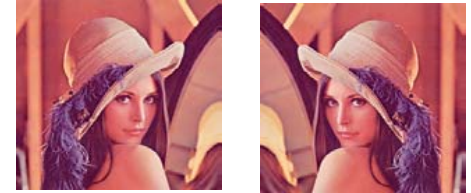
# Background (2/4)

- Global similarity  $\nrightarrow$  Local similarity
- Recursive image partition
  - o Quadtree decomposition



# Background (2/4)

- Global similarity  $\nRightarrow$  Local similarity
- Recursive image partition
  - o Quadtree decomposition

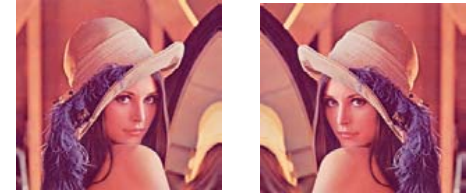


Level 1



# Background (2/4)

- Global similarity  $\nrightarrow$  Local similarity
- Recursive image partition
  - o Quadtree decomposition

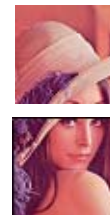
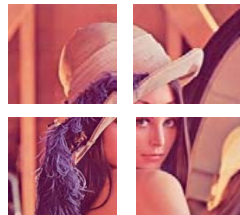
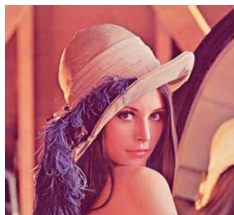


# Background (2/4)

- Global similarity  $\nrightarrow$  Local similarity
- Recursive image partition
  - o Nona-tree decomposition



Level 1



# Background (3/4)

## Multi-Level Feature Vector

- Representation of each quadrant by a visual descriptor
- Storage of the descriptors in tree nodes

# Background (3/4)

## Multi-Level Feature Vector

- Representation of each quadrant by a visual descriptor
- Storage of the descriptors in tree nodes

*0*



(180, 98, 105)

# Background (3/4)

## Multi-Level Feature Vector

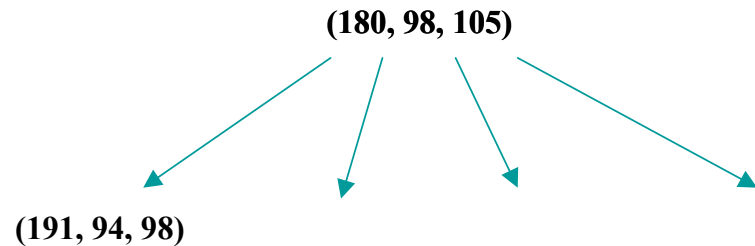
- Representation of each quadrant by a visual descriptor
- Storage of the descriptors in tree nodes



# Background (3/4)

## Multi-Level Feature Vector

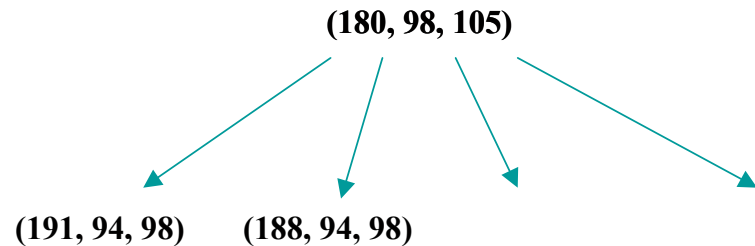
- Representation of each quadrant by a visual descriptor
- Storage of the descriptors in tree nodes



# Background (3/4)

## Multi-Level Feature Vector

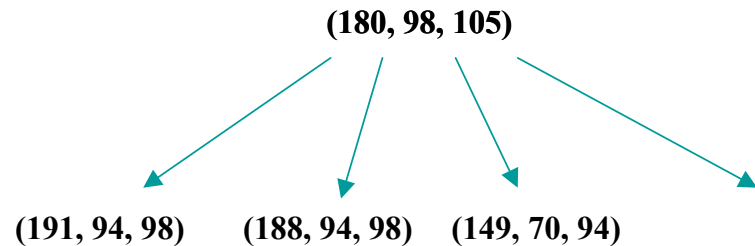
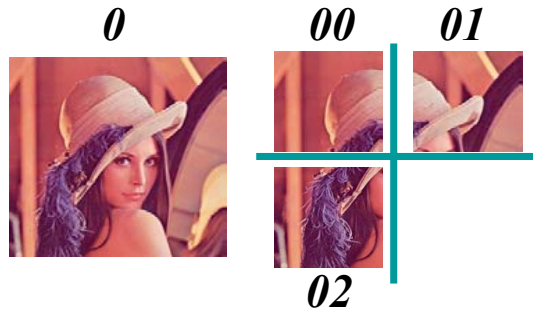
- Representation of each quadrant by a visual descriptor
- Storage of the descriptors in tree nodes



# Background (3/4)

## Multi-Level Feature Vector

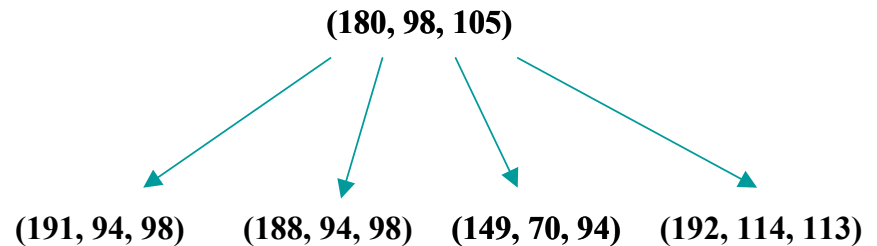
- Representation of each quadrant by a visual descriptor
- Storage of the descriptors in tree nodes



# Background (3/4)

## Multi-Level Feature Vector

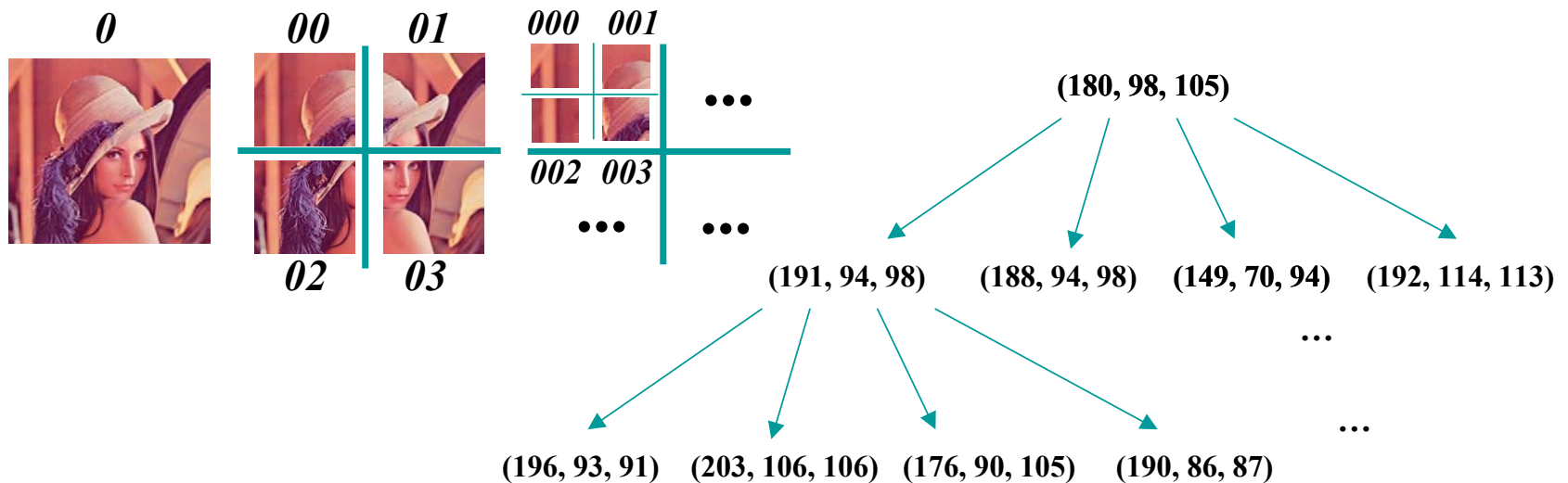
- Representation of each quadrant by a visual descriptor
- Storage of the descriptors in tree nodes



# Background (3/4)

## Multi-Level Feature Vector

- Representation of each quadrant by a visual descriptor
- Storage of the descriptors in tree nodes



# Background (4/4)

## Related works based on multi-level feature vector

| Reference                  | Tree                                  | Feature vector                                                           |
|----------------------------|---------------------------------------|--------------------------------------------------------------------------|
| (Jomier et al., 2005)      | quadtree                              | color moments                                                            |
| (Kim and Kim, 2000)        | quadtree                              | shape feature                                                            |
| (Lin et al., 2001)         | quadtree                              | average color vector in the root and color histograms in the other nodes |
| (Lu et al., 1994)          | quadtree                              | color histograms                                                         |
| (Luo and Nascimento, 2003) | mixed between a nona and a quad trees | mean and covariance color                                                |
| (Malki et al., 1999)       | quadtree                              | color and texture histograms                                             |
| (Remias et al., 1997)      | nona-tree                             | texture vectors                                                          |

# $\Delta$ -distance

**A generalized distance between multi-level feature vectors**

$$\Delta(i, j) = \frac{\sum_n w_n \delta(i, j, n)}{\sum_n w_n}$$

# $\Delta$ -distance

**A generalized distance between multi-level feature vectors**

$$\Delta(i, j) = \frac{\sum_n w_n \delta(i, j, n)}{\sum_n w_n}$$

- For all nodes  $n$  of both multi-level feature vectors

# $\Delta$ -distance

A generalized distance between multi-level feature vectors

$$\Delta(i, j) = \frac{\sum_n w_n \delta(i, j, n)}{\sum_n w_n}$$

- For all nodes  $n$  of both multi-level feature vectors
- $\delta(i, j, n)$  = metric distance between feature vectors stored in homologous node  $n$  of multi-level feature vectors of images  $i$  and  $j$

# $\Delta$ -distance

A generalized distance between multi-level feature vectors

$$\Delta(i, j) = \frac{\sum_n w_n \delta(i, j, n)}{\sum_n w_n}$$

- For all nodes  $n$  of both multi-level feature vectors
- $\delta(i, j, n)$  = metric distance between feature vectors stored in homologous node  $n$  of multi-level feature vectors of images  $i$  and  $j$
- $\delta$ -distances weighted by  $w_n$ ,  $w_n \geq 0$

# $\Delta$ -distance

A generalized distance between multi-level feature vectors

$$\Delta(i, j) = \frac{\sum_n w_n \delta(i, j, n)}{\sum_n w_n}$$

- For all nodes  $n$  of both multi-level feature vectors
- $\delta(i, j, n)$  = metric distance between feature vectors stored in homologous node  $n$  of multi-level feature vectors of images  $i$  and  $j$
- $\delta$ -distances weighted by  $w_n$ ,  $w_n \geq 0$
- $\Delta$ -distance normalized by  $W = \sum w_n$

# Global search using $\Delta$ -distance

Rank 1



Query result using a global distance



Query image

*From the  
prototype of  
(Jomier et al.,  
2005)*

Rank 4



Query result using  $\Delta$ -distance

**Search criteria : values and location of the image features**

# Multi-level filtering (1/3)

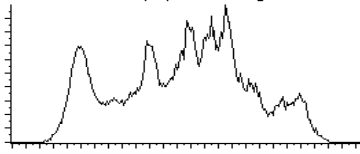
- $\Delta^{(l)}$  = approximation of  $\Delta$ -distance

$$\Delta^{(l)}(i,j) = \frac{1}{W} \sum_{k=0}^l (w_n \delta(i,j,n))$$

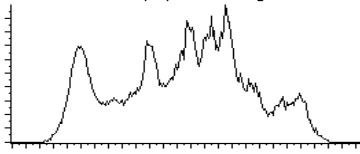
$$\Delta^{(l-1)}(i,j) \leq \Delta^{(l)}(i,j) \leq \Delta^{(l+1)}(i,j)$$

- Filtering process: Computing  $\Delta$ -distance level by level

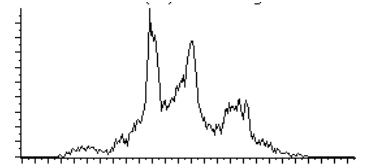




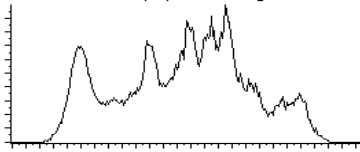
Feature vector of  
quadrant 0 of the  
query image  $q$



Feature vector of  
quadrant 0 of the  
query image  $q$

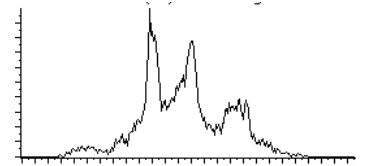


Feature vector of  
quadrant 0 of the  
image  $i$

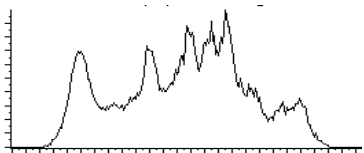


Feature vector of  
quadrant 0 of the  
query image  $q$

← If  $\Delta^{(0)}(i,j) \leq \alpha$  then proceed  
to the next level →

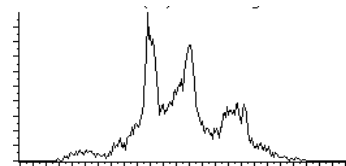


Feature vector of  
quadrant 0 of the  
image  $i$

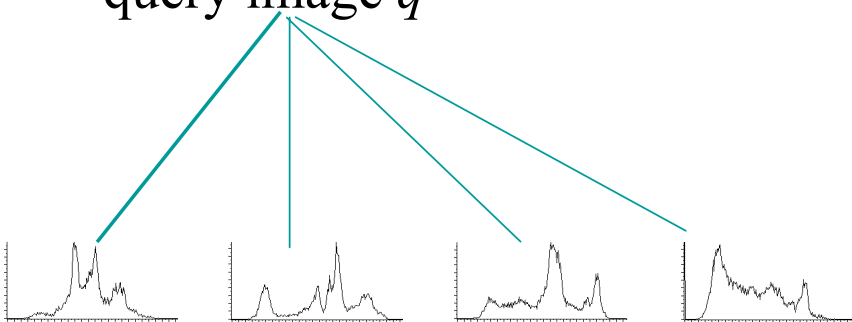


Feature vector of  
quadrant 0 of the  
query image  $q$

$\leftarrow$  If  $\Delta^{(0)}(i,j) \leq \alpha$  then proceed  
to the next level  $\rightarrow$



Feature vector of  
quadrant 0 of the  
image  $i$

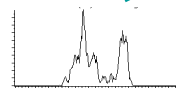


Quadrant  
00 of  
image  $q$

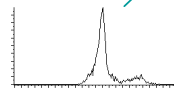
Quadrant  
01 of  
image  $q$

Quadrant  
02 of  
image  $q$

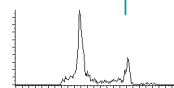
Quadrant  
03 of  
image  $q$



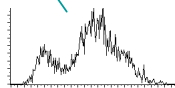
Quadrant  
00 of  
image  $i$



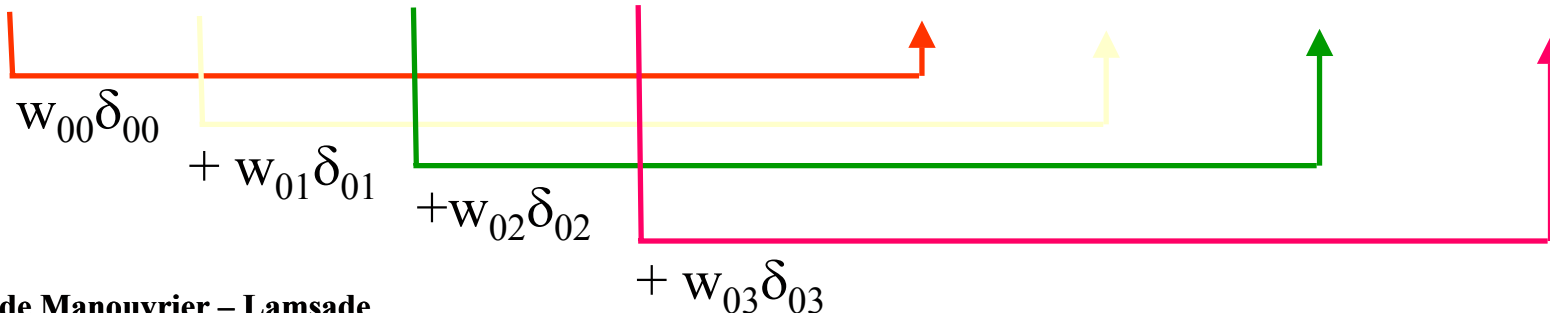
Quadrant  
01 of  
image  $i$

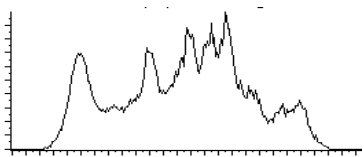


Quadrant  
02 of  
image  $i$

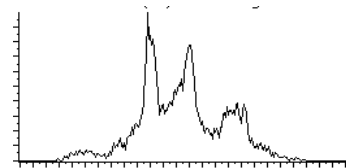


Quadrant  
03 of  
image  $i$





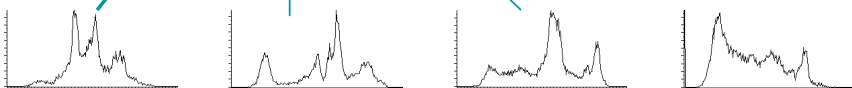
Feature vector of  
quadrant 0 of the  
query image  $q$



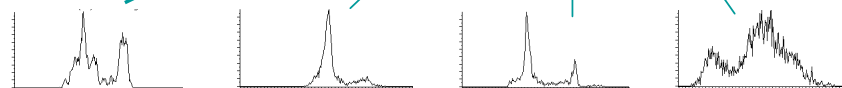
Feature vector of  
quadrant 0 of the  
image  $i$

If  $\Delta^{(0)}(i,j) \leq \alpha$  then proceed  
to the next level

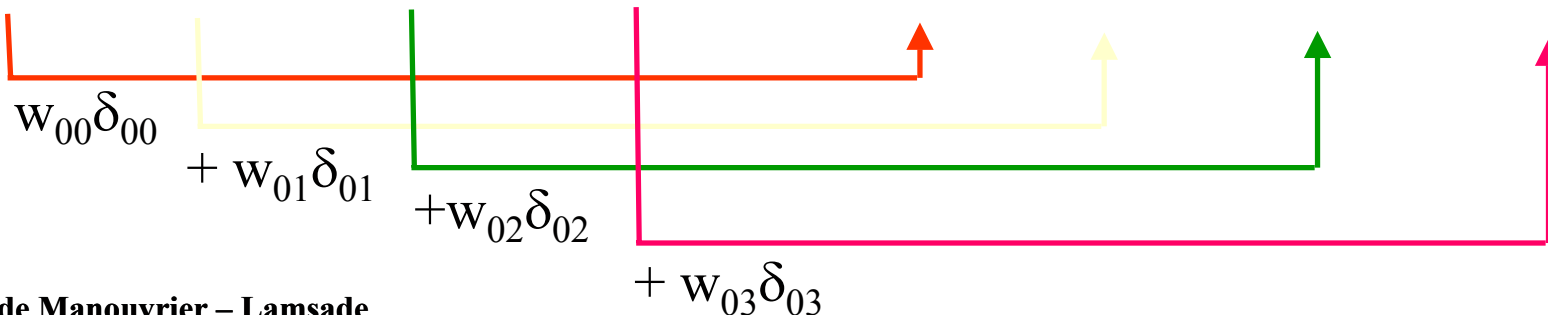
If  $\Delta^{(1)}(i,j) \leq \alpha$  then proceed to  
the next level



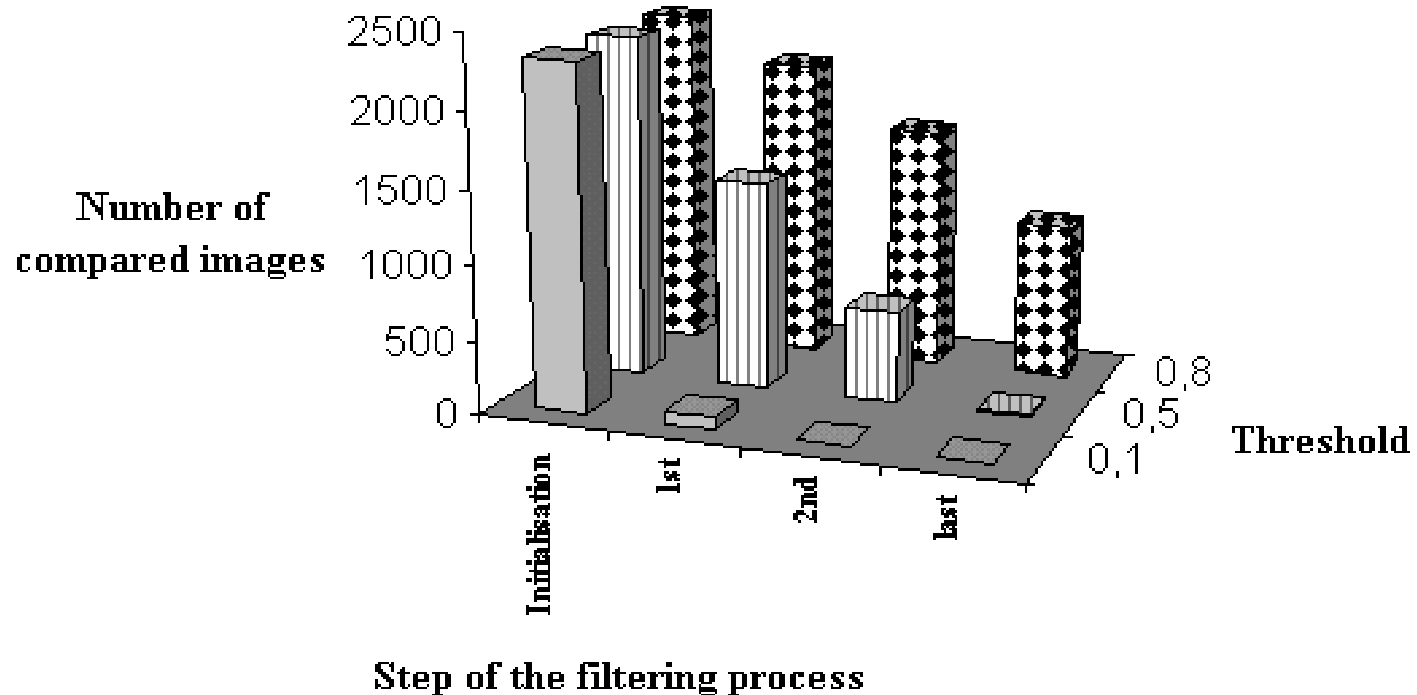
Quadrant 00 of image  $q$     Quadrant 01 of image  $q$     Quadrant 02 of image  $q$     Quadrant 03 of image  $q$



Quadrant 00 of image  $i$     Quadrant 01 of image  $i$     Quadrant 02 of image  $i$     Quadrant 03 of image  $i$



# Multi-level filtering (3/3)



# Pattern search (1/2)

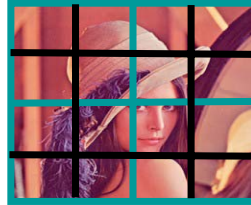
# Pattern search (1/2)

(a) Query image



# Pattern search (1/2)

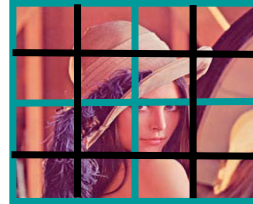
(a) Query image



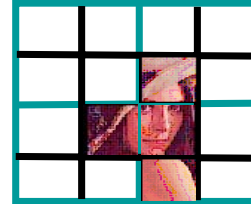
(b) 3 level quadtree image  
representation

# Pattern search (1/2)

(a) Query image



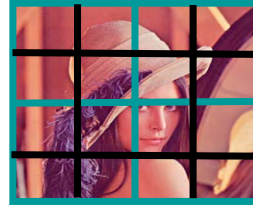
(b) 3 level quadtree image representation



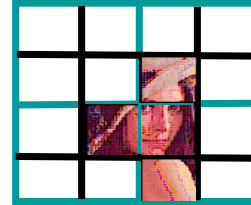
(c) Quadrants selected by the user

# Pattern search (1/2)

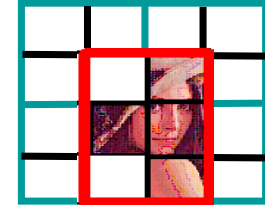
(a) Query image



(b) 3 level quadtree image representation



(c) Quadrants selected by the user

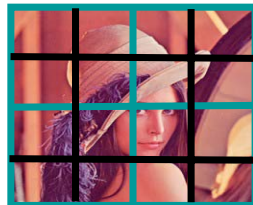


(d) Minimum bounding rectangle minimum containing the selected quadrants

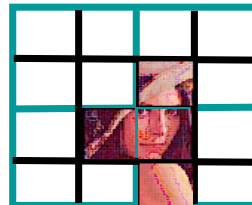
*Adapted from (Malki et al., 1999)*

# Pattern search (1/2)

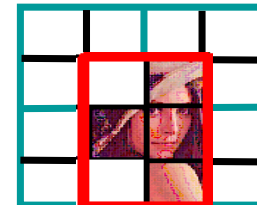
(a) Query image



(b) 3 level quadtree image representation



(c) Quadrants selected by the user



(d) Minimum bounding rectangle minimum containing the selected quadrants

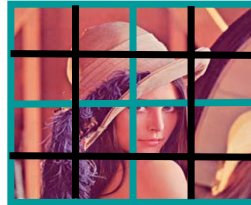
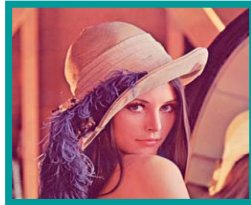
*Adapted from (Malki et al., 1999)*



(e) Translation of the minimum bounding rectangle

# Pattern search (1/2)

(a) Query image



(b) 3 level quadtree image representation

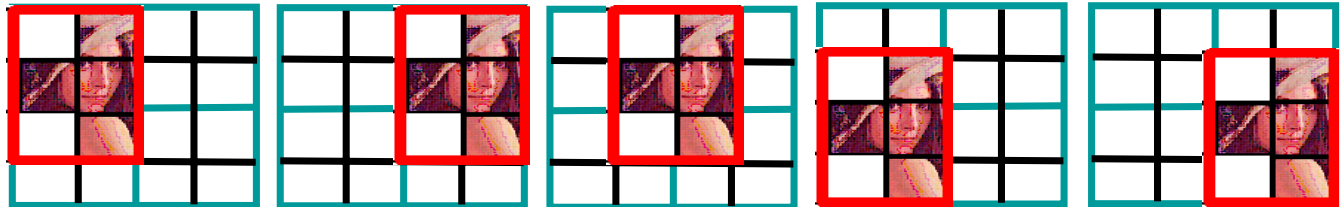


(c) Quadrants selected by the user



(d) Minimum bounding rectangle minimum containing the selected quadrants

*Adapted from (Malki et al., 1999)*



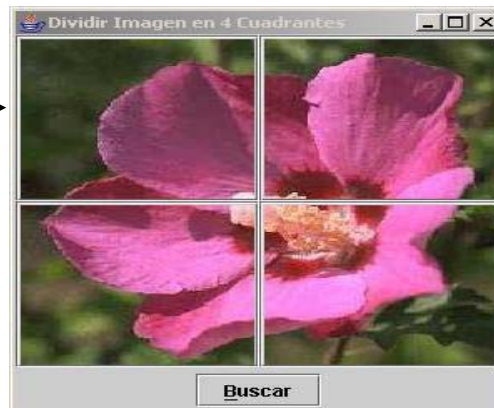
(e) Translation of the minimum bounding rectangle

Using  $\Delta_p(i, q)$  where  $w_n > 0$  for all selected image quadrants and  $w_n = 0$  for the other quadrants

# Pattern search (2/2)

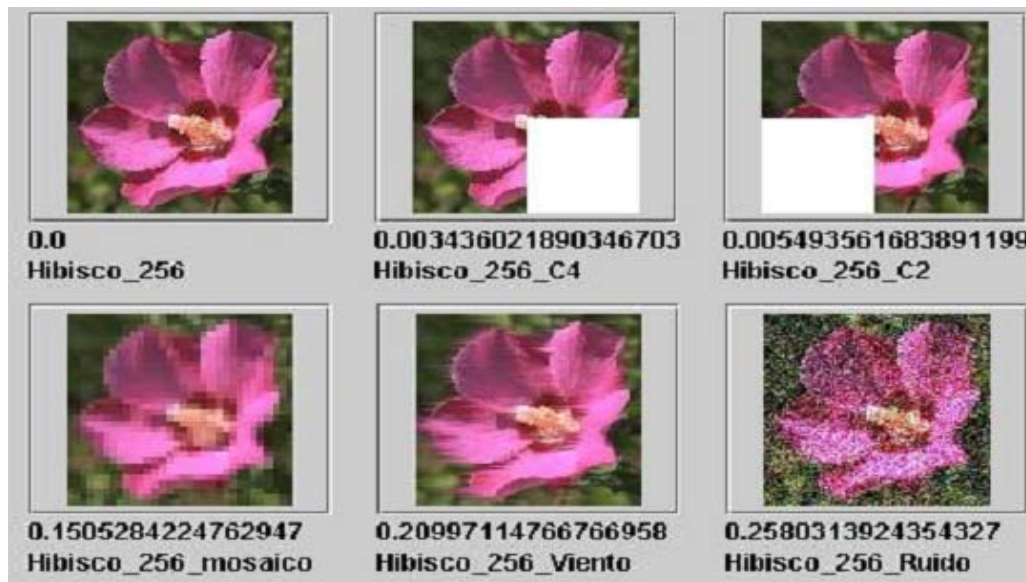
$$w_{00}=0.9$$

$$w_{01}=0.1$$



(a) Query image :  $w_{00}=0.9$  and  $w_{01}=0.1$  and other weights  $w_n=0$

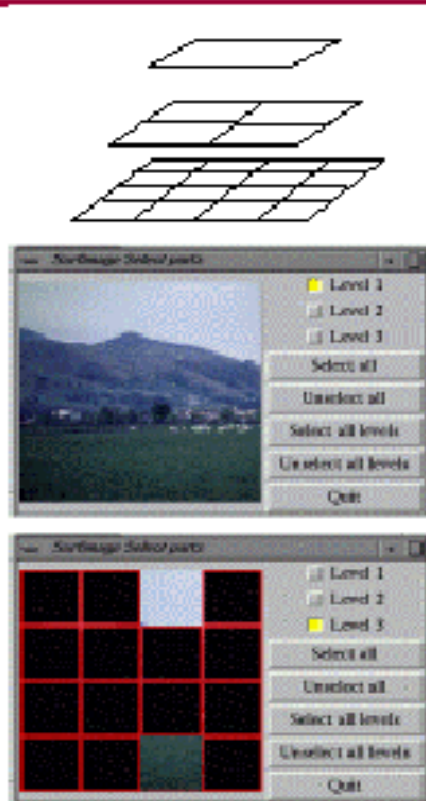
From the  
prototype of  
(Jomier et al.,  
2005)



(b) Query result

[MBNW99]

## *Segmentation systématique*



Requêtes partielles

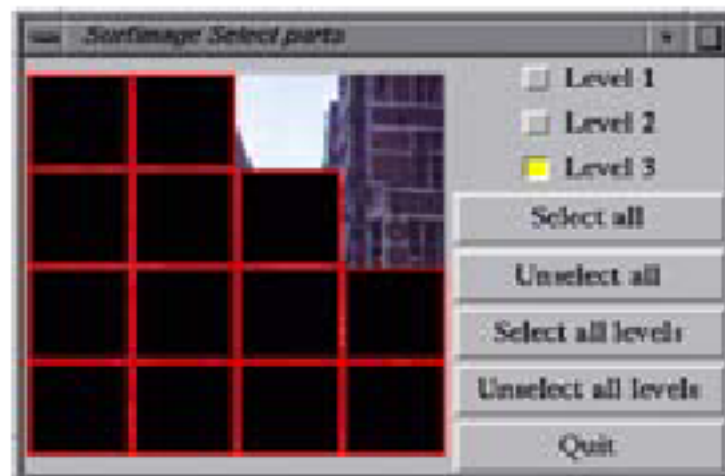
Résultat de la recherche sur la base générique

# Subdivision systématique

---



[MBNW99]



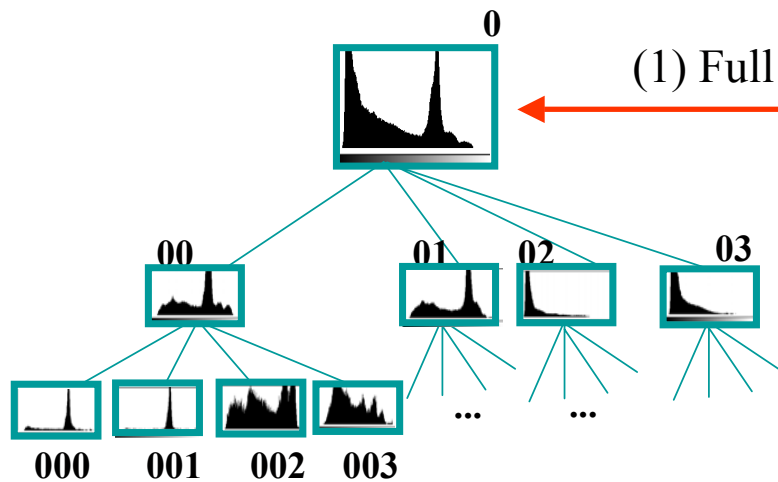
# **Sub-image search (1/2)**

**Comparing the query image with all image quadrants**

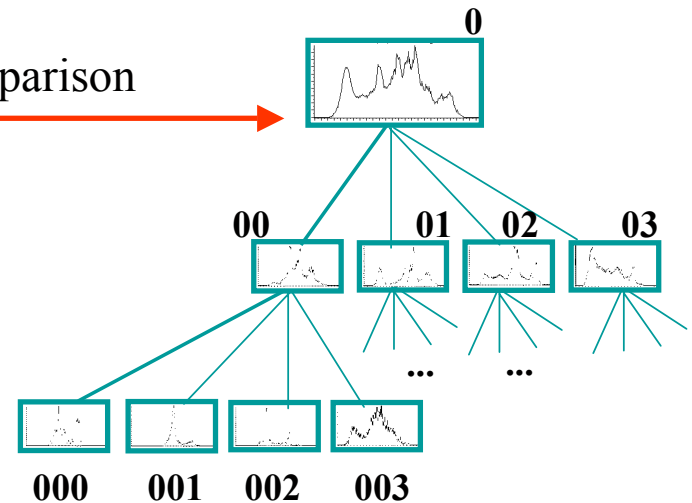
# Sub-image search (1/2)

Comparing the query image with all image quadrants

(a) Query Image



(b) Image  $i$  of the database



(1) Full tree comparison

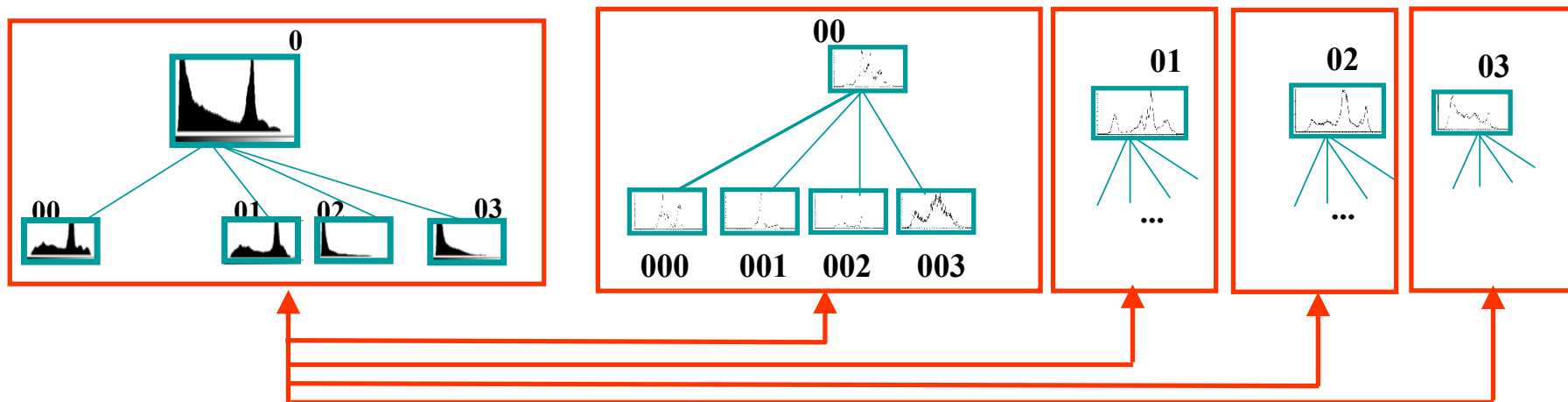


# Sub-image search (1/2)

Comparing the query image with all image quadrants

(a) Query Image

(b) Image  $i$  of the database



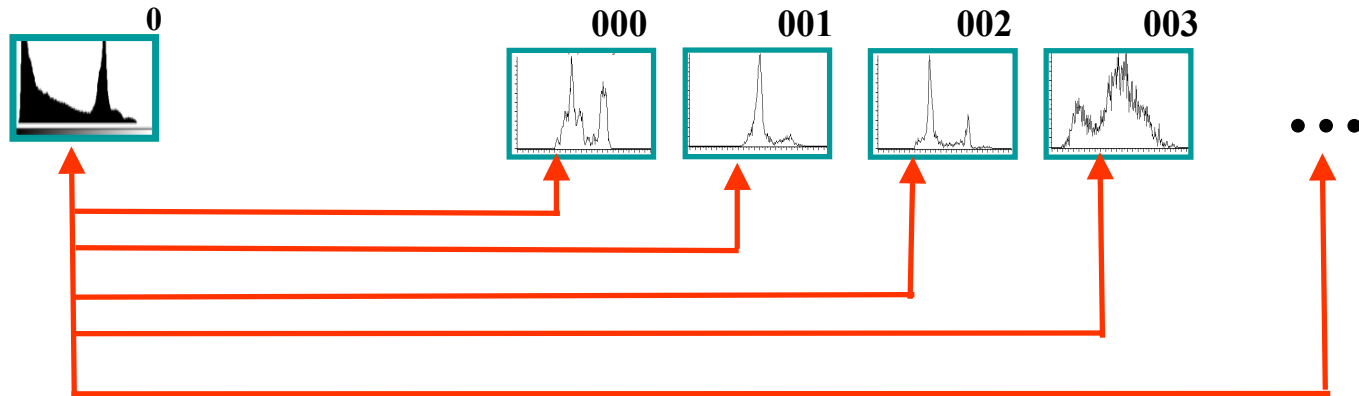
(2) Comparing the query image with the first level quadrants

# Sub-image search (1/2)

## Comparing the query image with all image quadrants

(a) Query Image

(b) Image  $i$  of the database

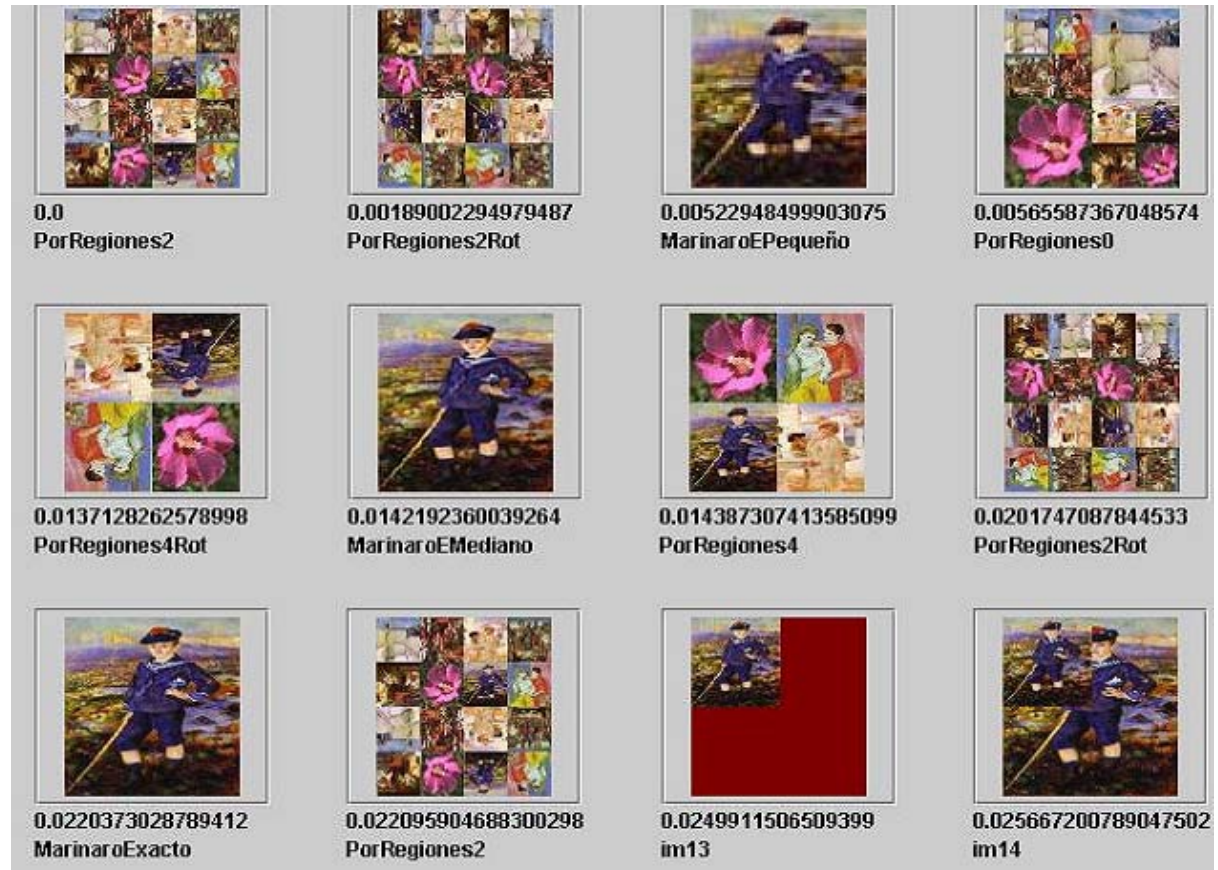


(3) Comparing the query image with the second level quadrants

# Sub-image search (2/2)



(a) Query image



(b) Query result

*From the prototype of (Jomier et al., 2005)*

# Particular cases of $\Delta$ -distance

| Reference                  | $\Delta$ -distance                           | $\delta$ | weights $w_n$                                                            |
|----------------------------|----------------------------------------------|----------|--------------------------------------------------------------------------|
| (Jomier et al., 2005)      | $\widetilde{\Delta}^{(\ell)}$ and $\Delta_p$ | $L_2$    | $4^{-\ell}$ for all nodes $n$ at level $\ell$                            |
| (Kim and Kim, 2000)        | $\Delta$                                     | $L_1$    | 1 for all nodes $n$                                                      |
| (Lin et al., 2001)         | $\widetilde{\Delta}^{(\ell)}$ and $\Delta_p$ | $L_2$    | $4^{-\ell}$ for all nodes $n$ at level $\ell$                            |
| (Lu et al., 1994)          | $\widetilde{\Delta}^{(\ell)}$                | $L_2$    | $4^{-\ell}$ for all nodes $n$ at level $\ell$                            |
| (Luo and Nascimento, 2003) | $\Delta$                                     | $L_1$    | $w_n = 0$ for all internal nodes $n$<br>and $w_n = 1$ for leaf nodes $n$ |
| (Malki et al., 1999)       | $\Delta_p$                                   | $d$      | $w_n = 0$ for all quadrants $n$<br>not selected by the user              |
| (Remias et al., 1997)      | $\Delta_p$                                   | $L_2$    | $w_n = 1$ for each<br>compared quadrants $n$<br>$w_n = 0$ otherwise      |

# Conclusion and future work

- **Done** :  $\Delta$ -distance presentation
  - Definition of a metric distance between images represented by multi-level feature vectors
  - Global image retrieval, computed by multi-level filtering
  - Pattern and sub-image searches
  - Generalization of existing distances, based on weights  $w_n$  and on  $\delta$ -distance between image quadrants
- **To do** : Develop a  $\Delta$ -distance based prototype
  - To compare existing works
  - To help the user to choose weights  $w_n$  and  $\delta$ -distances
  - Including full-balanced multi-level feature vectors and those based on image segmentation

# References

- G. Jomier, M. Manouvrier, V. Oria and M. Rukoz **Multilevel Index for Global and Partial Content-Based Image Retrieval**. *Proc. of the 1st IEEE Int. Workshop on Managing Data for Emerging Multimedia Applications (EMMA)*, Tokyo (Japan), pp. 66-75, April 8-9th 2005
- Hae-Kwang Kim and Jong-Deuk Kim. **Region-based shape descriptor invariant to rotation, scale and translation**. *Signal Processing: Image Communication*, 16(1-2):1-293, Sept. 2000
- S. Lin, M. Tamer Özsu, V. Oria, and R. Ng. **An Extendible Hash for Multi-Precision Similarity Querying of Image Databases**. In *Proc. of the 27th Int. Conf. on Very Large DataBases (VLDB'2001)*, Roma (Italy), pages 221-230, Sept. 2001
- H. Lu, B-C. Ooi, and K-L. Tan. **Efficient Image Retrieval by Color Contents**. In *First Int. Conf. on Applications of Databases (ADB-94)*, Vadstena (Sweden), June 1994. Lecture Notes in Computer Sciences - 819 - Springer Verlag
- J. Luo and M.A. Nascimento. **Content Based Sub-image Retrieval via Hierarchical Tree Matching**. In *Proc. of the First ACM Int. Workshop on Multimedia Databases (ACM MMDB 2003)*, New Orleans, Louisiana (USA), pages 63--69, Nov. 2003
- J. Malki, N. Boujemaa, C. Nastar, and A. Winter. **Region Queries without Segmentation for Image Retrieval by Content**. In *3rd Int. Conf. on Visual Information Systems (Visual'99) - Lecture Notes in Computer Science 1614*, Springer Verlag, Amsterdam (The Netherlands), pages 115-122, June 1999
- E. Remias, G. Sheikholeslami, A. Zhang, and T.F. Syeda-Mahmood. **Supporting Content-Based Retrieval in Large Image Database Systems**. *Multimedia Tools and Applications*, 4(2):153-170, March 1997

# Similarité d'images représentées par des arbres quaternaires

- **Histogramme de couleurs :**

- GUPTA A., JAIN R., "Visual Information Retrieval", *CACM*, 1997
- LU H., OOI B.-C., TAN K.-L., "Efficient Image Retrieval by Color Contents", *ADB-94*

- **Vecteurs moyenne des couleurs :**

LIN S., TAMER ÖZSU M., ORIA V., NG R., "An Extendible Hash for Multi-Precision Similarity Querying of Image Databases", *VLDB'2001*

- **Histogrammes combinant la couleur et la texture :**

MALKI J., BOUJEMAA N., NASTAR C., WINTER A., "Region Queries without Segmentation for Image Retrieval by Content", *Visual'99*

- **Descripteurs de forme :**

KIM H.-K., KIM J.-D., "Region-based shape descriptor invariant to rotation, scale and translation", *Signal Processing : Image Com.*, 2000

- **Définition de distances entre images représentées par des arbres quaternaires :**

RUKOZ M., MANOUVRIER M., JOMIER G., "Distances de similarité d'images basées sur les arbres quaternaires", *BDA'2002*

- **Structure d'index pour des images représentées par des arbres quaternaires :**

- MANOUVRIER M., RUKOZ M., JOMIER G., "A Generalized Metric Distance between Hierarchically Partitioned Images.", *MDM-KDD'2005*
- JOMIER G., MANOUVRIER M., V. ORIA, RUKOZ M., "Multi-level index for global and partial content-based image retrieval.", *IEEE EMMA'2005*

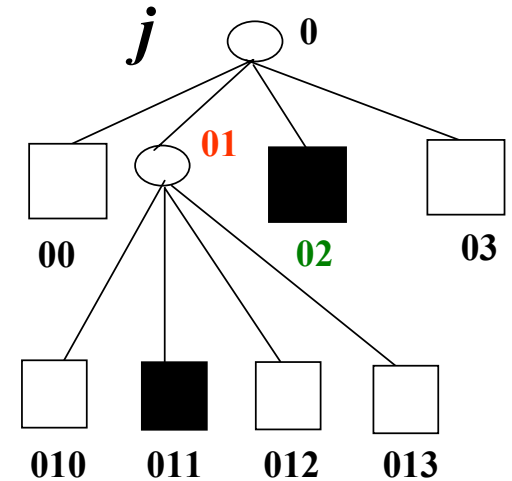
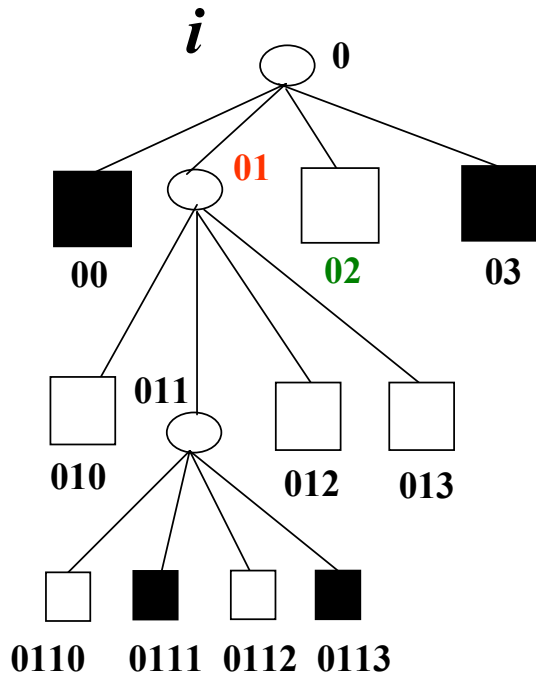
[JMO+04,  
MRJ05]

# $\Delta$ -distance

$$\Delta(i, j) = \frac{\sum_n w_n \delta(i, j, n)}{\sum_n w_n}$$

**En fonction des poids  $w_n$  plusieurs distances spécifiques peuvent être définies**

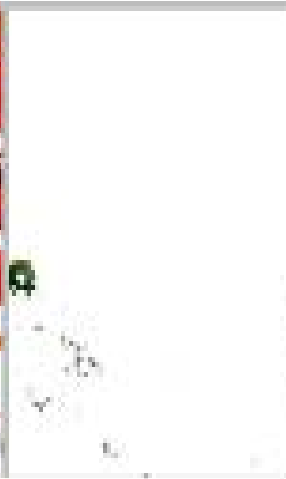
# Distance T : distance entre structures d'arbres



# Distance T : distance entre structures d'arbres



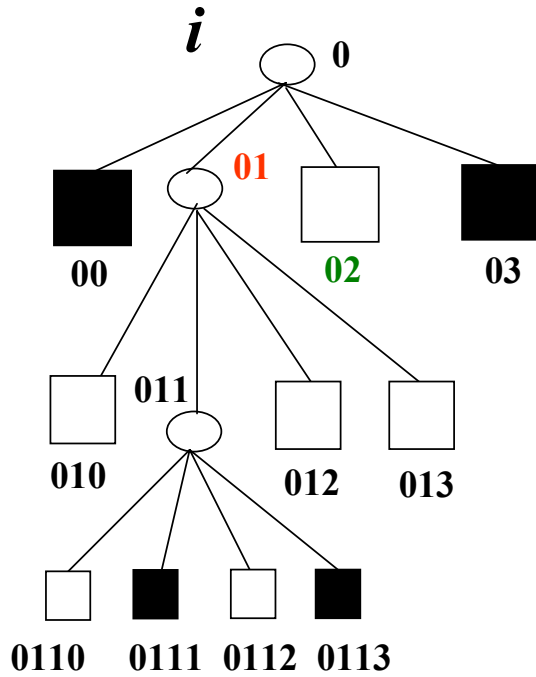
(a)



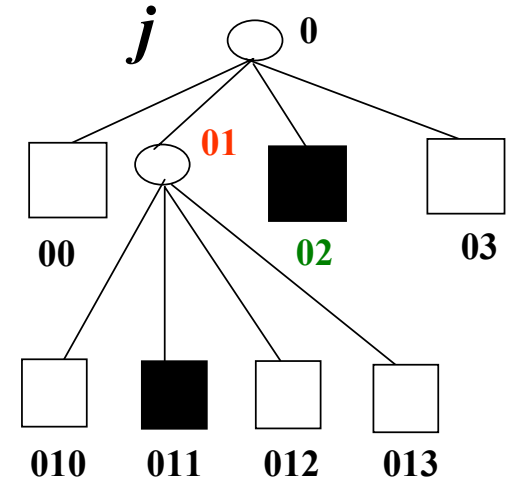
(b)

E. Albuz, E.D. Kocalar, and A.A. Khokhar. *Quantized CIELab\* Space and Encoded Spatial Structure for Scalable Indexing of Large Color Image Archives*. IEEE ICASSP, June 2000

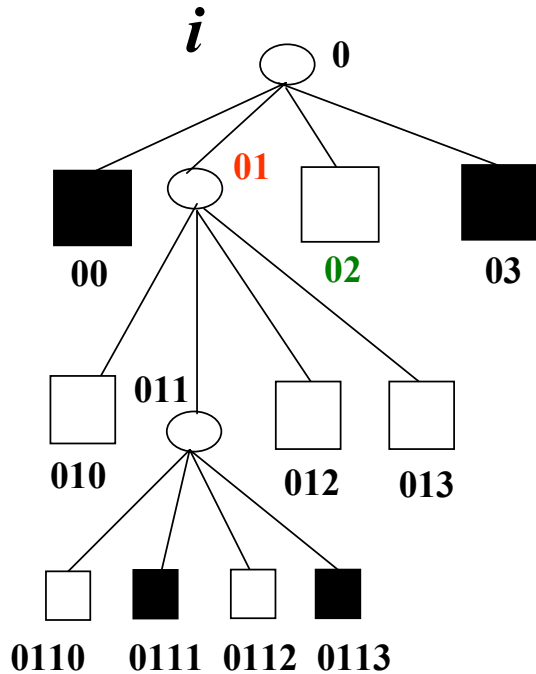
# Distance T : distance entre structures d'arbres



$$\delta_{01}(I, J) = 0$$

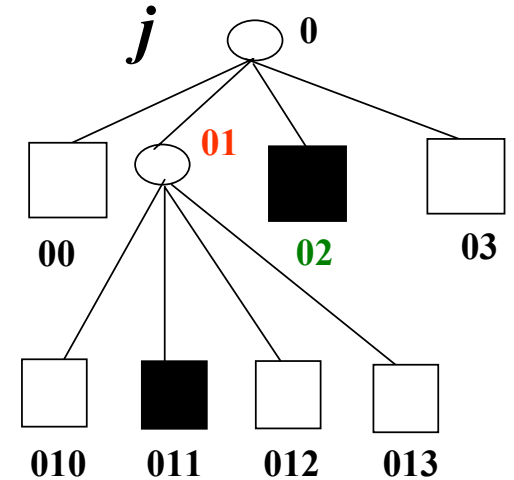


# Distance T : distance entre structures d'arbres

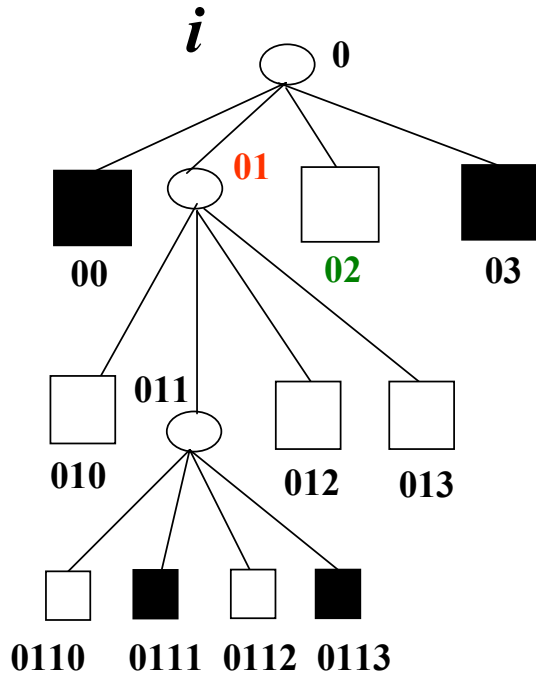


$$\delta_{01}(I, J) = 0$$

$$\delta_{02}(I, J) = 0$$



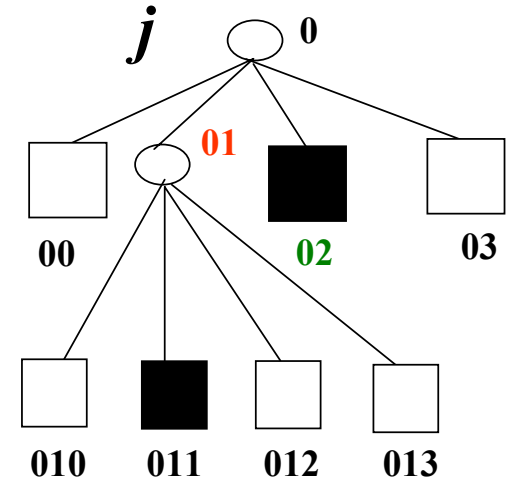
# Distance T : distance entre structures d'arbres



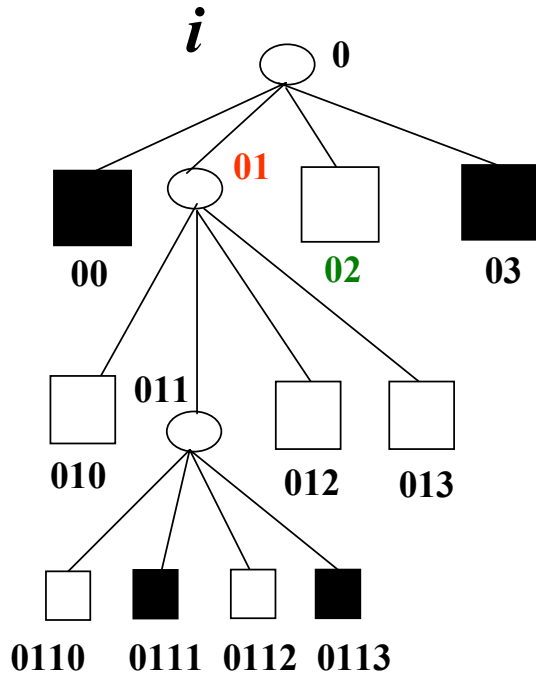
$$\delta_{01}(I, J) = 0$$

$$\delta_{02}(I, J) = 0$$

$$\delta_{011}(I, J) = 1$$



# Distance T : distance entre structures d'arbres

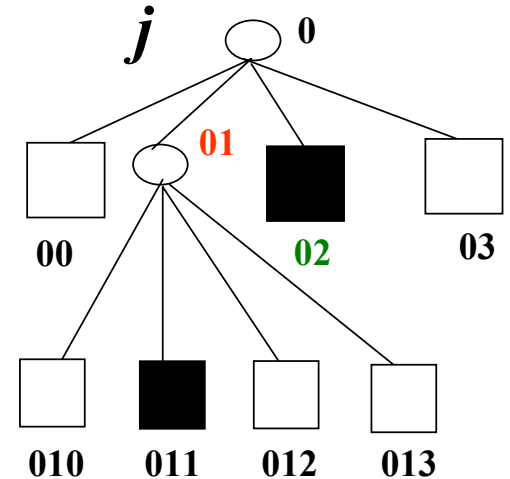


$$\delta_{01}(I, J) = 0$$

$$\delta_{02}(I, J) = 0$$

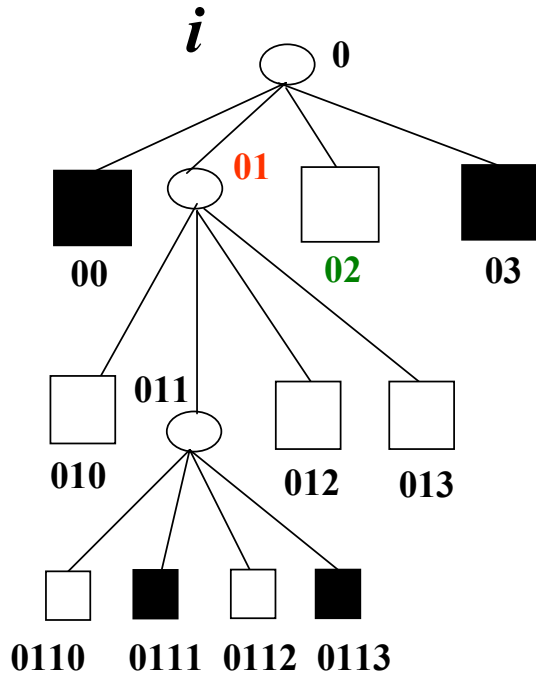
$$\delta_{011}(I, J) = 1$$

$$\delta_{011x}(I, J) = 1$$



Valeur particulière  $\perp$

# Distance T : distance entre structures d'arbres

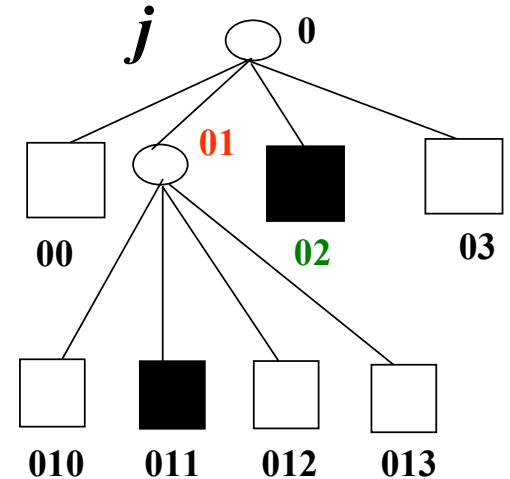


$$\delta_{01}(I, J) = 0$$

$$\delta_{02}(I, J) = 0$$

$$\delta_{011}(I, J) = 1$$

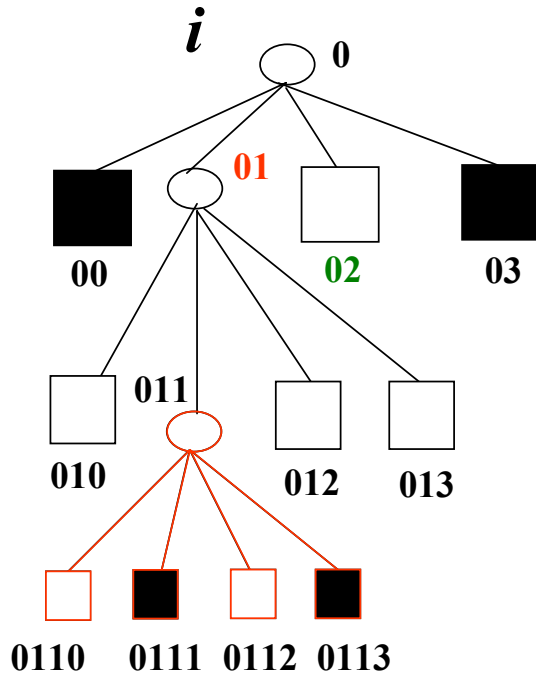
$$\delta_{011x}(I, J) = 1$$



Valeur particulière  $\perp$

*Distance entre les arbres quaternaires d'une même image découpée selon deux critères différents :*

# Distance T : distance entre structures d'arbres

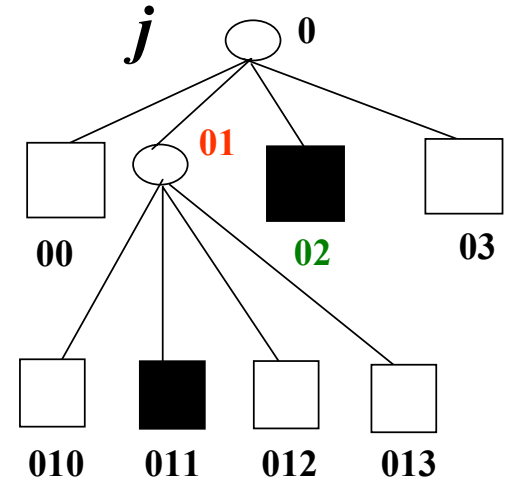


$$\delta_{01}(I, J) = 0$$

$$\delta_{02}(I, J) = 0$$

$$\delta_{011}(I, J) = 1$$

$$\delta_{011x}(I, J) = 1$$

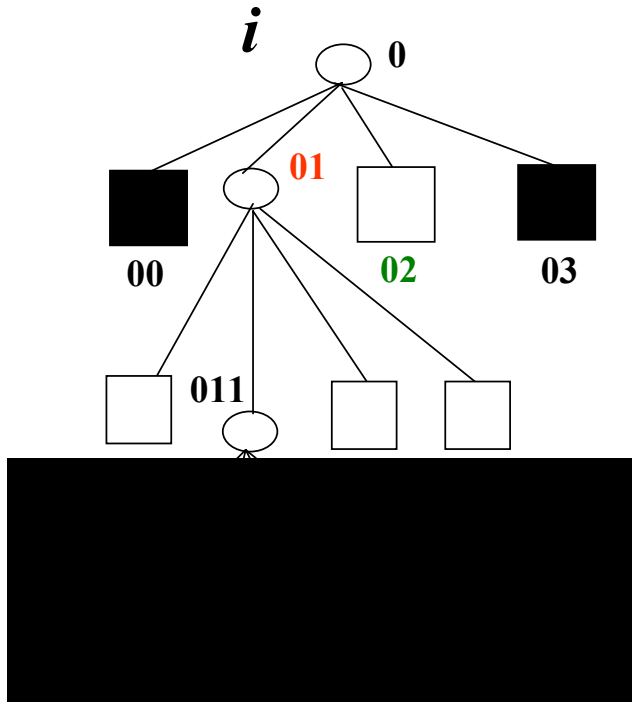


Valeur particulière  $\perp$

*Distance entre les arbres quaternaires d'une même image découpée selon deux critères différents :*

$$T(i, j) = 5 / 13$$

# Distance T : distance entre structures d'arbres

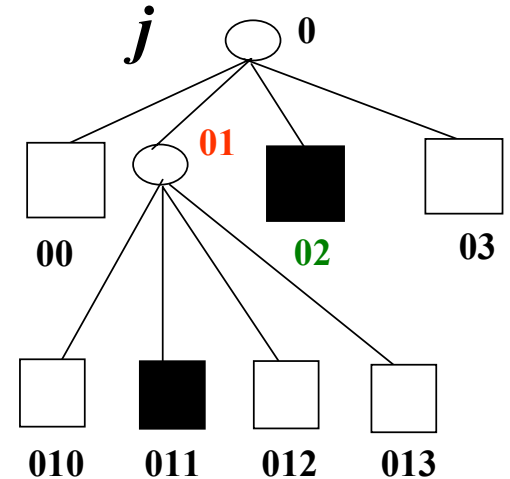


$$\delta_{01}(I, J) = 0$$

$$\delta_{02}(I, J) = 0$$

$$\delta_{011}(I, J) = 1$$

$$\delta_{011x}(I, J) = 1$$

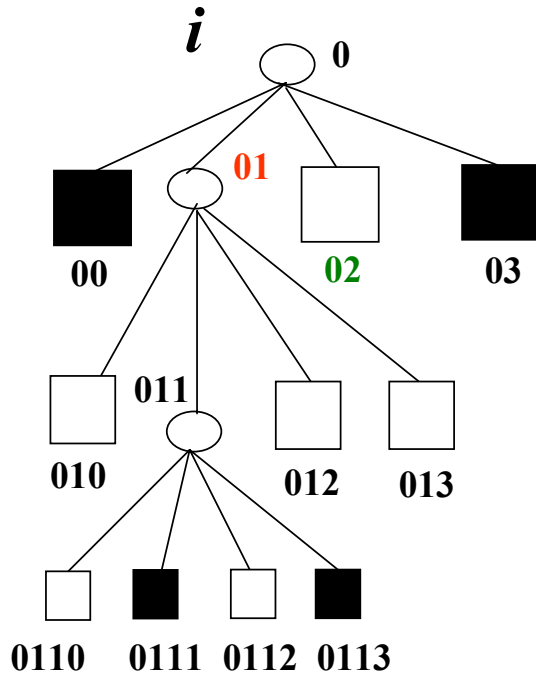


Valeur particulière  $\perp$

*Distance entre les arbres quaternaires d'une même image découpée selon deux critères différents :*

$$T^{(2)}(i, j) = 1/9$$

# Distance T : distance entre structures d'arbres

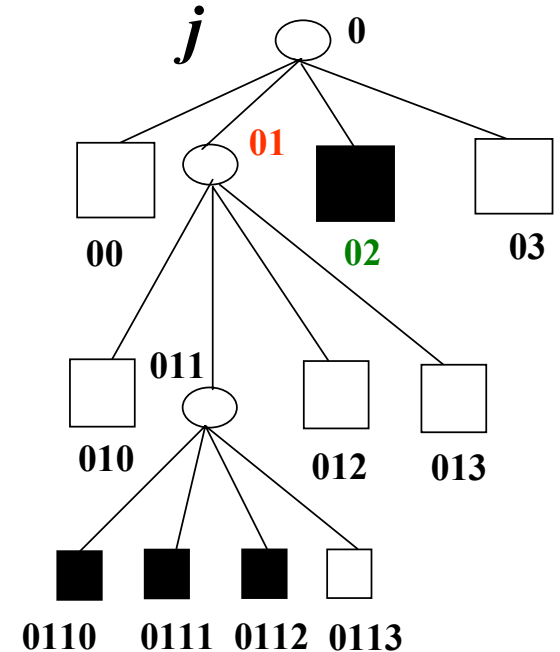


$$\delta_{01}(i, j) = 0$$

$$\delta_{02}(i, j) = 0$$

$$\delta_{011}(i, j) = 0$$

$$\delta_{011x}(i, j) = 0$$



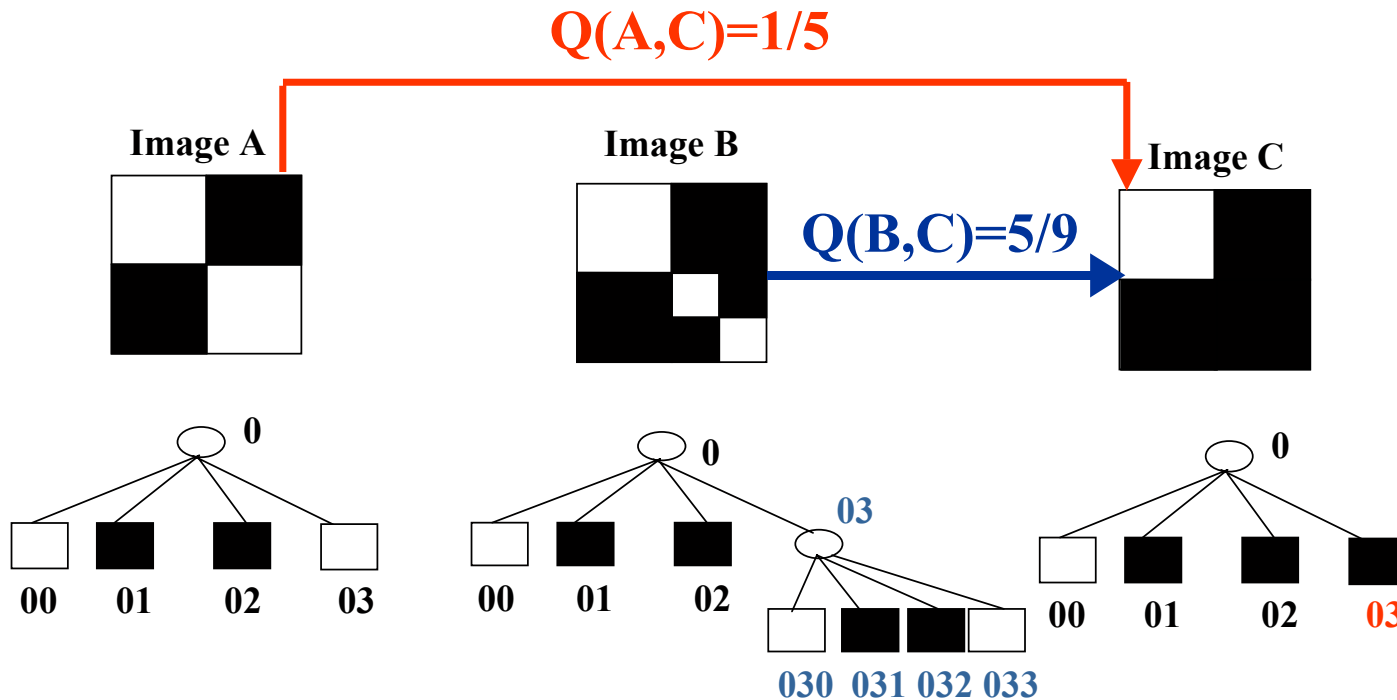
*Distance entre les arbres quaternaires de deux images différentes découpées selon le même critère :*

$$T(i, j) = 0$$

# Distance Q



Pour évaluer le partage entre arbres quaternaires

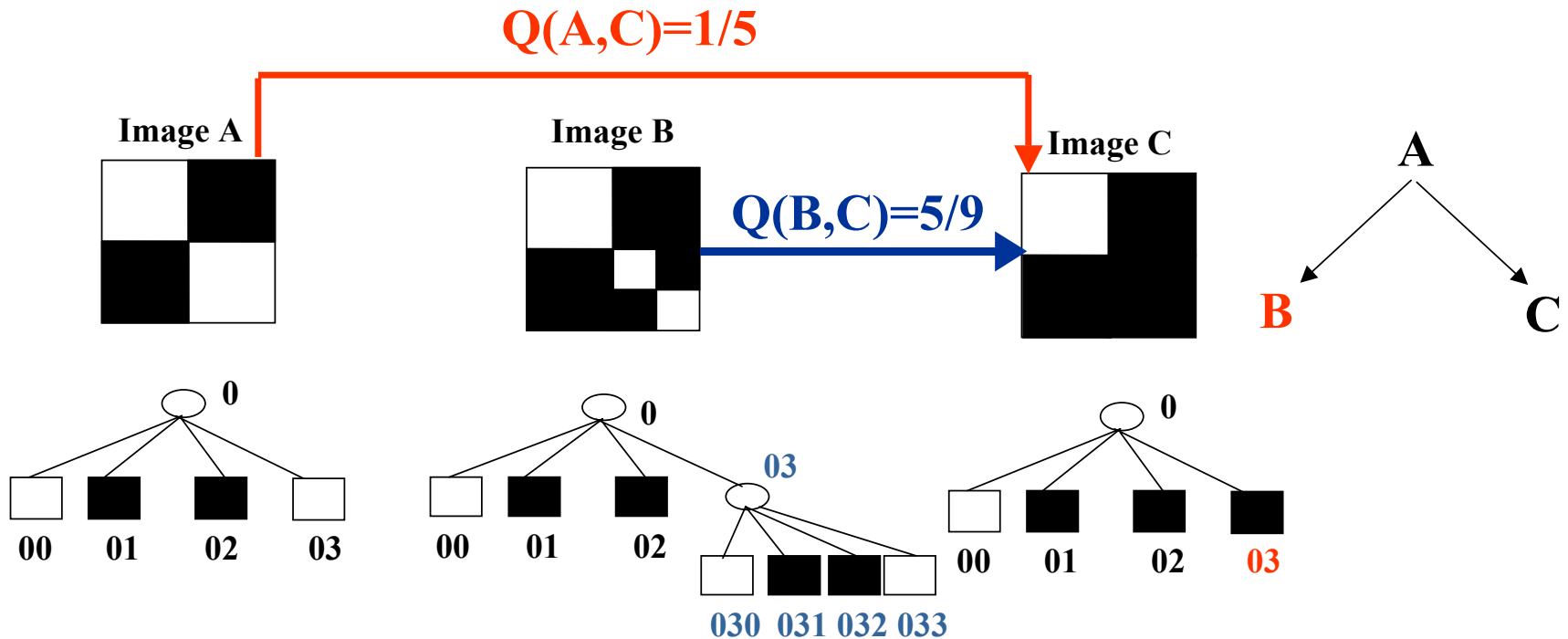


*M. Manouvrier, M. Rukoz, and G. Jomier. Quadtree representations for storage and manipulation of clusters of images. Image and Vision Computing, 2002*

# Distance Q



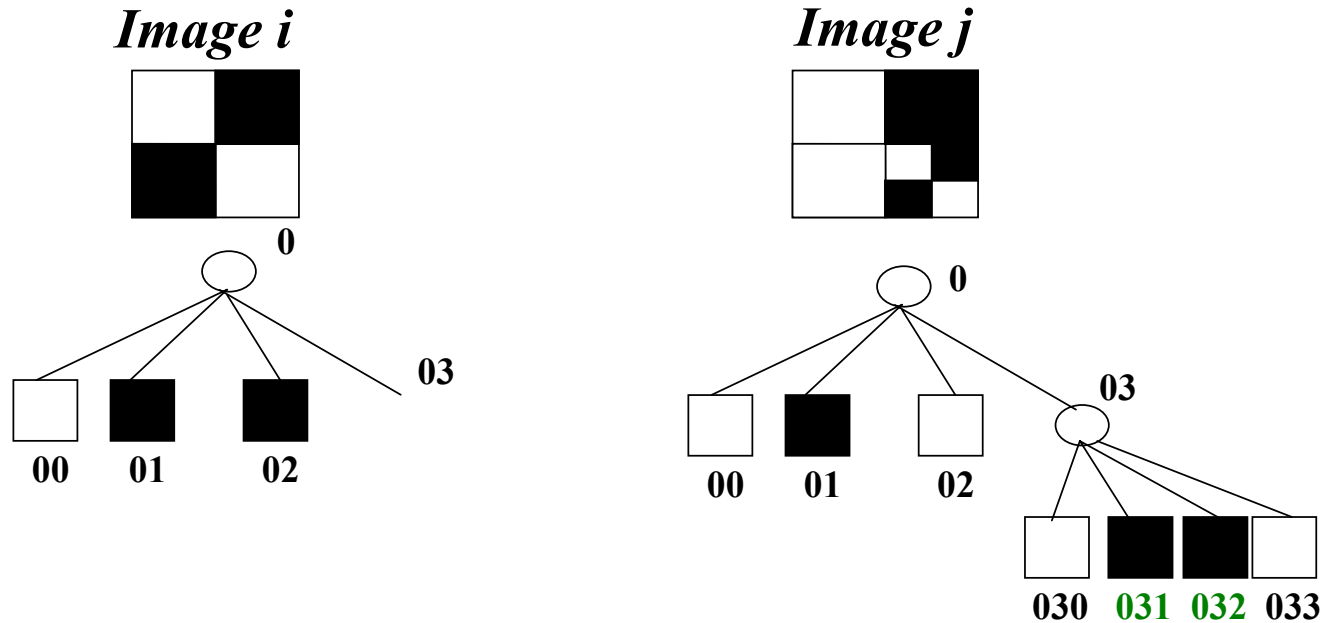
Pour évaluer le partage entre arbres quaternaires



*M. Manouvrier, M. Rukoz, and G. Jomier. Quadtree representations for storage and manipulation of clusters of images. Image and Vision Computing, 2002*

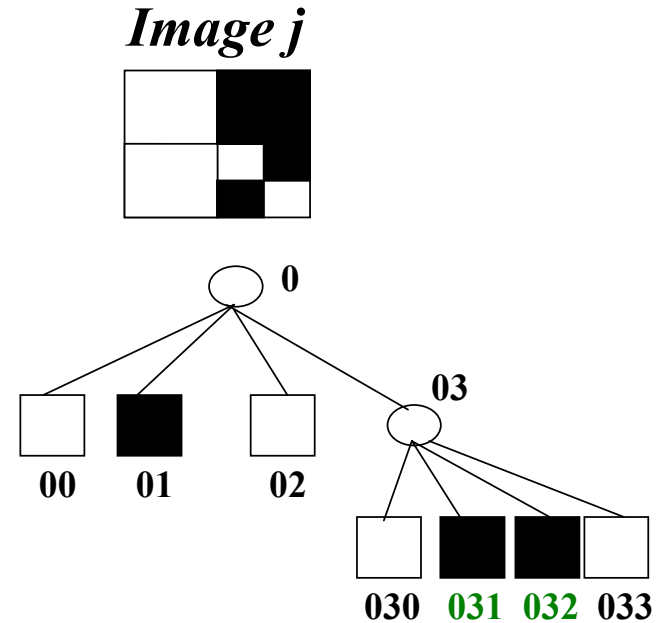
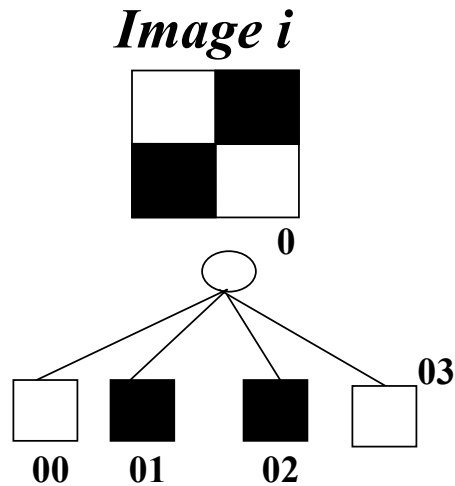
# Distance V

Distance visuelle entre images organisées en arbre quaternaire



# Distance V

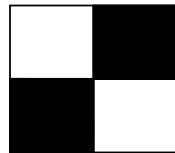
Distance visuelle entre images organisées en arbre quaternaire



# Distance V

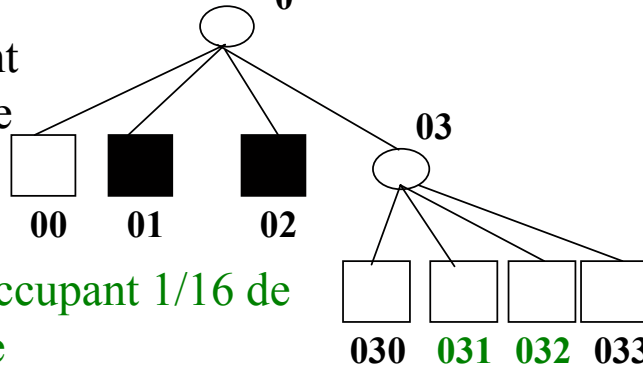
## Distance visuelle entre images organisées en arbre quaternaire

*Image i*



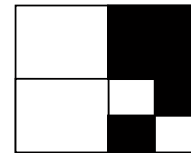
0

Nœuds occupant  
1/4 de la surface  
de l'image

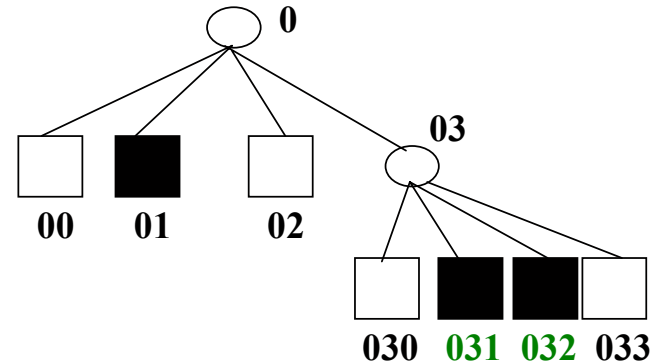


Nœuds occupant 1/16 de  
la surface

*Image j*



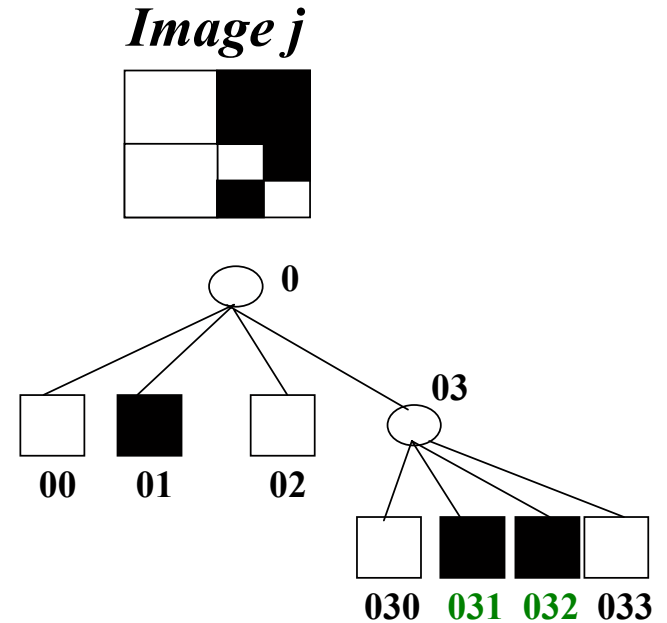
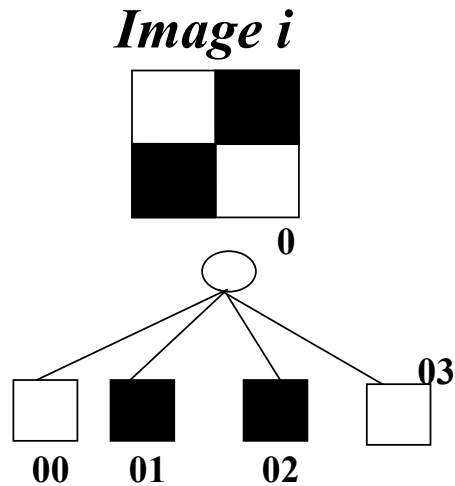
0



$$V(i,j) = \frac{1 * \textcolor{red}{1/4} + \textcolor{green}{2} * \textcolor{red}{1/16}}{3 * 1/4 + 4 * 1/16} = \frac{3}{8}$$

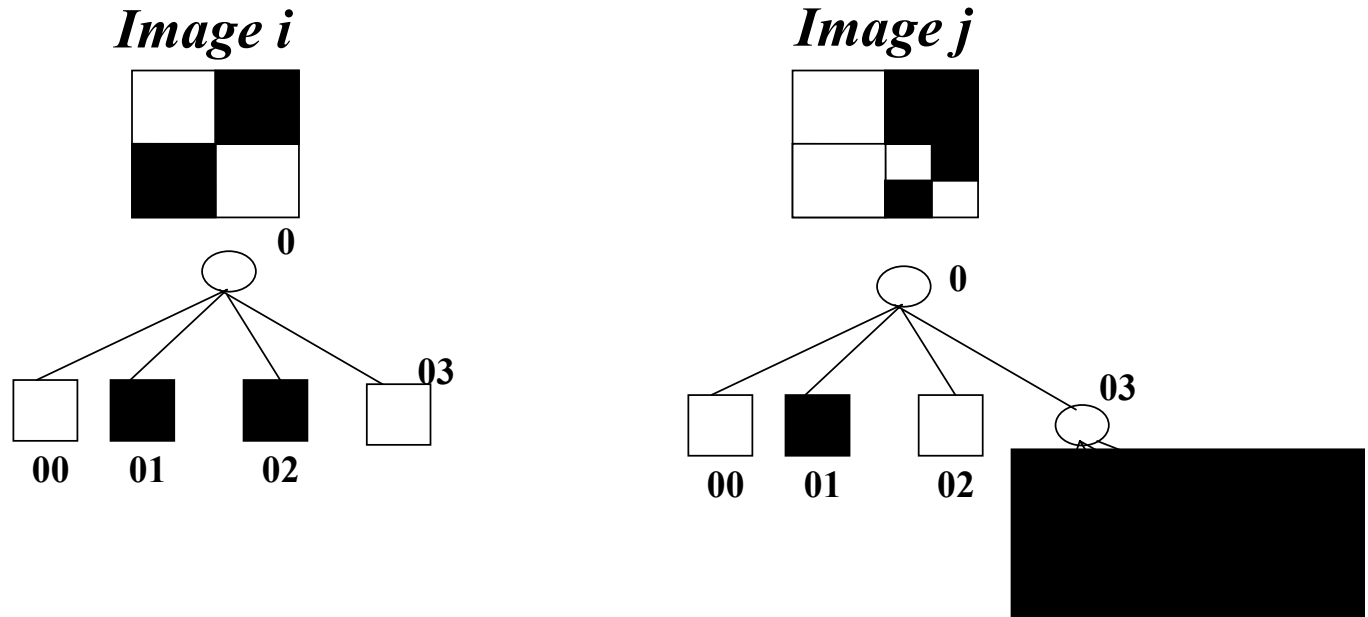
# Distance V

Distance visuelle entre images organisées en arbre quaternaire



# Distance V

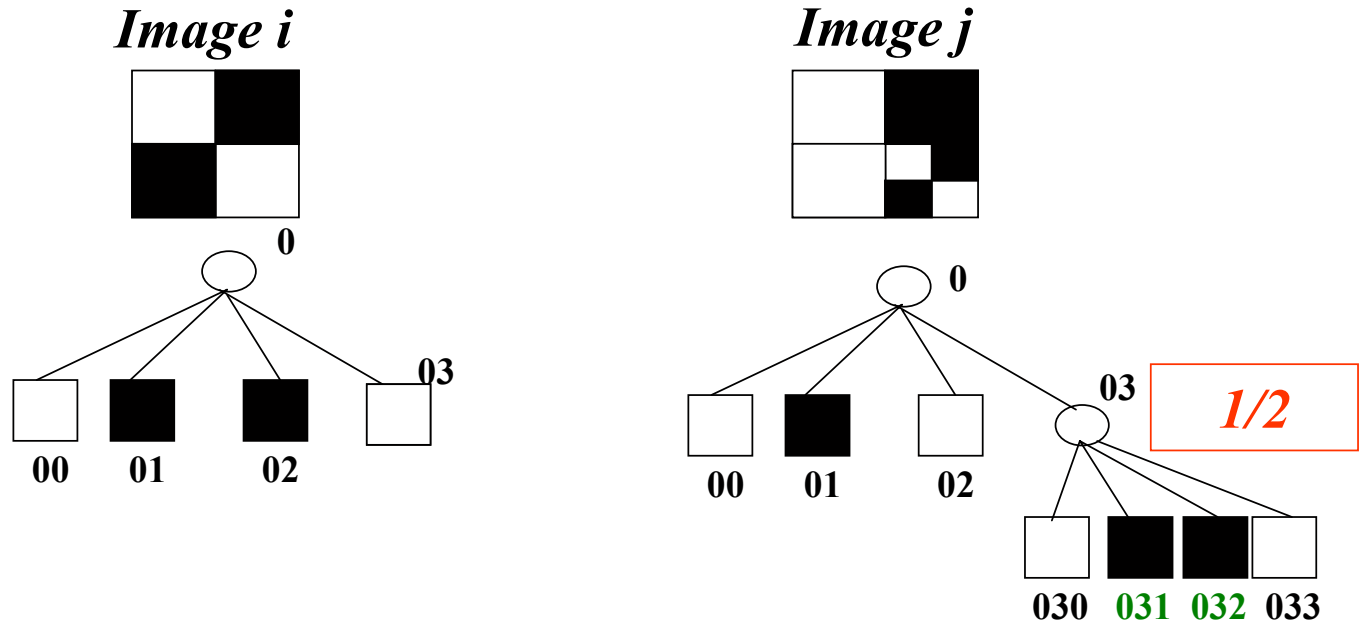
Distance visuelle entre images organisées en arbre quaternaire



$$V^{(1)}(i, j) = \frac{1}{4} + \frac{1}{4} = \frac{1}{2}$$

# Distance V

Distance visuelle entre images organisées en arbre quaternaire



$$\tilde{V}^{(1)}(i, j) = \frac{1}{4} + \left( \left| 0 - \frac{1}{2} \right| * \frac{1}{4} \right) = \frac{3}{8}$$

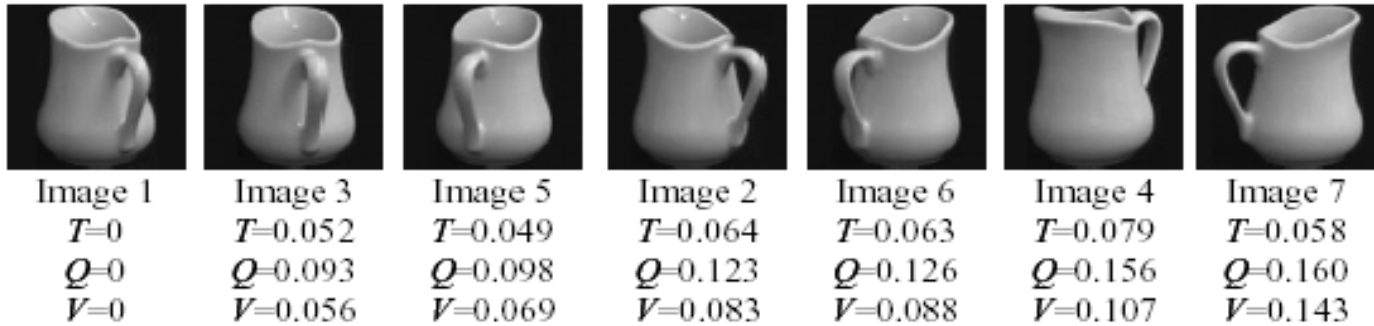


Figure 9. Images of Figure 7 ordered by their  $\mathcal{V}$ -Distance values from image 1, using the contrast criterion with a threshold equal to 1.

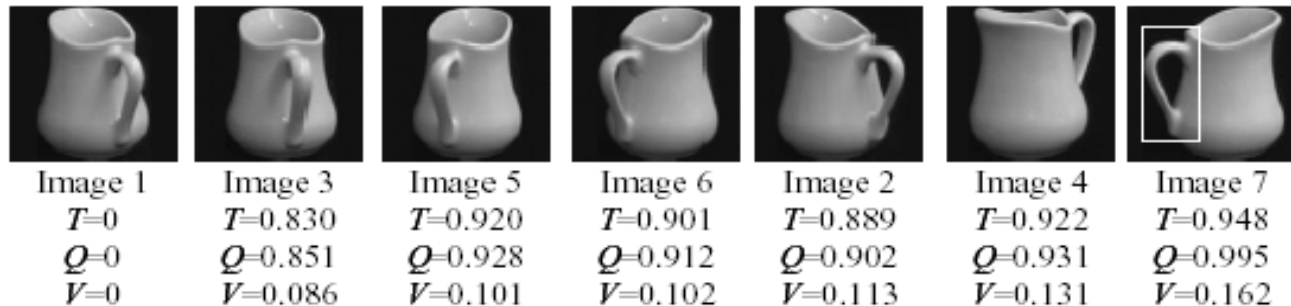


Figure 10. Images of Figure 7 ordered by their  $\mathcal{V}$ -Distance values from image 1, using the uniformity criterion, with a 20-threshold.

Images décomposées en *quadtree* et ordonnées en fonction de leur  $\Delta$ -distance.

Images issues du logiciel Ikona <http://www-rocq.inria.fr/imedia/ikona/>



Image 1  
 $V=0$



Image 7  
 $V=0.117$

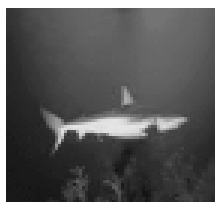


Image 6  
 $V=0.135$



Image 2  
 $V=0.154$

(a) Using the contrast criterion with 1 - threshold



Image 1  
 $V=0$



Image 7  
 $V=0.106$

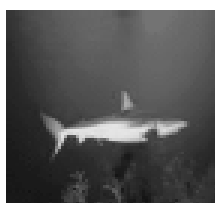


Image 6  
 $V=0.144$



Image 2  
 $V=0.155$

(b) Using the uniformity criterion with a 10 - threshold



Image 1  
 $V=0$



Image 7  
 $V=0.101$



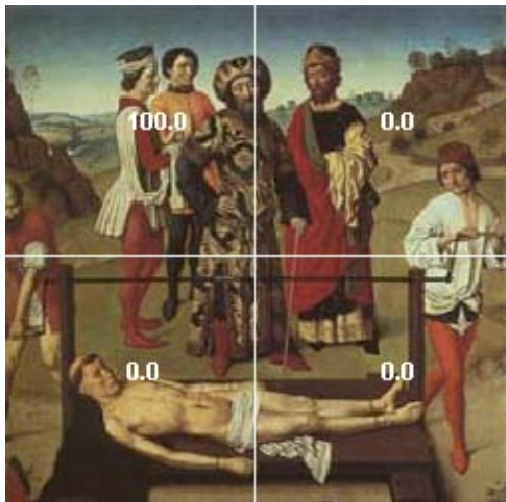
Image 2  
 $V=0.134$



Image 6  
 $V=0.141$

(c) Using the uniformity criterion with a 20 - threshold

**Images décomposées en *quadtree* et ordonnées en fonction de leur  $V$ -distance.**



(a) A query image with  $w_{00}=1$



(b) Returned images ordered by  $Q$ -distance



(c) Images not returned by the query

**Rank 1   Rank 2   Rank 3   Rank 4   Rank 5   Rank 6   Rank 7**



**Original image** 513 x 900   **Rotated and resized image** 900 x 411   **Modified resized image** 513 x 900   **Rotated and resized** 500x900   **Resized image** 236x236   **Resized image** 118x118   **Blurred image** 513x900



**(b) Query result using a global distance (computing  $V^{(0)}$  –distance)**

**(a)**  
**Query image**   **Rank 1   Rank 2   Rank 3   Rank 4   Rank 5   Rank 6   Rank 7**

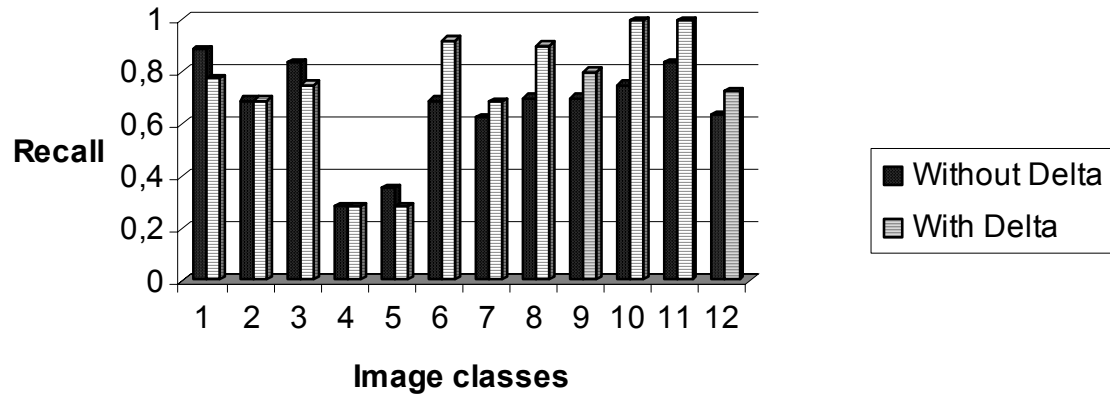
513 x 900



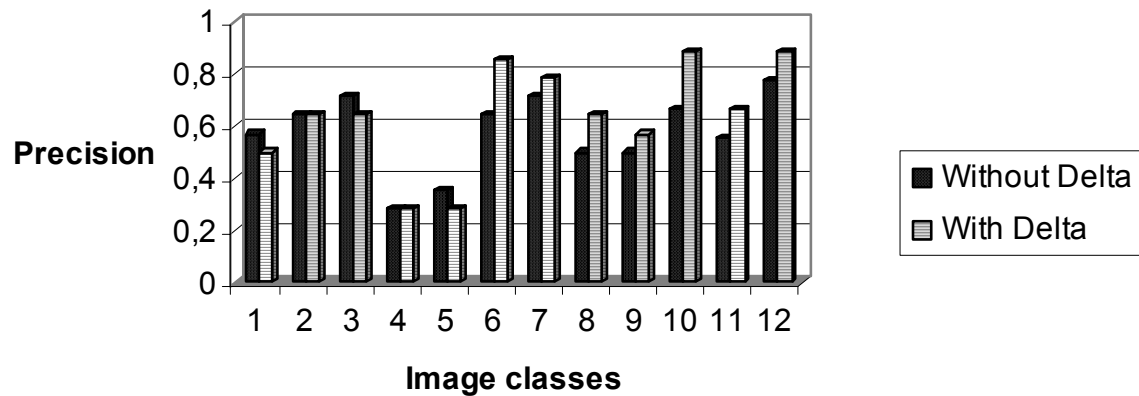
**Original image** 513 x 900   **Modified image resized** 513 x 900   **Blurred image** 513x900   **Resized image** 236x236   **Resized image** 118x118   **Rotated and resized** 500x900   **Similar image** 513x900

**(c) Query result using V-distance**

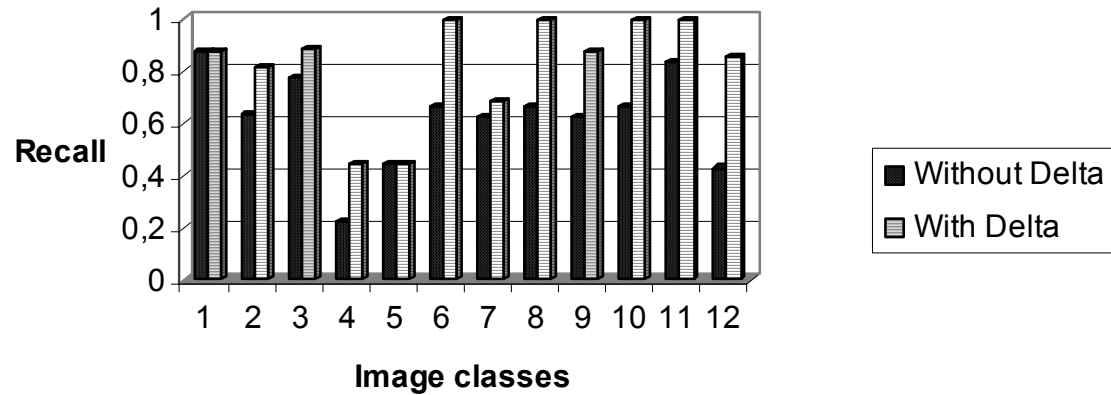
**Recall with global search criteria**



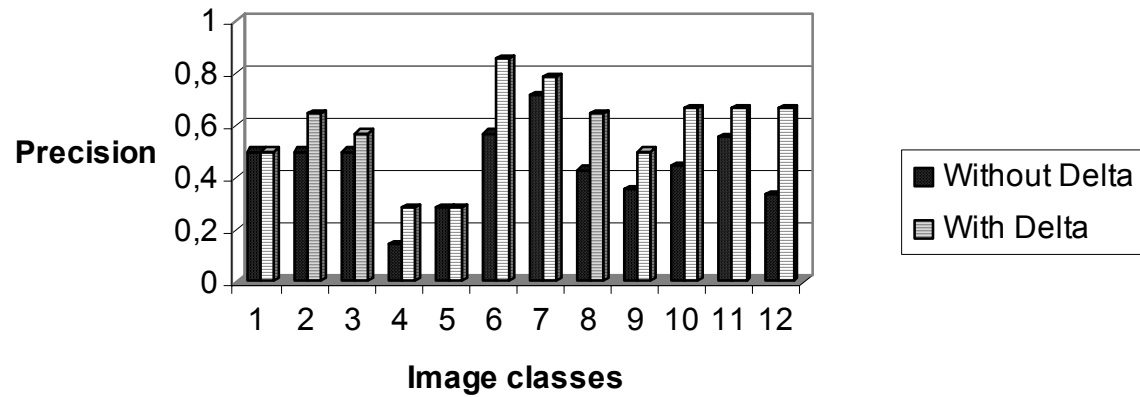
**Precision with global search criteria**



**Recall with local search criteria**



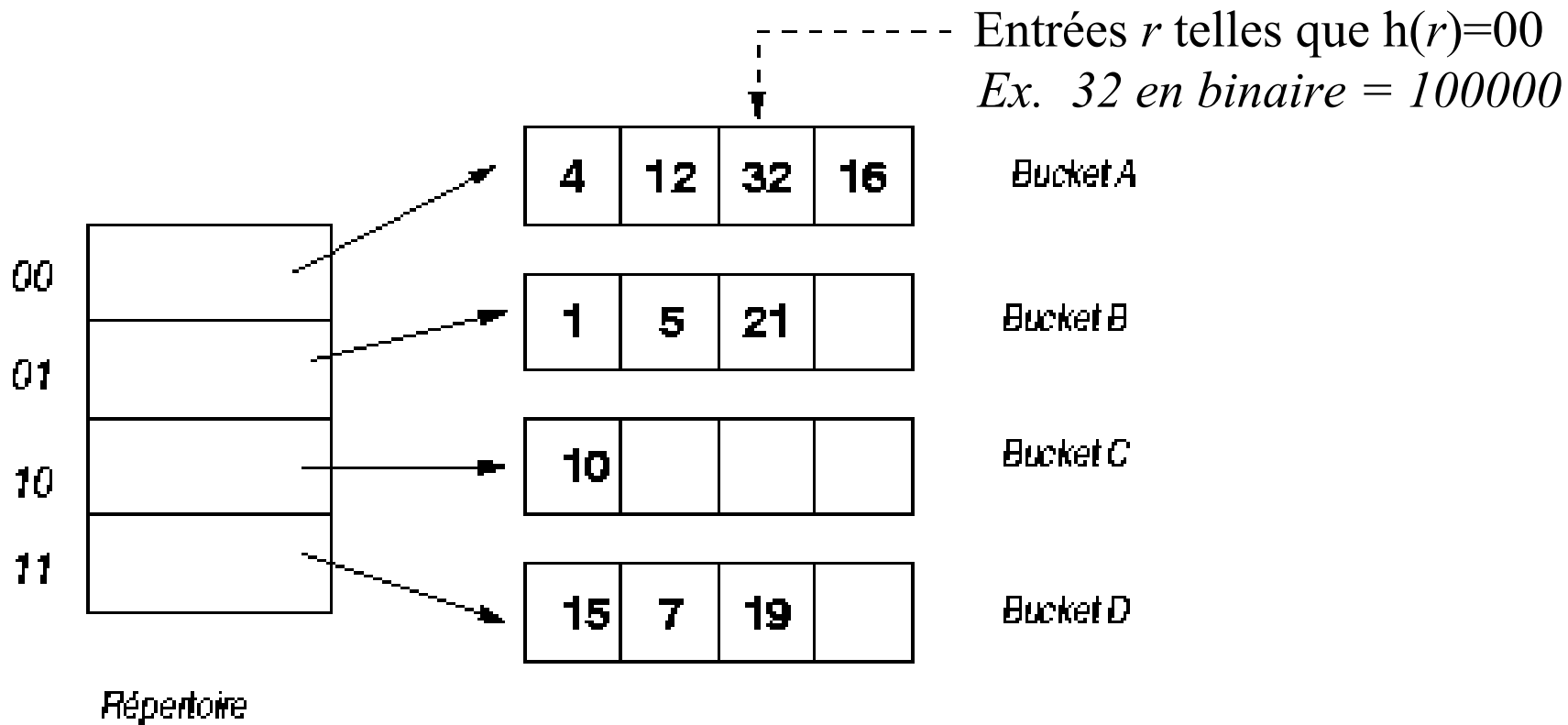
**Precision with local search criteria**



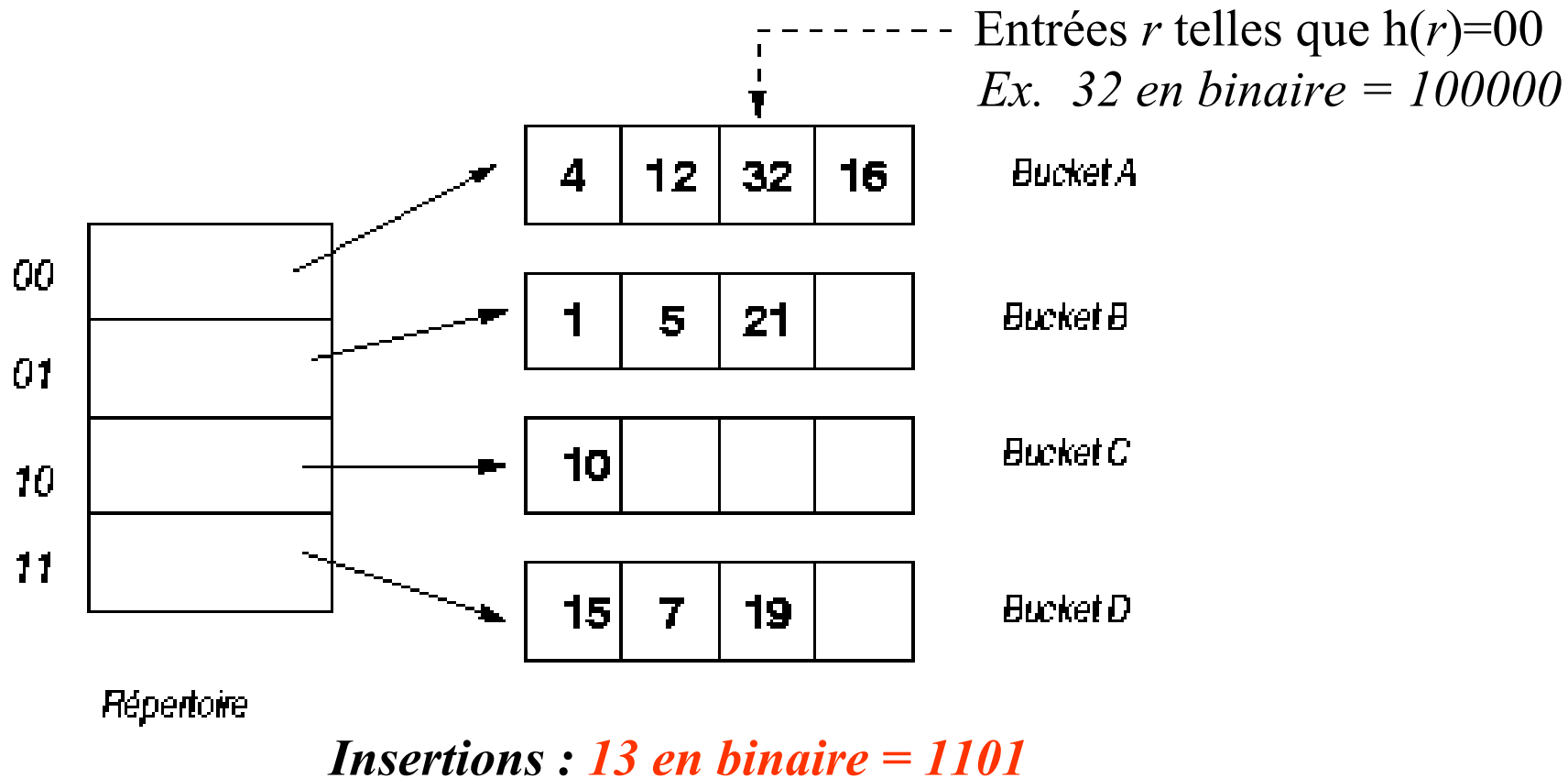
# Indexation d'images similaires à l'aide d'un arbre quaternaire

- Utilisation du hachage extensible
- Utilisation du QUIP-tree

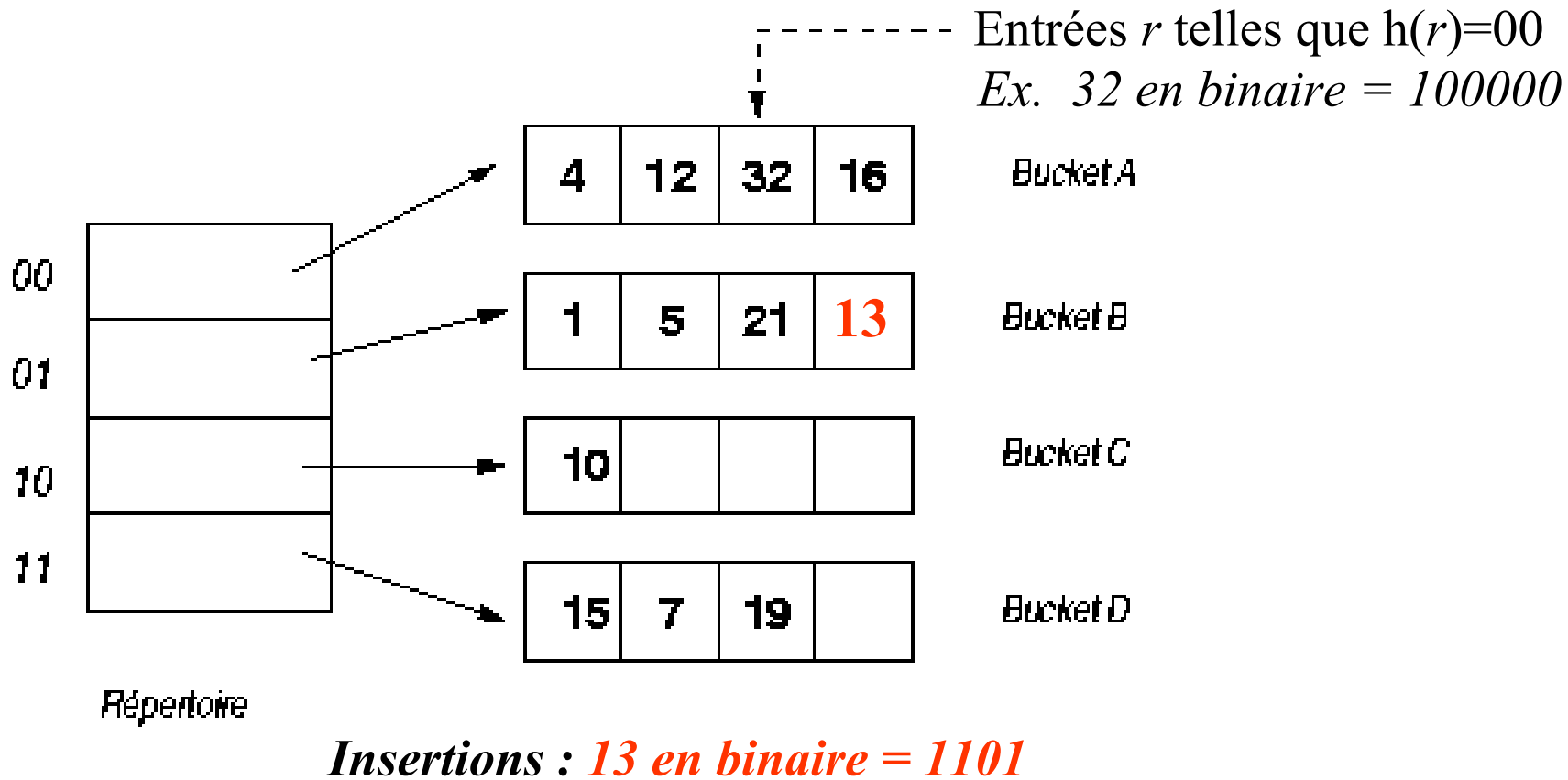
# Hachage extensible (1/2)



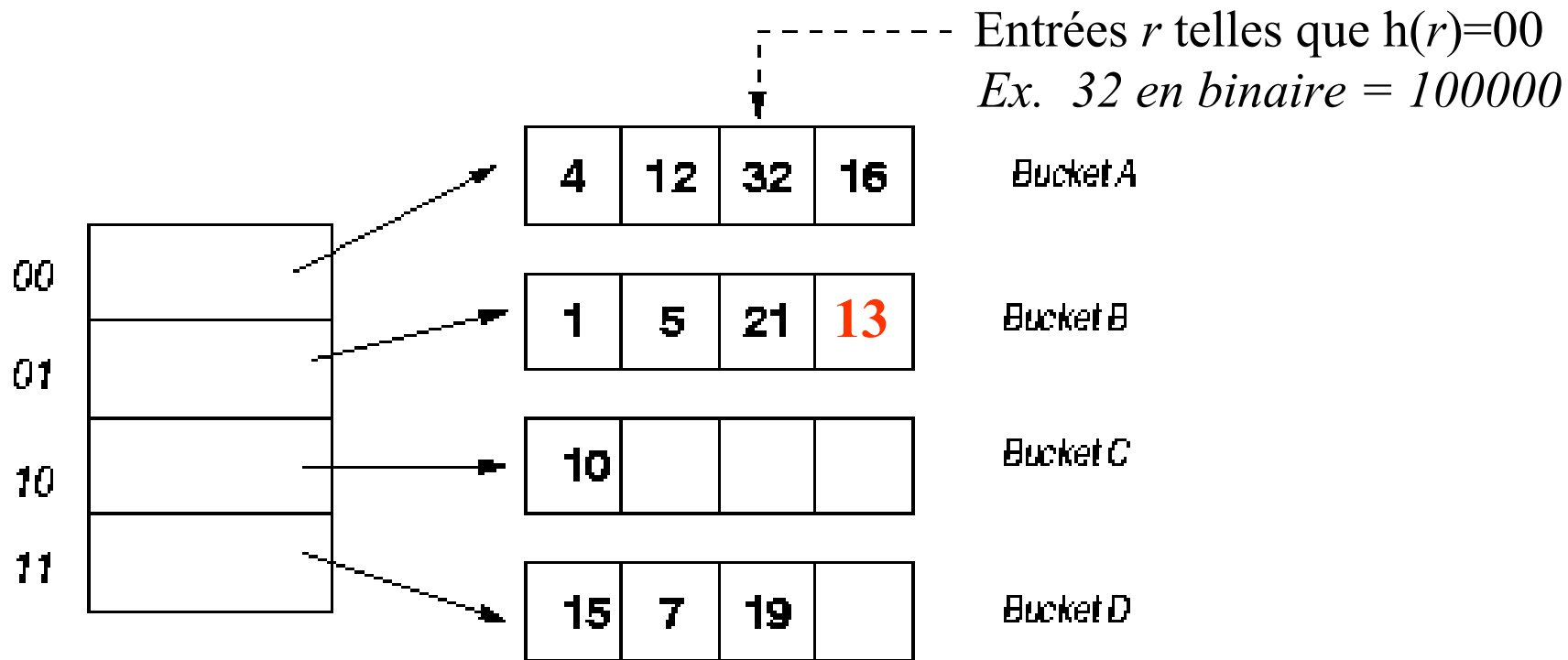
# Hachage extensible (1/2)



# Hachage extensible (1/2)



# Hachage extensible (1/2)



*Insertions : 13 en binaire = 1101*

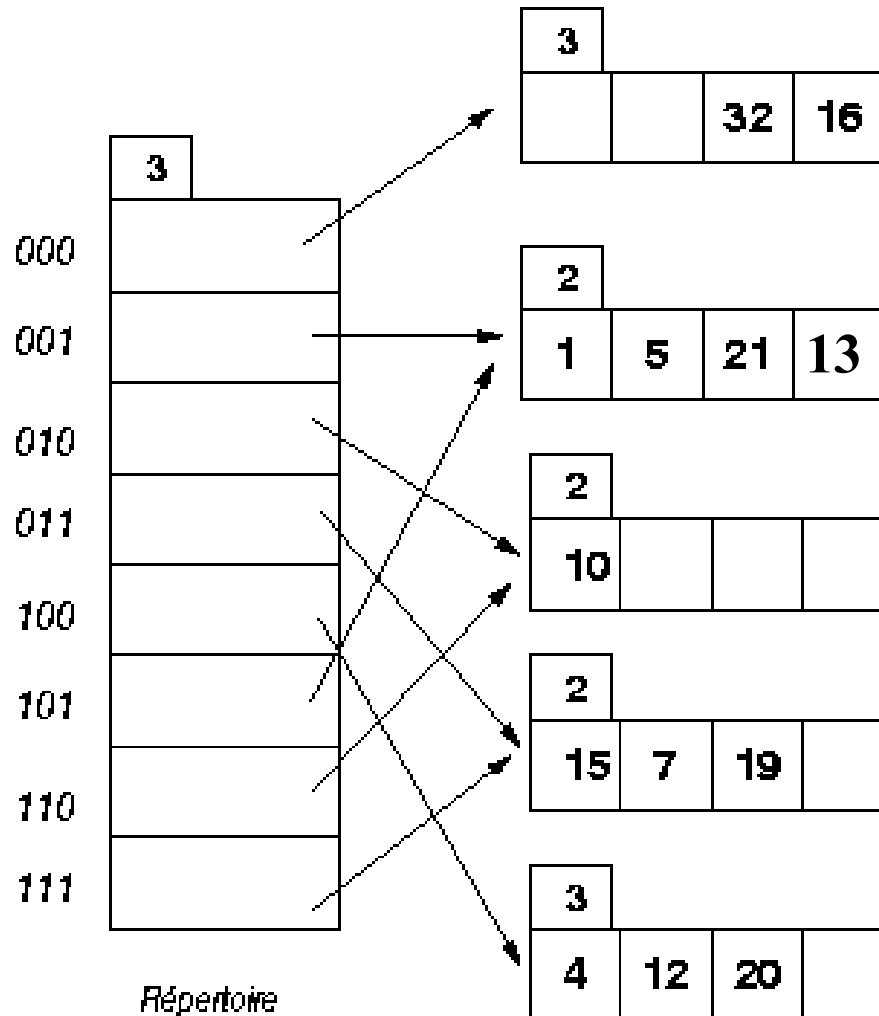
*20 en binaire = 10100 mais bucket A plein*

# Hachage extensible (2/2)

**La taille du  
répertoire  
est doublée  
et le bucket  
A est divisé  
en 2**

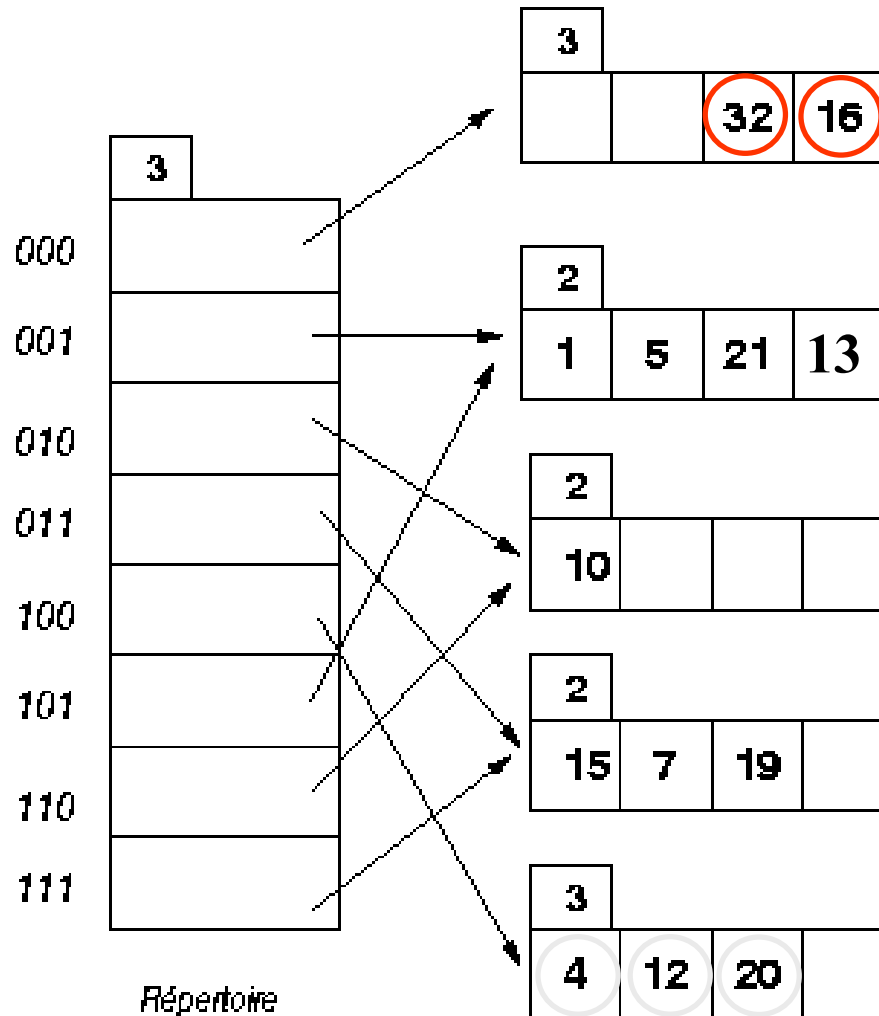
# Hachage extensible (2/2)

La taille du  
répertoire  
est doublée  
et le bucket  
A est divisé  
en 2



# Hachage extensible (2/2)

La taille du répertoire est doublée et le bucket A est divisé en 2



$h(4)=100$

$h(12)=100$

$h(16) = 000$

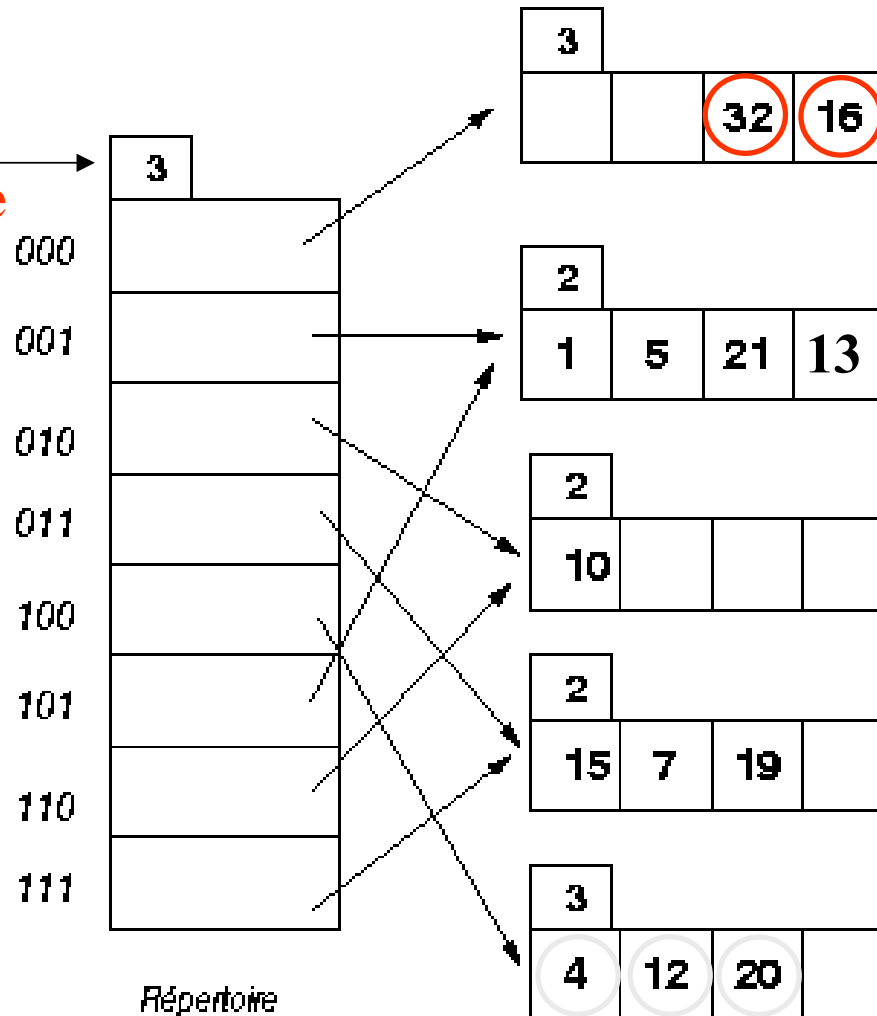
$h(32)=000$

$h(20) = 100$

# Hachage extensible (2/2)

Taille globale

La taille du répertoire est doublée et le bucket A est divisé en 2



$h(4)=100$

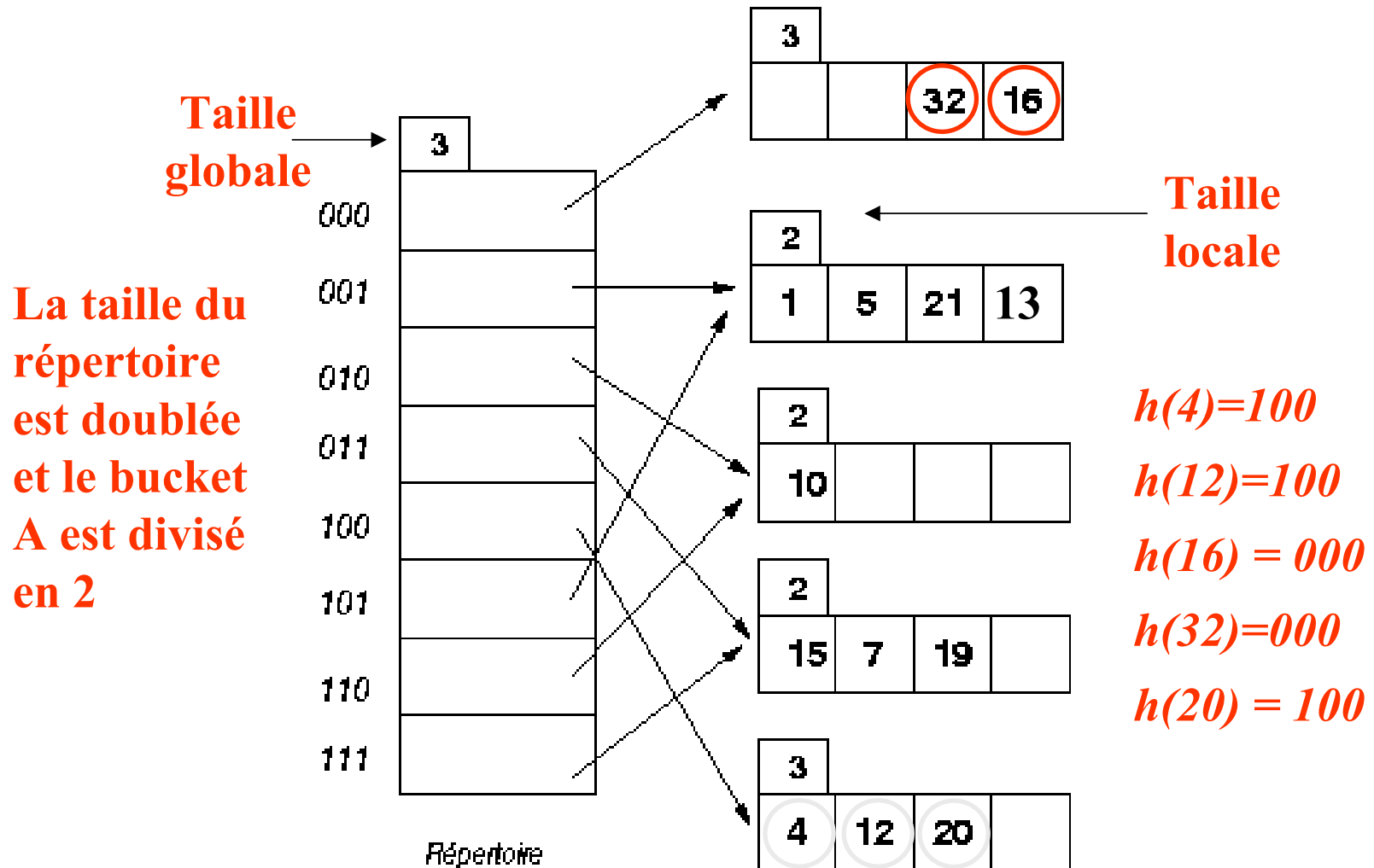
$h(12)=100$

$h(16) = 000$

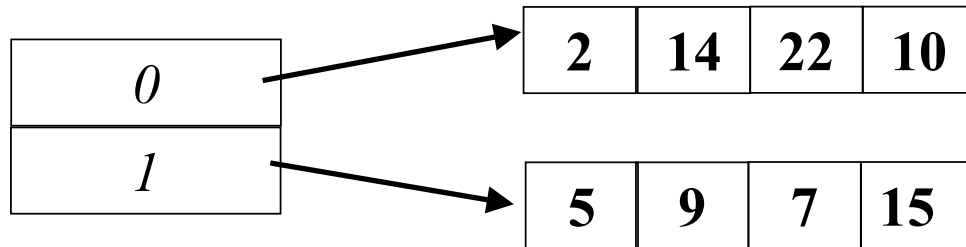
$h(32)=000$

$h(20) = 100$

# Hachage extensible (2/2)



# Autre exemple de hachage extensible (1/4)



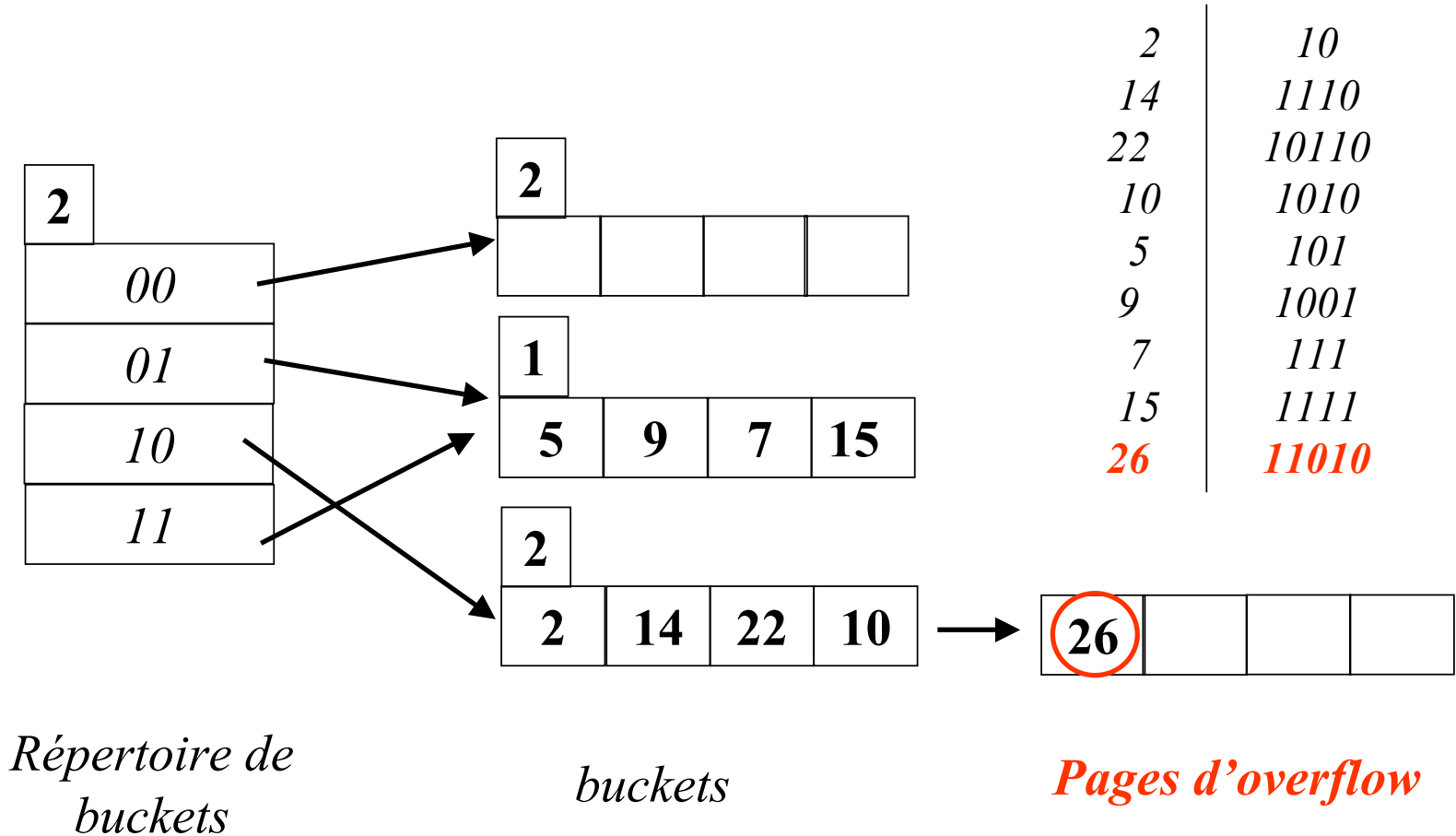
*Répertoire de  
buckets*

*buckets*

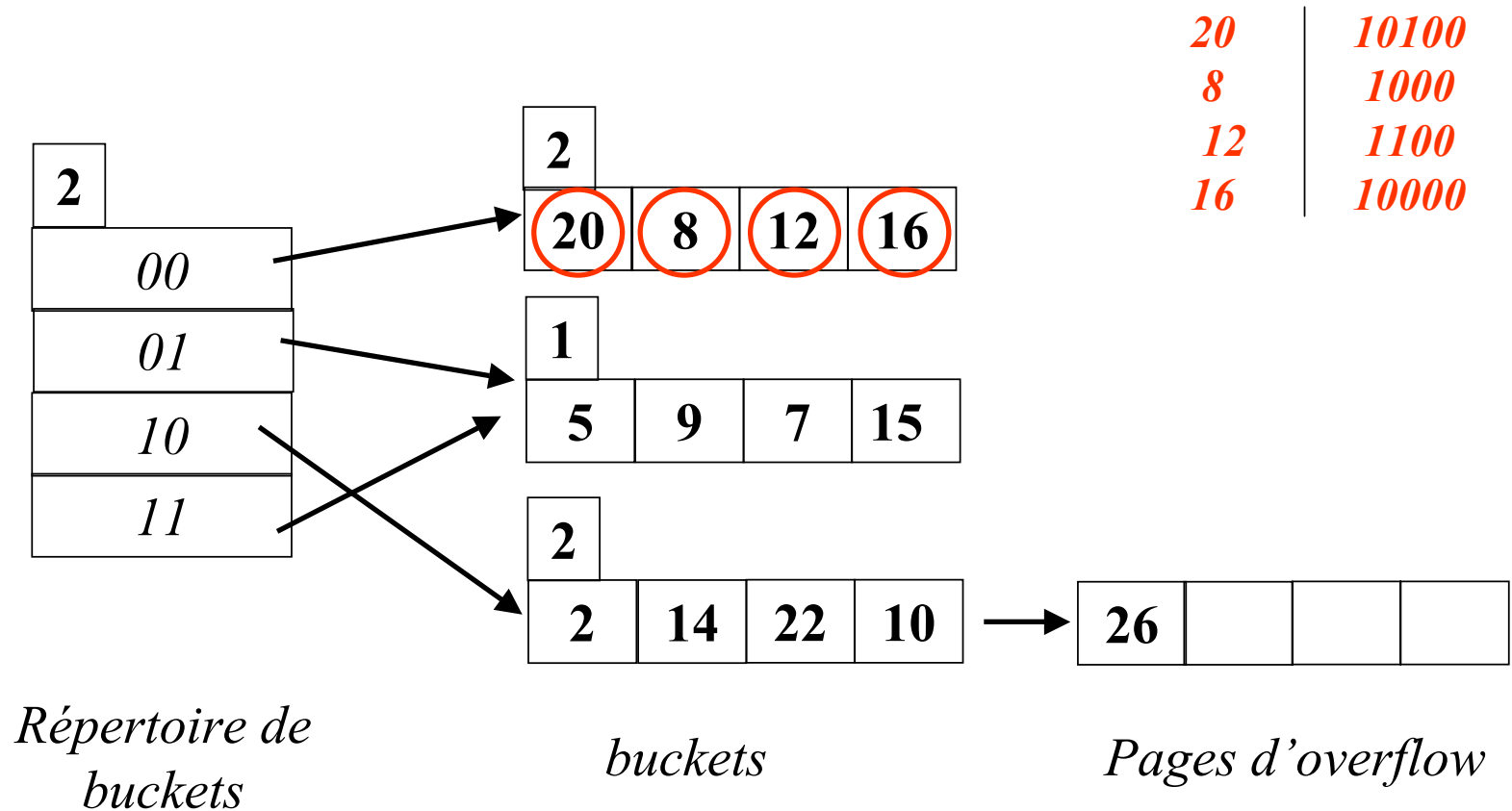
**On ajoute l'entrée 26 (11010)**

|    |       |
|----|-------|
| 2  | 10    |
| 14 | 1110  |
| 22 | 10110 |
| 10 | 1010  |
| 5  | 101   |
| 9  | 1001  |
| 7  | 111   |
| 15 | 1111  |

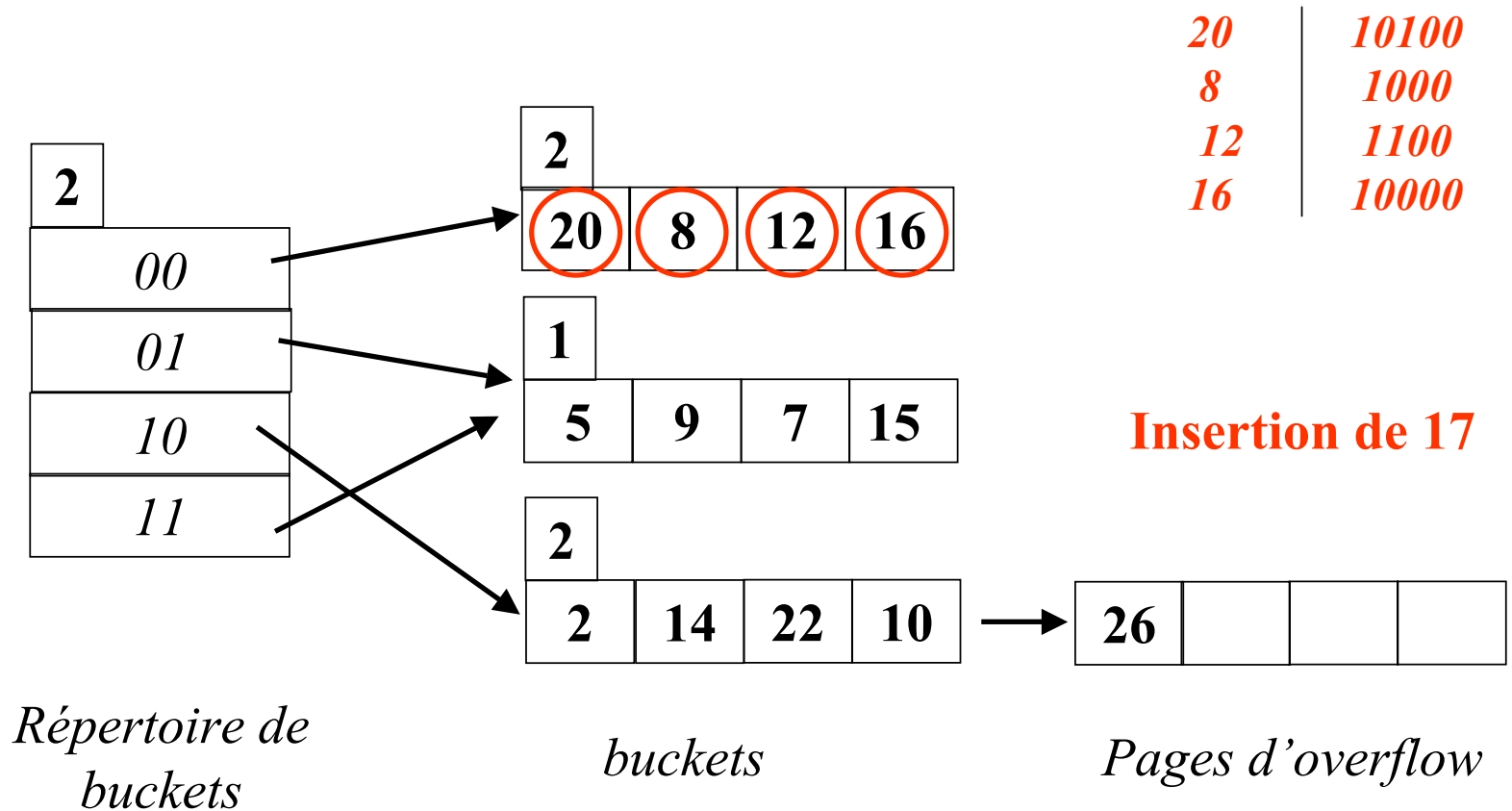
# Autre exemple de hachage extensible (2/4)



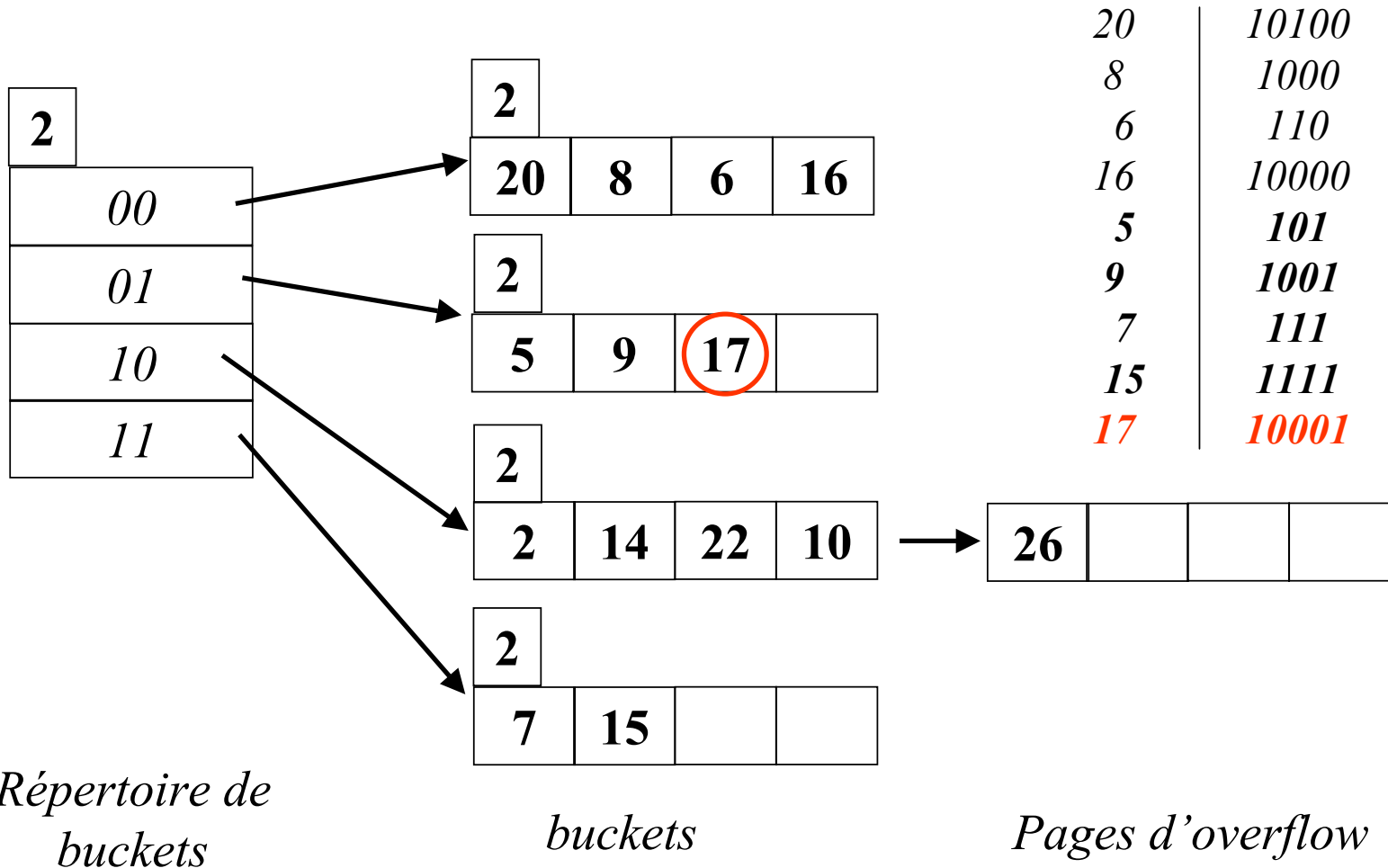
# Autre exemple de hachage extensible (3/4)



# Autre exemple de hachage extensible (3/4)

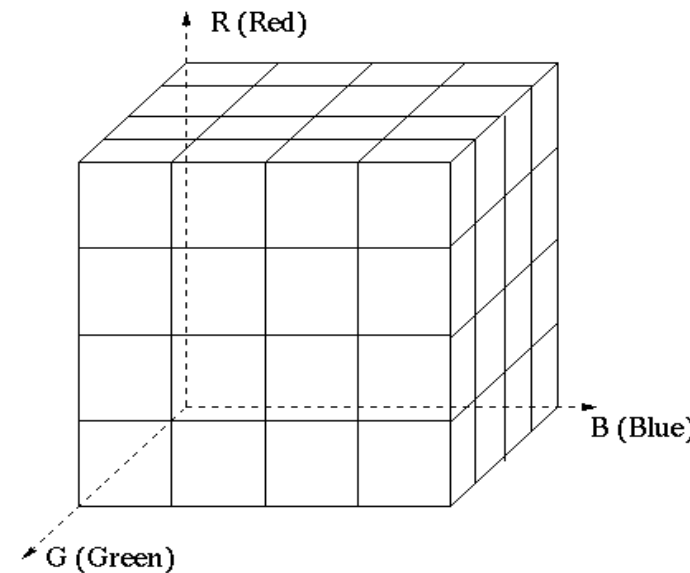
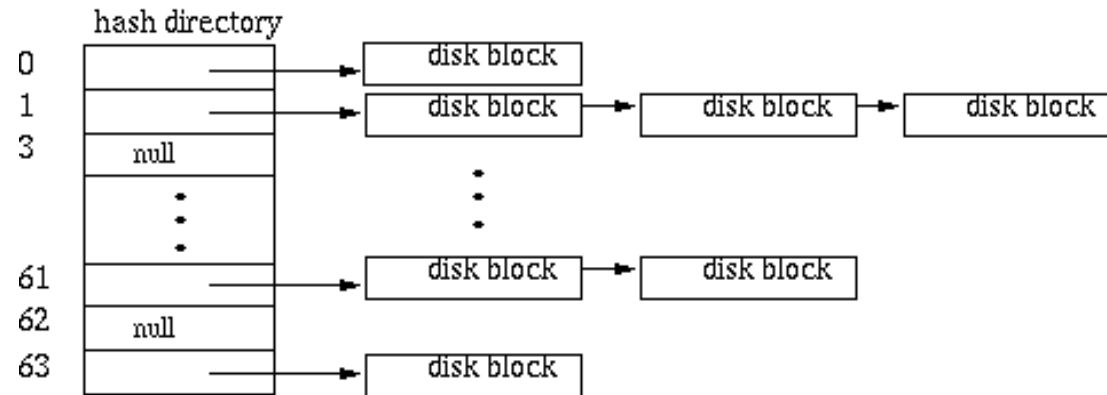


# Autre exemple de hachage extensible (4/4)



# Three-dimensional Extendible Hashing (3DEH) [LTO+01]

- Index three-dimensional color values
- A static structure



$$H(R, G, B) = \lfloor R \div 64 \rfloor \times 16 + \lfloor G \div 64 \rfloor \times 4 + \lfloor B \div 64 \rfloor$$

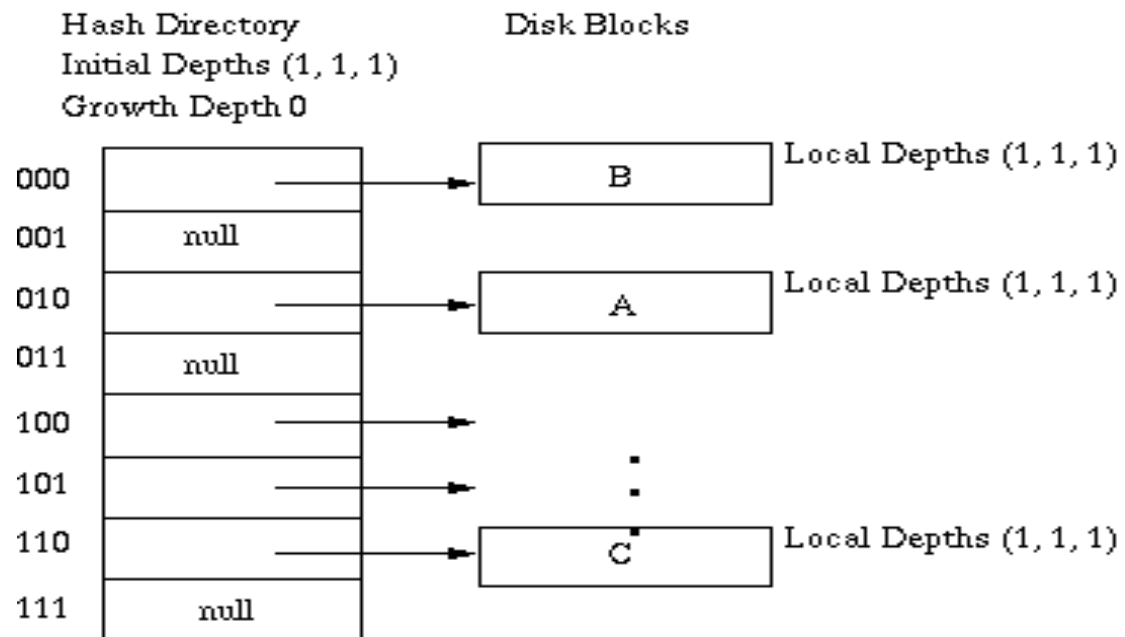
$$R, G, B \in [0, 255]$$

e. g.  $H(21, 147, 208) = 11$

i. e.  $H(00010101, 10010011, 11010000) = 001011$

# 3DEH: Initial State [LTO+01]

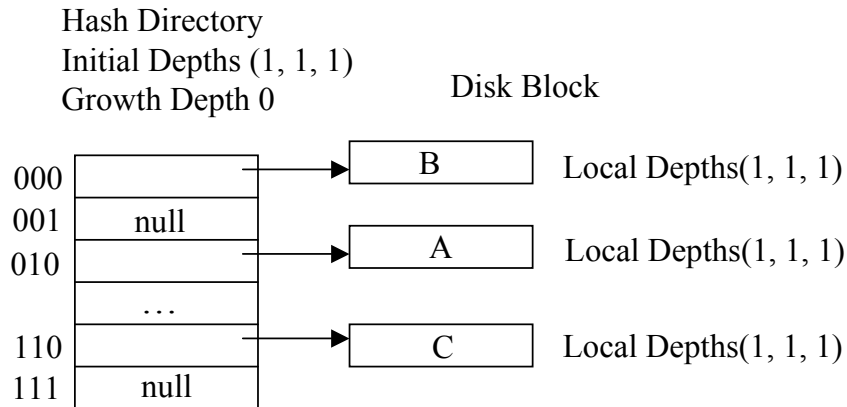
- ◆ Similar to the classic structure
- ◆ The size depends on the number of bits per color component



e. g. (01111100, 10101000, 00011001) is hashed to entry 010

# 3DEH: Insertion with split and directory doubling [LTO+01]

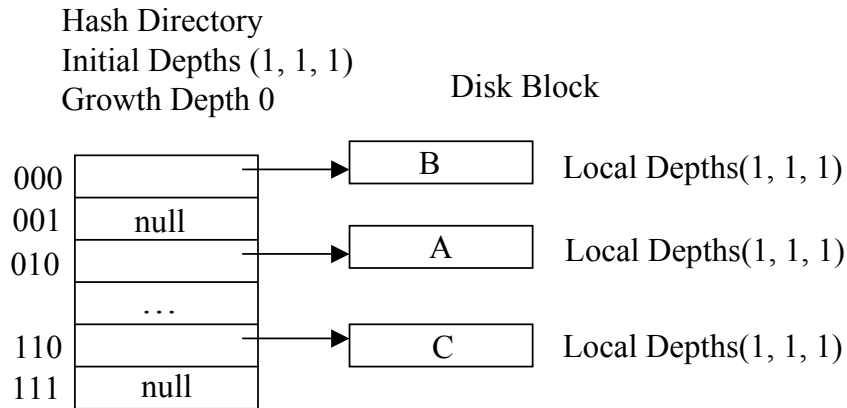
- Split along the dimension with the highest variance



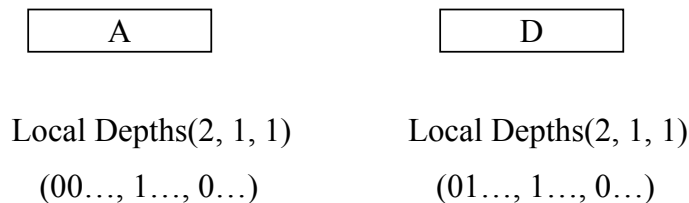
- Bucket A (010) overflows along the R dimension
- The directory doubles
- A new bucket D(1010) is allocated

# 3DEH: Insertion with split and directory doubling [LTO+01]

- Split along the dimension with the highest variance

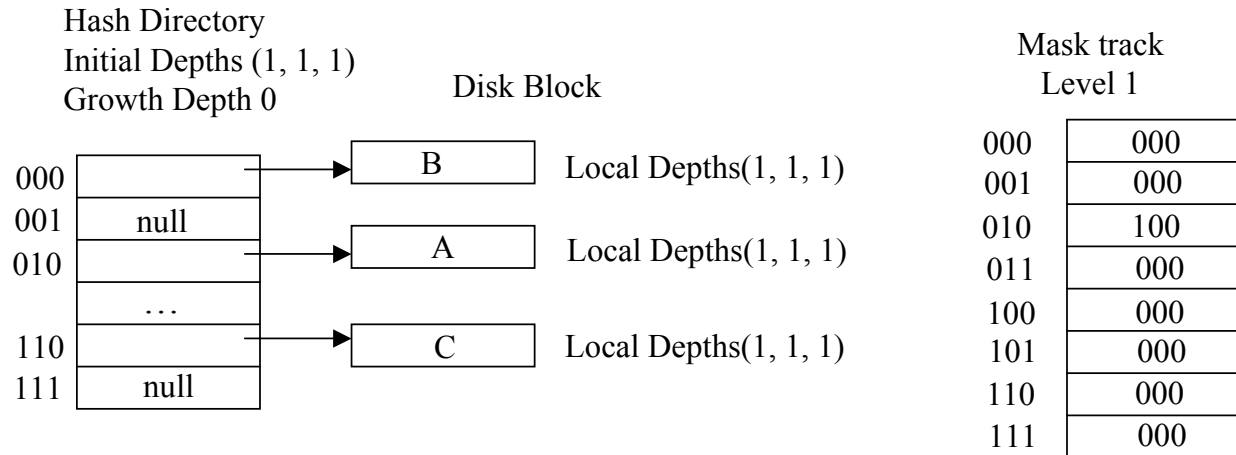


- Bucket A (010) overflows along the R dimension
- The directory doubles
- A new bucket D(1010) is allocated

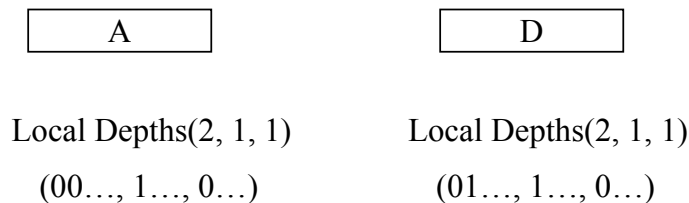


# 3DEH: Insertion with split and directory doubling [LTO+01]

- Split along the dimension with the highest variance



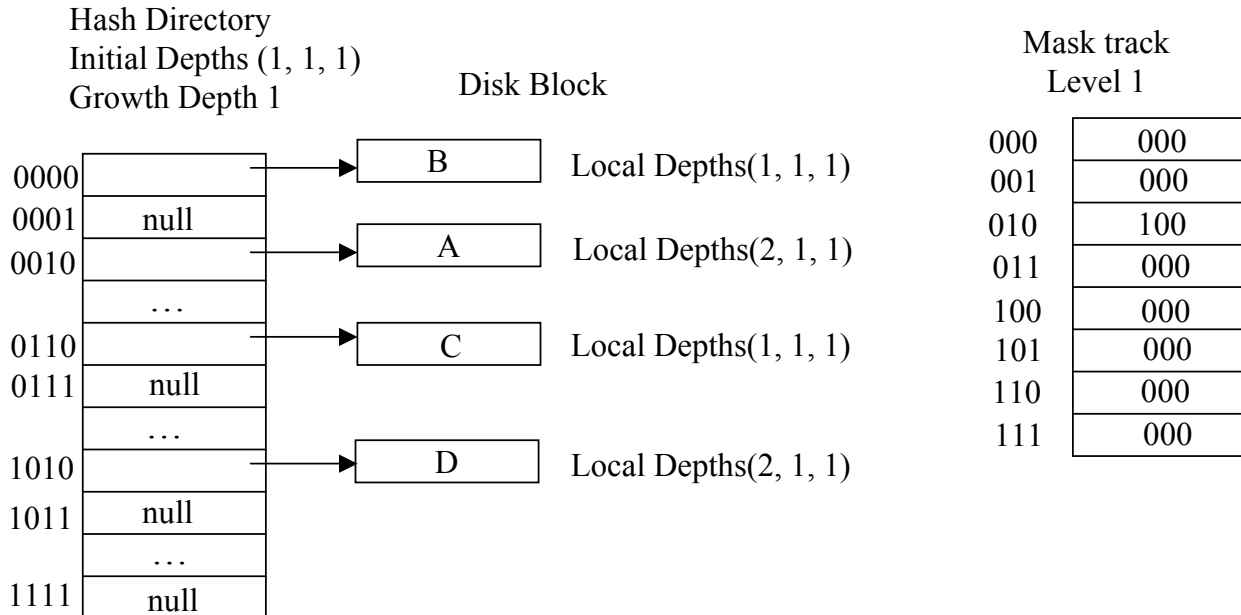
- Bucket A (010) overflows along the R dimension
- The directory doubles
- A new bucket D(1010) is allocated



# 3DEH: Insertion with split and directory doubling (cont) [LTO+01]



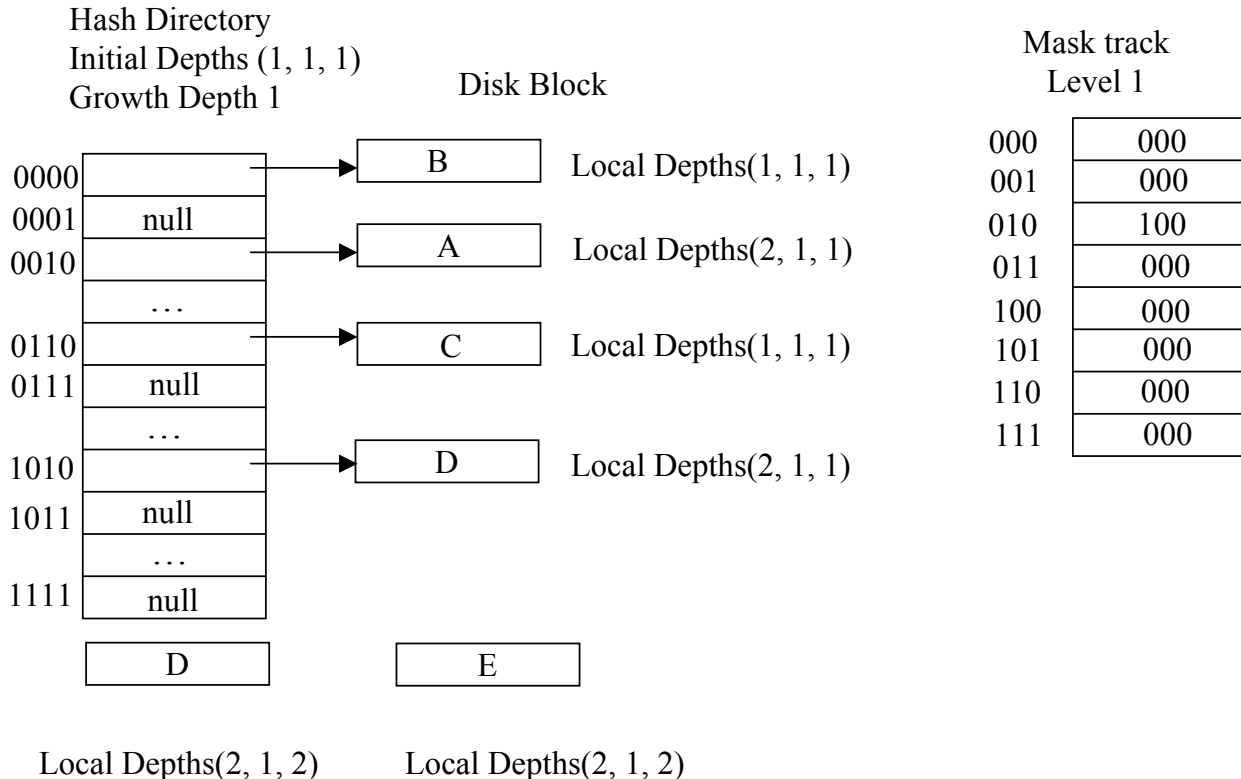
- D(1010) overflows along the B dimension
- The directory doubles; A new bucket E(11010) is allocated



# 3DEH: Insertion with split and directory doubling (cont) [LTO+01]

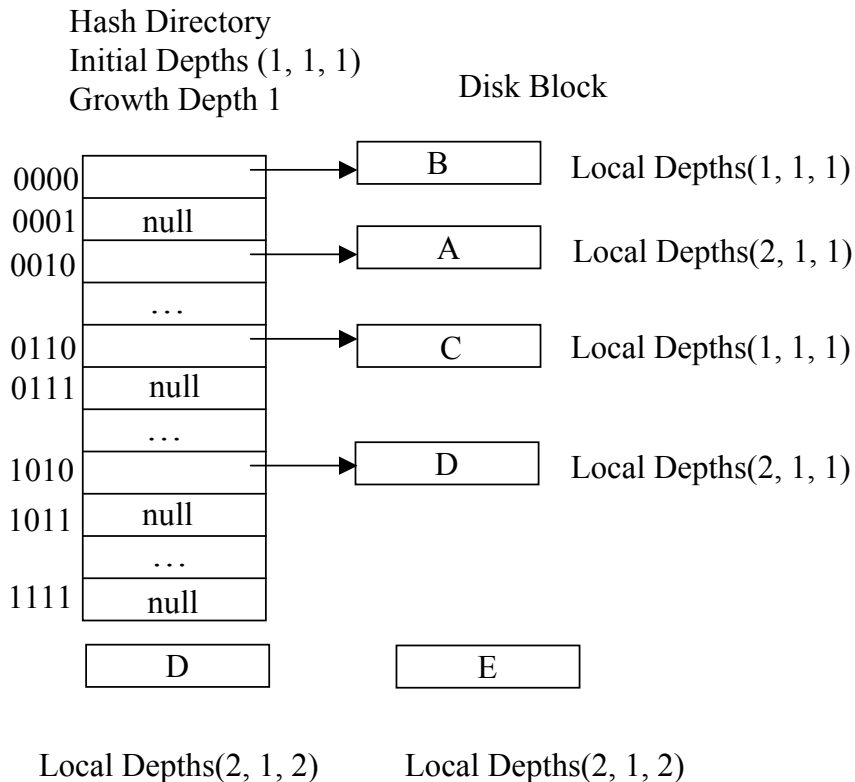


- D(1010) overflows along the B dimension
- The directory doubles; A new bucket E(11010) is allocated



# 3DEH: Insertion with split and directory doubling (cont) [LTO+01]

- D(1010) overflows along the B dimension
- The directory doubles; A new bucket E(11010) is allocated



| Mask track |     | Level 2 |     |
|------------|-----|---------|-----|
| Level 1    |     |         |     |
| 000        | 000 | 0000    | 000 |
| 001        | 000 | 0001    | 000 |
| 010        | 100 | 0010    | 000 |
| 011        | 000 | 0011    | 000 |
| 100        | 000 | 0100    | 000 |
| 101        | 000 | 0101    | 000 |
| 110        | 000 | 0110    | 000 |
| 111        | 000 | 0111    | 000 |
|            |     | 1000    | 000 |
|            |     | 1001    | 000 |
|            |     | 1010    | 001 |
|            |     | 1011    | 000 |
|            |     | 1100    | 000 |
|            |     | 1101    | 000 |
|            |     | 1110    | 000 |
|            |     | 1111    | 000 |

# 3DEH: Summary [LTO+01]

- Growing of the hash structure
  - ◆ When a disk block overflows, it splits along the dimension with the highest variance.
  - ◆ The hash directory doubles when no entry is available to accommodate the new allocated disk block.
  - ◆ A data structure named mask track is used to keep track of the splitting history of disk blocks.



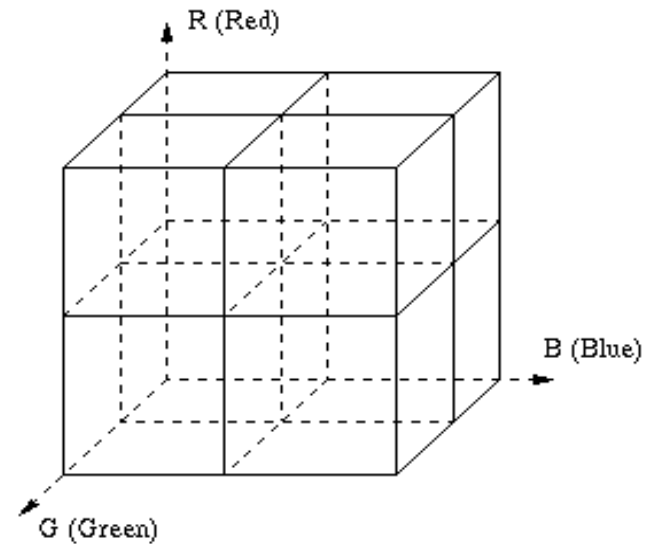
# Range Query Algorithm [LTO+01]

- Compute the set of initial partitions that overlap the minimum bounding cube of the search region.



# Range Query Algorithm [LTO+01]

- Compute the set of initial partitions that overlap the minimum bounding cube of the search region.



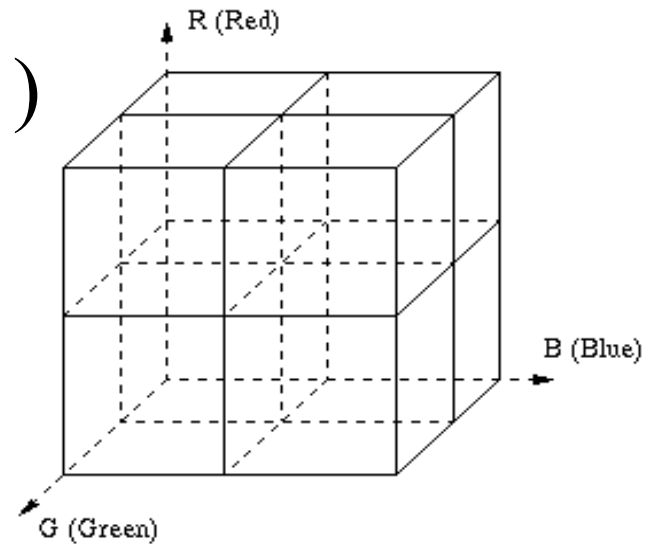


# Range Query Algorithm [LTO+01]

- Compute the set of initial partitions that overlap the minimum bounding cube of the search region.

$$(r, g, b) = (124, 168, 25)$$

$$\varepsilon = 12$$



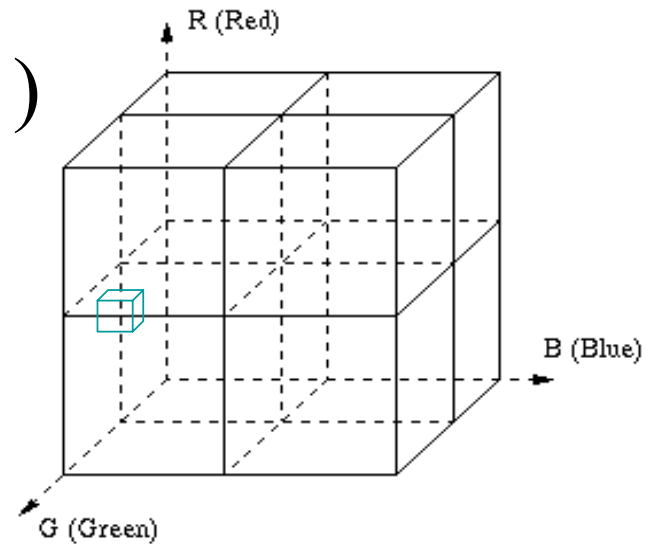


# Range Query Algorithm [LTO+01]

- Compute the set of initial partitions that overlap the minimum bounding cube of the search region.

$$(r, g, b) = (124, 168, 25)$$

$$\varepsilon = 12$$



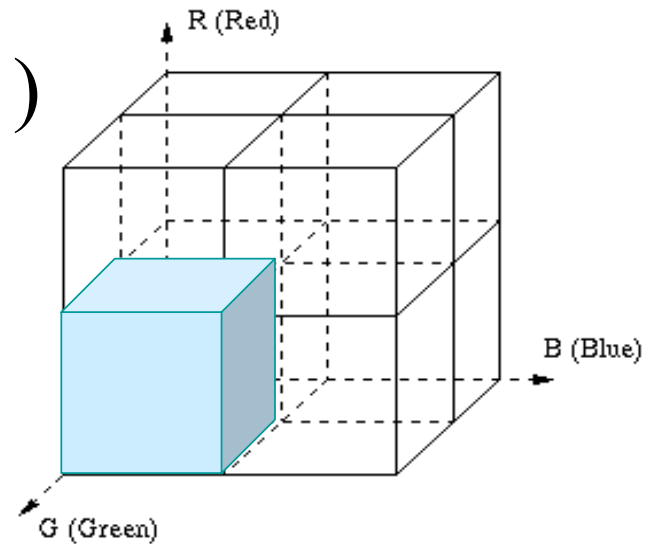


# Range Query Algorithm [LTO+01]

- Compute the set of initial partitions that overlap the minimum bounding cube of the search region.

$$(r, g, b) = (124, 168, 25)$$

$$\varepsilon = 12$$



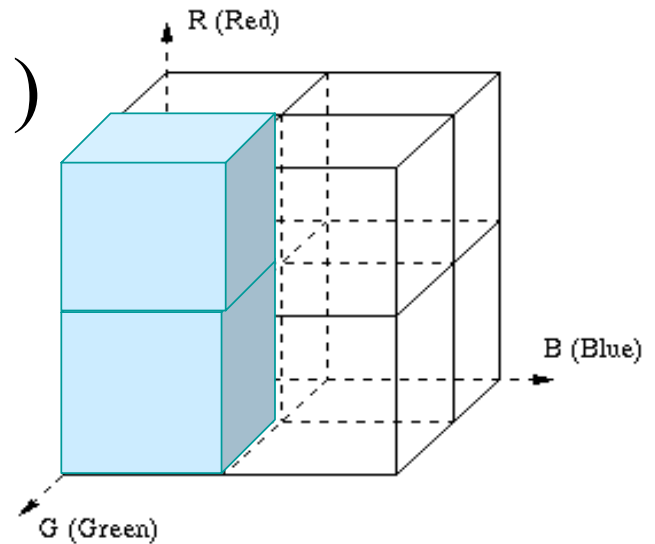


# Range Query Algorithm [LTO+01]

- Compute the set of initial partitions that overlap the minimum bounding cube of the search region.

$$(r, g, b) = (124, 168, 25)$$

$$\varepsilon = 12$$





# Range Query Algorithm [LTO+01]

- Compute the set of initial partitions that overlap the minimum bounding cube of the search region.

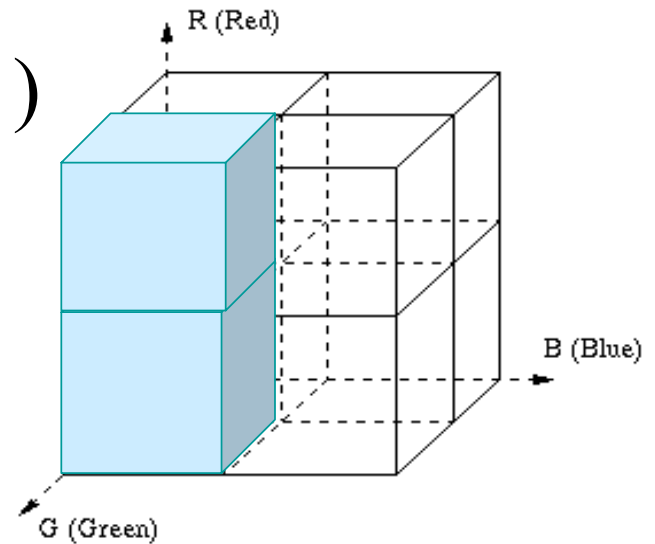
$$(r, g, b) = (124, 168, 25)$$

$$\varepsilon = 12$$

$$S = \{010, 110\}$$

$$(r, g, b) = (01111100, 10101000, 00011001)$$

$$(r, g, b) \pm 12 = ([0111000, 10001000], [10011100, 10110100], [00001101, 00100101])$$

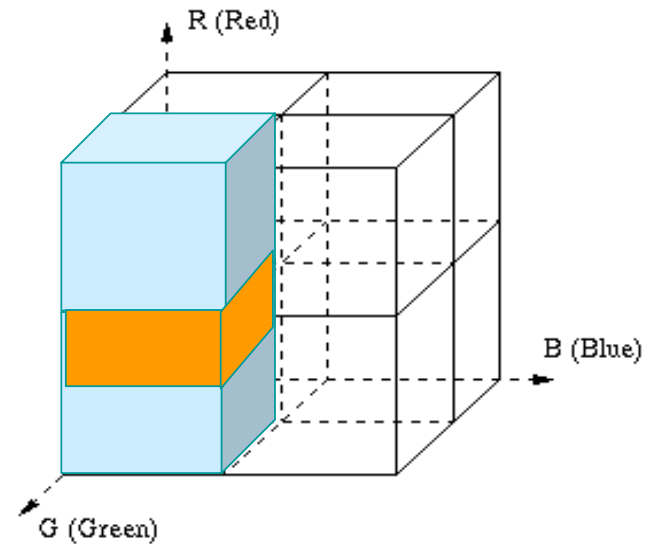




# Range Query Algorithm (Cont'd) [LTO+01]

- Look up the mask track to find out how the initial partitions have been further partitioned.

$$S = \{010, 110\}$$



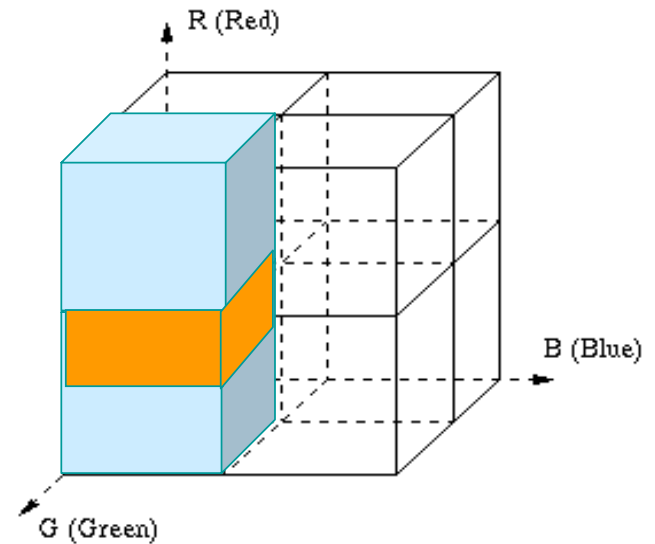


# Range Query Algorithm (Cont'd) [LTO+01]

- Look up the mask track to find out how the initial partitions have been further partitioned.

$$S = \{010, 110\}$$

$$S = \{1010, 110\}$$



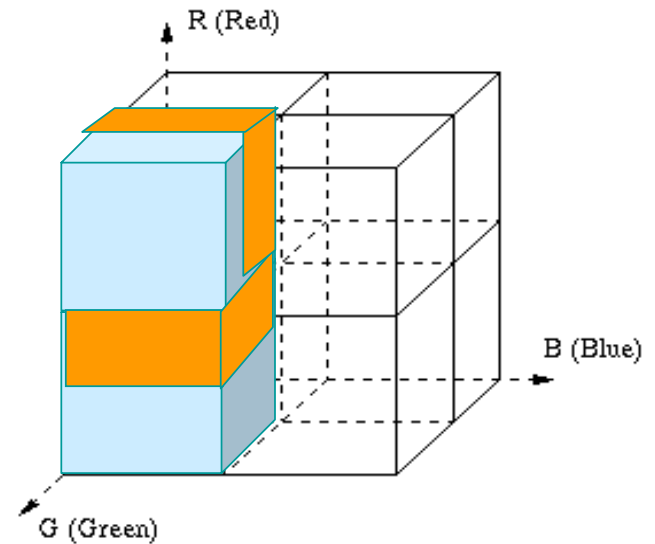


# Range Query Algorithm (Cont'd) [LTO+01]

- Look up the mask track to find out how the initial partitions have been further partitioned.

$$S = \{010, 110\}$$

$$S = \{1010, 110\}$$





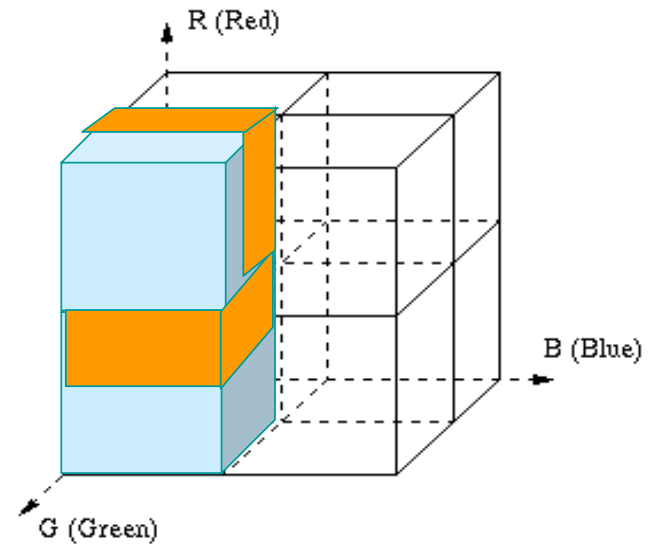
# Range Query Algorithm (Cont'd) [LTO+01]

- Look up the mask track to find out how the initial partitions have been further partitioned.

$$S = \{010, 110\}$$

$$S = \{1010, 110\}$$

$$S = \{1010, 0110\}$$





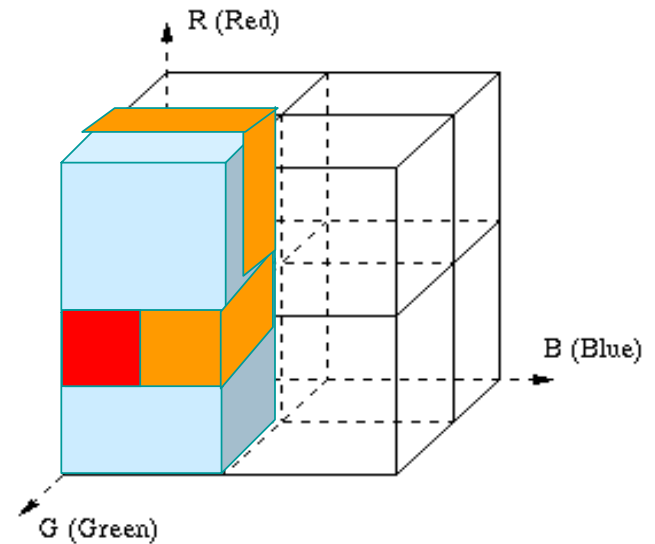
# Range Query Algorithm (Cont'd) [LTO+01]

- Look up the mask track to find out how the initial partitions have been further partitioned.

$$S = \{010, 110\}$$

$$S = \{1010, 110\}$$

$$S = \{1010, 0110\}$$





# Range Query Algorithm (Cont'd) [LTO+01]

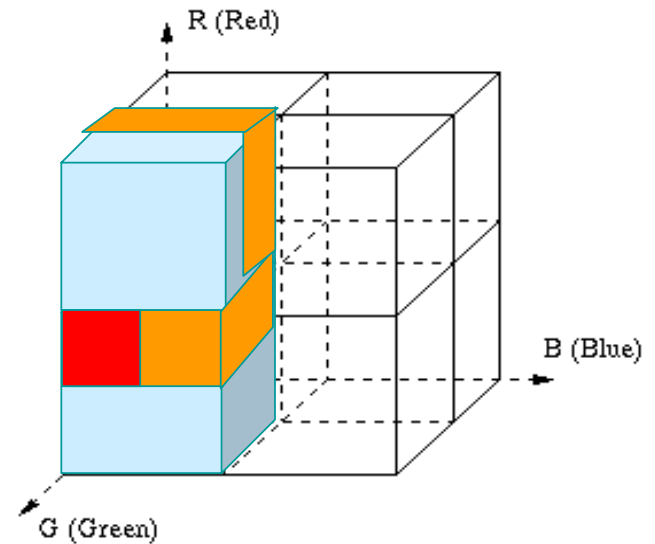
- Look up the mask track to find out how the initial partitions have been further partitioned.

$$S = \{010, 110\}$$

$$S = \{1010, 110\}$$

$$S = \{1010, 0110\}$$

$$S = \{01010, 0110\}$$





# Range Query Algorithm (Cont'd) [LTO+01]

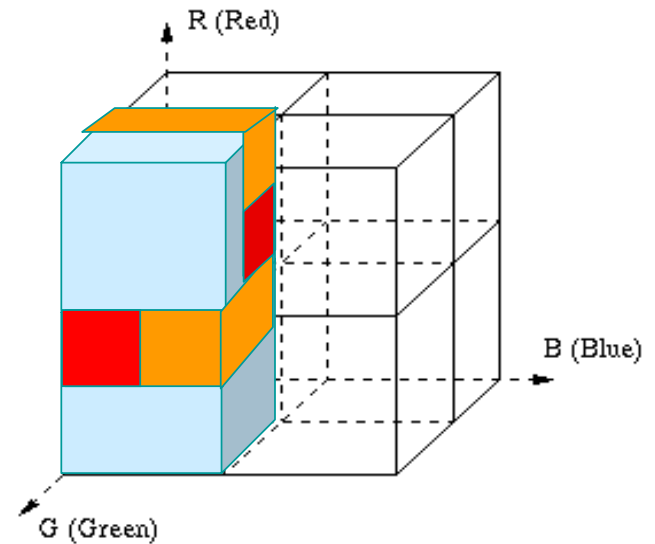
- Look up the mask track to find out how the initial partitions have been further partitioned.

$$S = \{010, 110\}$$

$$S = \{1010, 110\}$$

$$S = \{1010, 0110\}$$

$$S = \{01010, 0110\}$$





# Range Query Algorithm (Cont'd) [LTO+01]

- Look up the mask track to find out how the initial partitions have been further partitioned.

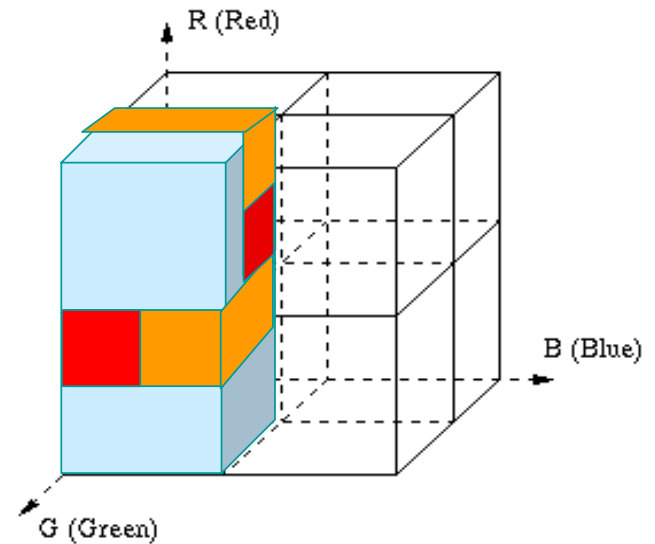
$$S = \{010, 110\}$$

$$S = \{1010, 110\}$$

$$S = \{1010, 0110\}$$

$$S = \{01010, 0110\}$$

$$S = \{01010, 00110\}$$



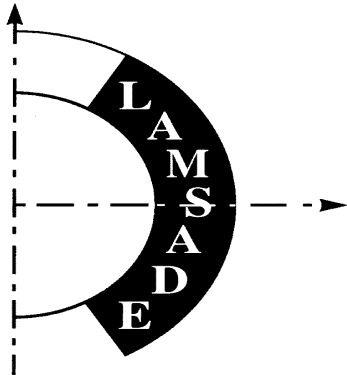
# Indexation multi-niveau pour la recherche globale et partielle d'images par le contenu\*

Geneviève Jomier<sup>1</sup>

Maude Manouvrier<sup>1</sup>

Vincent Oria<sup>2</sup>

Marta Rukoz<sup>3</sup>



\* Réalisé en partie dans le cadre d'une coopération scientifique CNRS-FONICIT/CDCH

1. LAMSADE - Université Paris-Dauphine - France

2. New Jersey Institute of Technology - USA

3. CCPD - Université Centrale du Venezuela - Caracas

# Plan de l'exposé

- Recherche d'images par le contenu
- Similarité d'images représentées par des arbres quaternaires
- *Quadtree-based Index for Image Retrieval (QUIP-tree)*
- Résultats d'expérimentation
- Conclusion

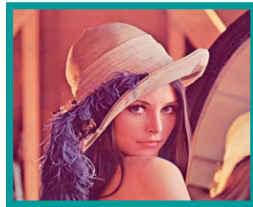
# **Recherche par le contenu d'images représentées par des arbres quaternaires**

- **Similarité tenant compte de la localisation des pixels**
- **Filtrage multi-niveau**
- **Possibilité de requête sur des régions**

# Recherche par le contenu d'images représentées par des arbres quaternaires

- Similarité tenant compte de la localisation des pixels
- Filtrage multi-niveau
- Possibilité de requête sur des régions

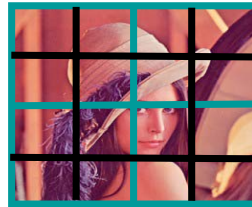
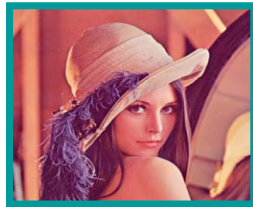
(a) Image requête



# Recherche par le contenu d'images représentées par des arbres quaternaires

- Similarité tenant compte de la localisation des pixels
- Filtrage multi-niveau
- Possibilité de requête sur des régions

(a) Image requête

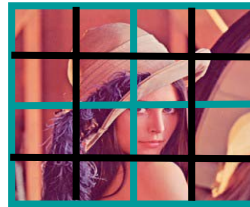
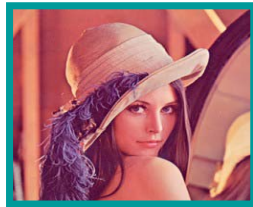


(b) Image représentée par un vecteur de caractéristiques de 3 niveaux

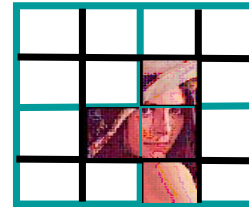
# Recherche par le contenu d'images représentées par des arbres quaternaires

- Similarité tenant compte de la localisation des pixels
- Filtrage multi-niveau
- Possibilité de requête sur des régions

(a) Image requête



(b) Image représentée par un vecteur de caractéristiques de 3 niveaux

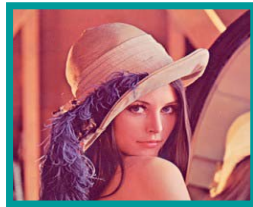


(c) Quadrants sélectionnés par l'utilisateur

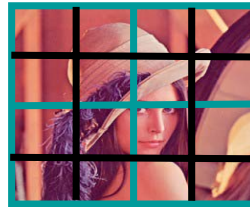
# Recherche par le contenu d'images représentées par des arbres quaternaires

- Similarité tenant compte de la localisation des pixels
- Filtrage multi-niveau
- Possibilité de requête sur des régions

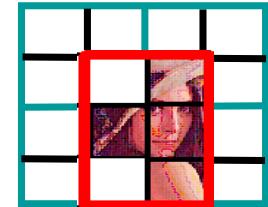
(a) Image requête



(b) Image représentée par un vecteur de caractéristiques de 3 niveaux



(c) Quadrants sélectionnés par l'utilisateur

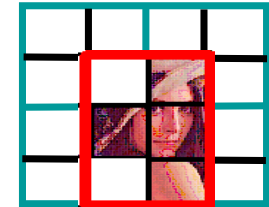
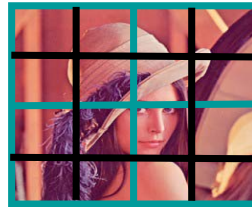
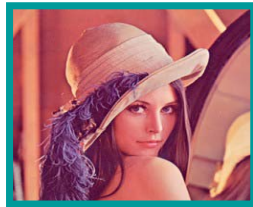


(d) Rectangle minimum englobant les quadrants sélectionnés

# Recherche par le contenu d'images représentées par des arbres quaternaires

- Similarité tenant compte de la localisation des pixels
- Filtrage multi-niveau
- Possibilité de requête sur des régions

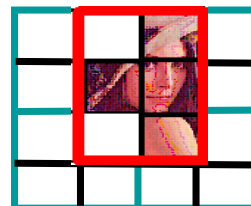
(a) Image requête



(b) Image représentée par un vecteur de caractéristiques de 3 niveaux

(c) Quadrants sélectionnés par l'utilisateur

(d) Rectangle minimum englobant les quadrants sélectionnés

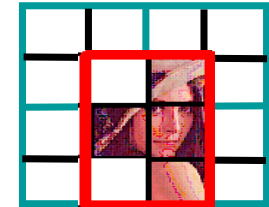
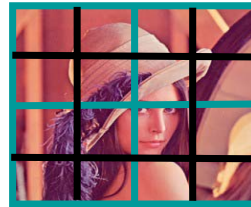
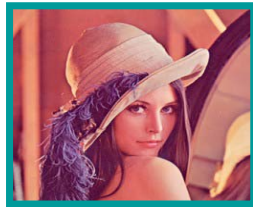


(e) Translations du rectangle englobant

# Recherche par le contenu d'images représentées par des arbres quaternaires

- Similarité tenant compte de la localisation des pixels
- Filtrage multi-niveau
- Possibilité de requête sur des régions

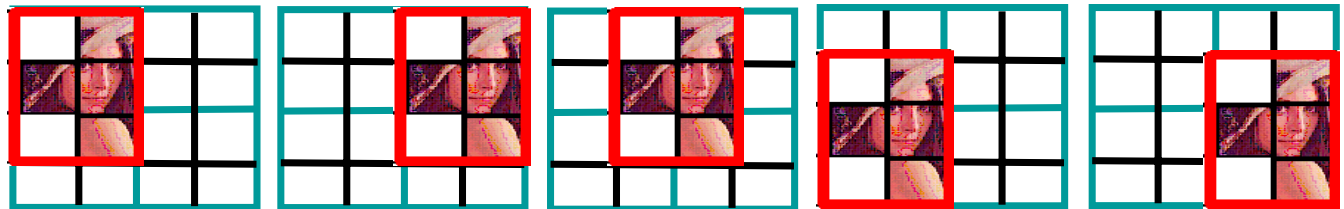
(a) Image requête



(b) Image représentée par un vecteur de caractéristiques de 3 niveaux

(c) Quadrants sélectionnés par l'utilisateur

(d) Rectangle minimum englobant les quadrants sélectionnés



(e) Translations du rectangle englobant

Adaptée de  
(Malki et al.,  
1999)

# Recherche par le contenu d'images représentées par des arbres quaternaires

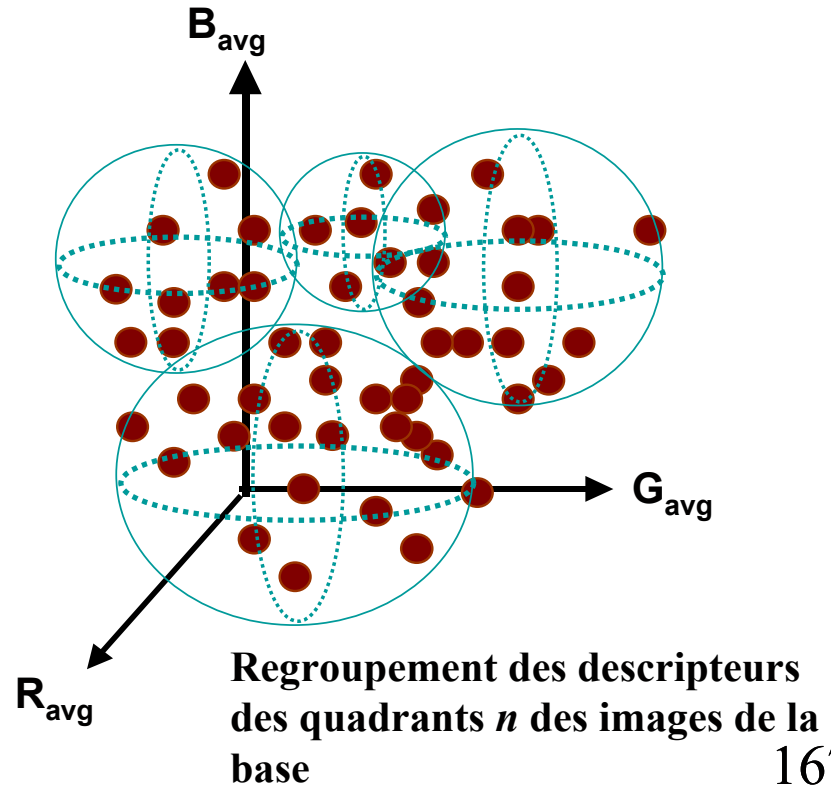
- **Similarité tenant compte de la localisation des pixels**
  - **Filtrage multi-niveau**
  - **Possibilité de requête sur des régions**
  - **Utilisation d'index sur les racines des arbres quaternaires**
    - Arbre R dans (Lu et al., 1994)
    - Arbre k-d dans (Malki et al., 1999)
    - Hachage extensible dans (Lin et al., 2001)
- ⇒ Pour diminuer le nombre de comparaison des vecteurs globaux
- ⇒ Pour comparer exhaustivement les arbres quaternaires que d'un sous-ensemble d'images globalement similaires

# **Notre approche : le QUIP-tree** ***(Quadtree-based Index for Image Retrieval)***

- **Représentation de tous les quadrants d'images dans un même arbre quaternaire, l'arbre QUIP**
- **Regroupement des quadrants homologues similaires**

# Notre approche : le QUIP-tree (*Quadtree-based Index for Image Retrieval*)

- Représentation de tous les quadrants d'images dans un même arbre quaternaire, l'arbre QUIP
- Regroupement des quadrants homologues similaires

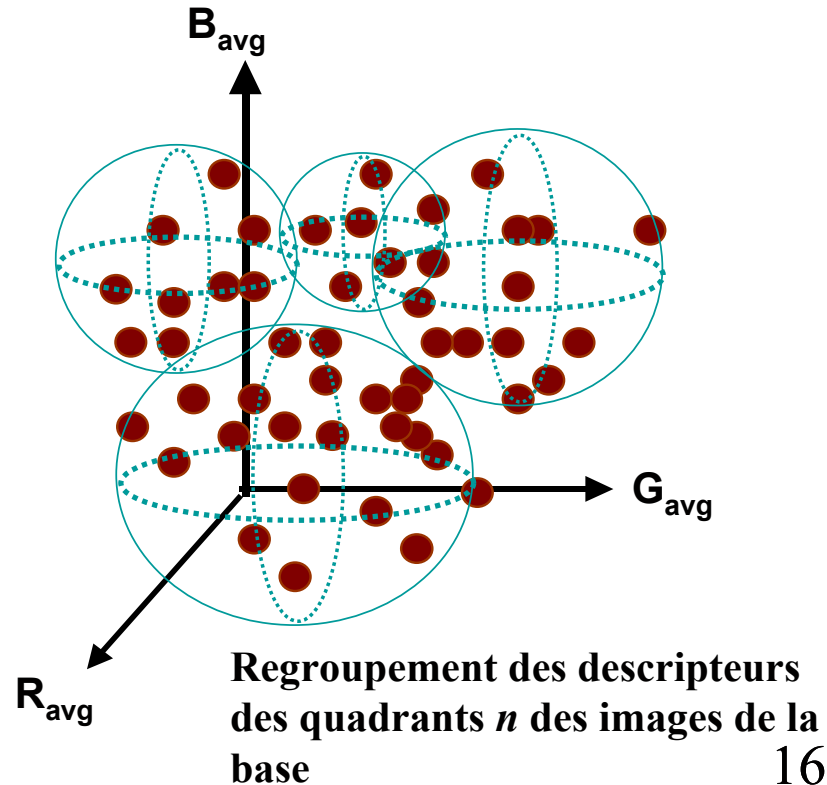


# Notre approche : le QUIP-tree (*Quadtree-based Index for Image Retrieval*)

- Représentation de tous les quadrants d'images dans un même arbre quaternaire, l'arbre QUIP
- Regroupement des quadrants homologues similaires

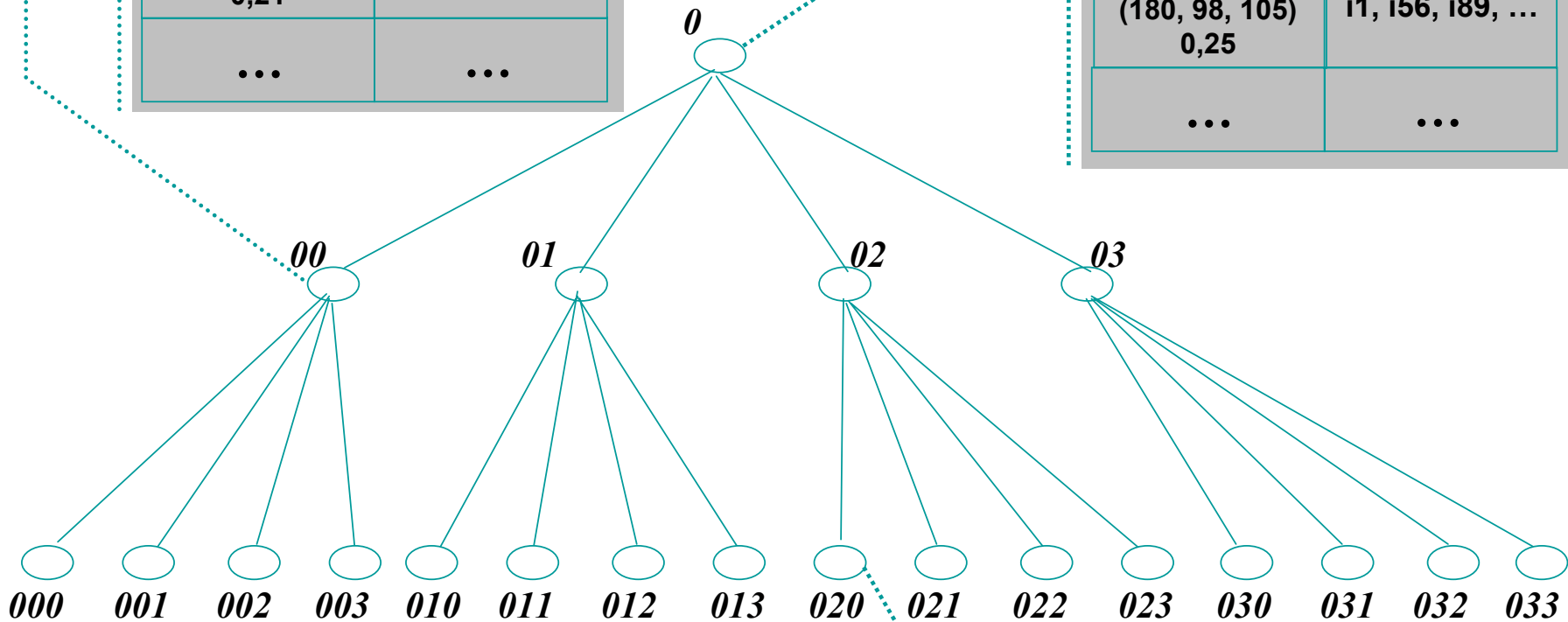
| <i>Centroïde et rayon des clusters</i> | <i>Id. d'images</i>             |
|----------------------------------------|---------------------------------|
| [ (180, 98, 105) ; 0,25 ]              | $i_1, i_{56}, i_{89}, \dots$    |
| [ (137, 128, 113) ; 0,12 ]             | $i_{23}, i_{54}, i_{78}, \dots$ |
| [ (86, 88, 84) ; 0,44 ]                | $i_3, i_5, i_{80}, \dots$       |
| ...                                    | ...                             |

Un exemple de nœud  $n$  de l'arbre QUIP



| Cluster               | Id. d'images     |
|-----------------------|------------------|
| (13, 152, 78)<br>0,21 | i4, i5, i89, ... |
| ...                   | ...              |

| Cluster                | Id. d'images      |
|------------------------|-------------------|
| (180, 98, 105)<br>0,25 | i1, i56, i89, ... |
| ...                    | ...               |



## Structure de l'arbre QUIP

| Cluster                | Id. d'images      |
|------------------------|-------------------|
| (134, 102, 78)<br>0,12 | i4, i56, i89, ... |
| ...                    | ...               |

# QUIP-tree

- Indépendant des descripteurs de quadrant
- Indépendant des distances choisies entre quadrants avec comme contrainte :
$$\Delta^{(p-1)}(i, j) \leq \Delta^{(p)}(i, j) \leq \Delta^{(p+1)}(i, j)$$
- Au niveau du prototype implémenté (par R.E. Perozo et J.M. Lanna)
  - ❑ Utilisation des 3 premiers moments statistiques sur la couleur : descripteurs à 9 dimensions
  - ❑ Arbre quaternaire de 3 niveaux : 21 quadrants/image
  - ❑ Utilisation de l'algorithme de regroupement *k-means* : 5 clusters en moyenne
  - ❑ Stockage des descripteurs dans une base de données gérée par Oracle 9i : requêtes exprimées en SQL
  - ❑ Bases de 2311 images dont certaines ont subies des traitements (distorsion, rotation, bruitage, changement de taille ...)
  - ❑ Utilisation par défaut de la distance euclidienne et de poids  $w_n=4^{-n}$

# QUIP-tree

- **Recherche globale** : recherche des images similaires à une image requête en fonction des caractéristiques globales (en partant du quadrant 0 et en utilisant un filtrage multi-niveau)
- **Recherche de sous-images** : Recherche des images ou quadrants d'images similaires à une image requête, quelle que soit la taille de l'image requête et celle des quadrants
- **Recherche locale ou partielle** : Recherche d'images ayant des régions similaires aux régions sélectionnées dans l'image requête, une région étant composée de quadrants d'image

# Exemple de requête de sous-image



Image requête



Exemple d'images de la  
base

# Búsqueda de Imagen Por 16 Regiones



Umbral Seleccionado : 1.0

Cantidad Máxima de Imágenes a Recuperar : 16



0.0  
PorRegiones2



0.00189002294979487  
PorRegiones2Rot



0.00522948499903075  
MarinaroEPequeño



0.00565587367048574  
PorRegiones0



0.0137128262578998  
PorRegiones4Rot



0.0142192360039264  
MarinaroEMediano



0.014387307413585099  
PorRegiones4



0.0201747087844533  
PorRegiones2Rot



0.0220373028789412  
MarinaroExacto



0.022095904688300298  
PorRegiones2



0.0249911506509399  
im13



0.025667200789047502  
im14



0.029805646928916203



0.0638639226406814



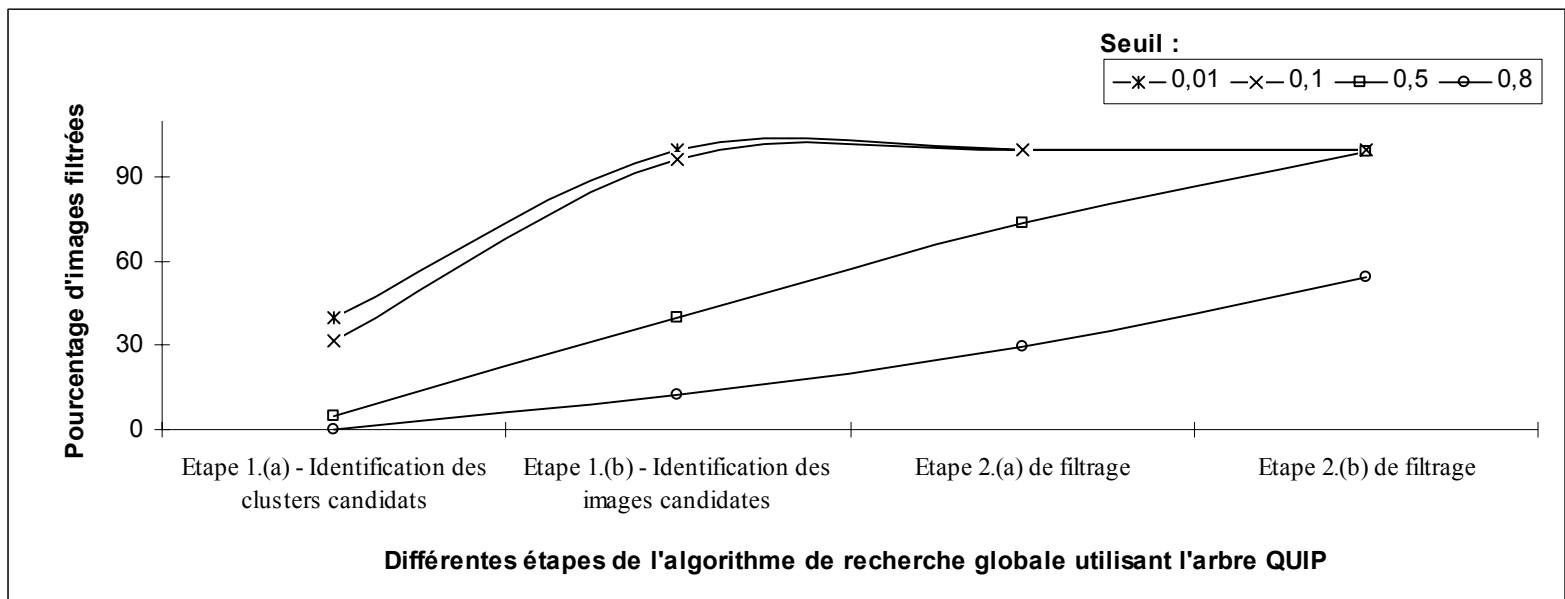
0.0660500952675006



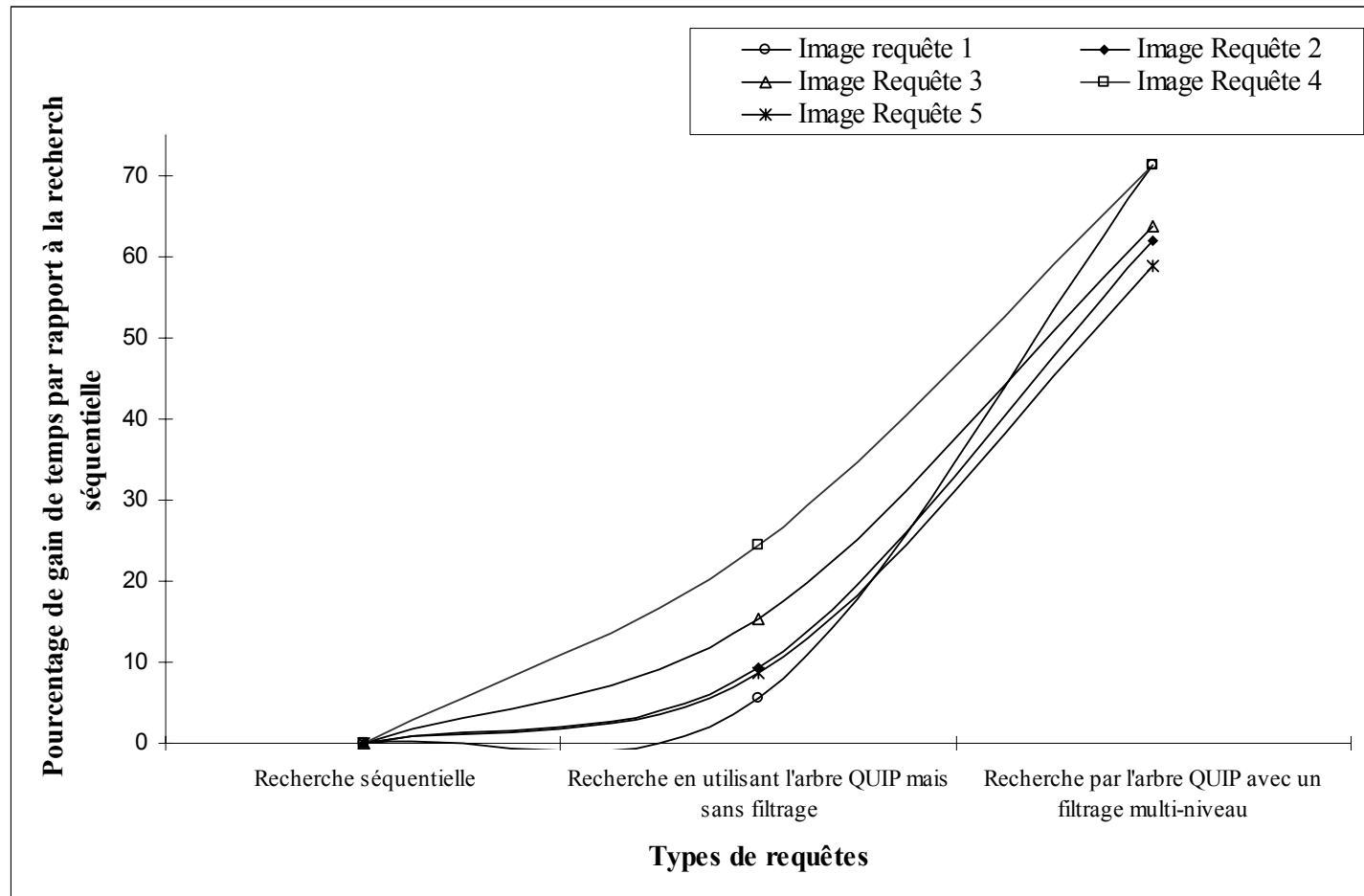
0.0660500952675006

# Efficacité du filtrage multi-niveau pour la recherche globale

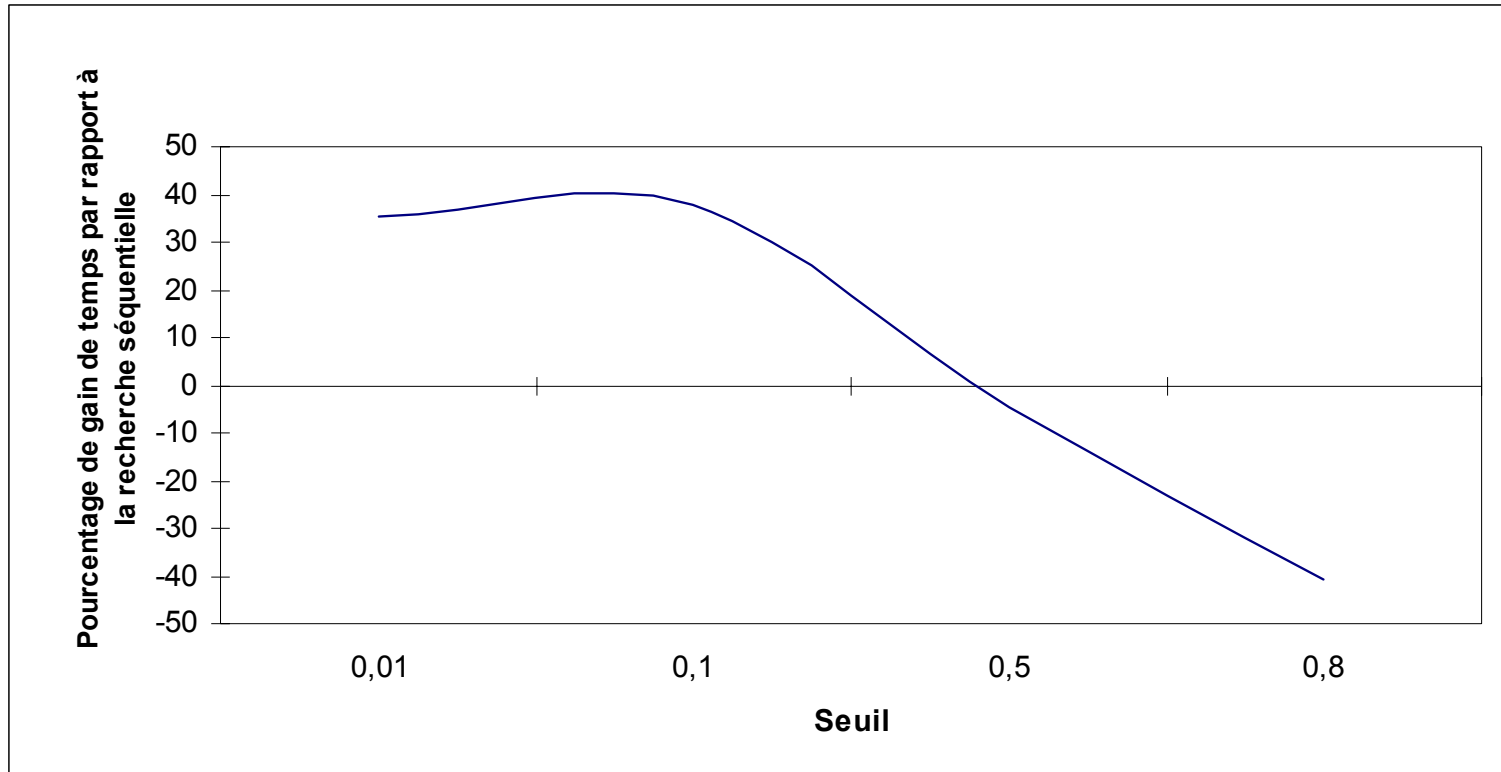
- Etape 1.(a) : Identification des clusters candidats
- Etape 1.(b) : Identification des images candidates par comparaison des quadrants 0
- Etape 2.(a) : Comparaison des quadrants 00 à 03
- Etape 2.(b) : Comparaison des quadrants de niveau 3



# Gain de temps par rapport à une recherche séquentielle pour la recherche globale



# Gain/Perte de temps par rapport à une recherche séquentielle pour la recherche globale



# Conclusion

- Proposition du QUIP-tree :
  - Structure d'index pour la recherche d'images par le contenu
  - Combinaison d'une méthode de regroupement et d'une structure spatiale, l'arbre quaternaire
  - Plusieurs types de requêtes possibles (globale, locale et sous-image)
  - Indépendance vis-à-vis des descripteurs de quadrants, du poids associés aux quadrants et des distances utilisées
- Ce qu'il reste à faire :
  - Passage à l'échelle
  - Evaluation du nombre critique de groupes par nœud
  - Utilisation d'un index de clusters
  - Comparaison avec des approches similaires

## **4. Représentation des images basées sur le stockage**

- **Compression d'images**
- **Compactage d'images : l'arbre quaternaire linéaire**
- **Chevauchement d'arbres quaternaires**
- **Arbre Quaternaire Générique**

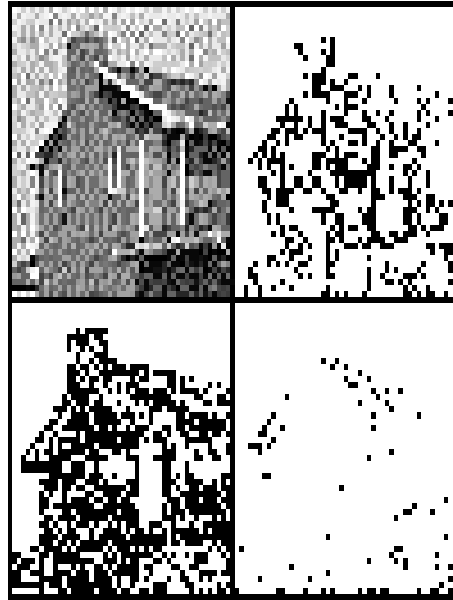
# Compression d'images

- Beaucoup d'espace occupé par les images :

*Une image de 512\*512 pixels , chaque pixel étant codé sur 24 octets, occupe 768KB*

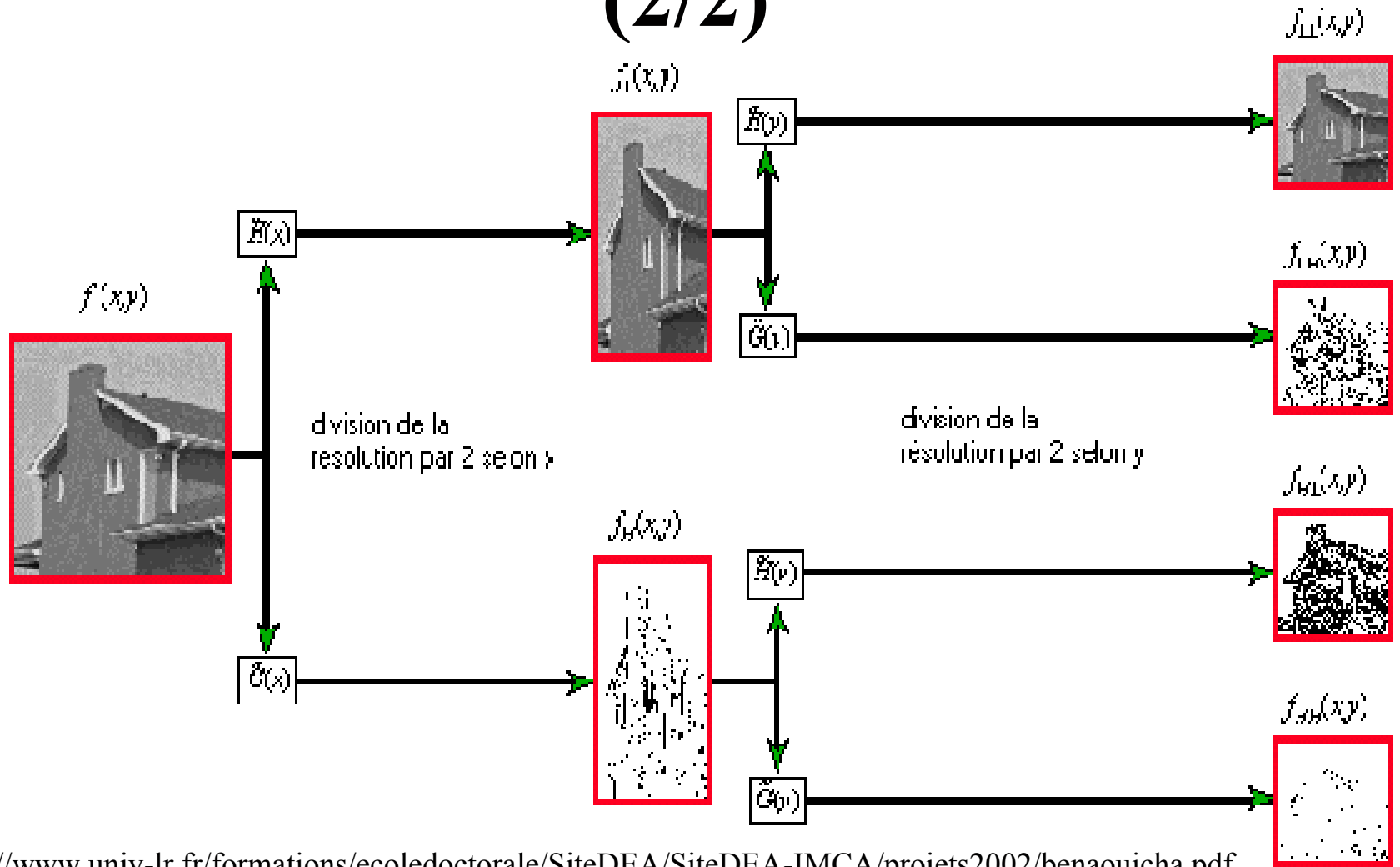
- Compression d'image : suppression de la redondance d'information ayant le moins d'impact à l'œil nu
- Plusieurs techniques de compression : JPEG (*Joint Photographic Experts Group*), ondelettes (*wavelets*) ...

# Compression en ondelettes (JPEG2000) (1/2)



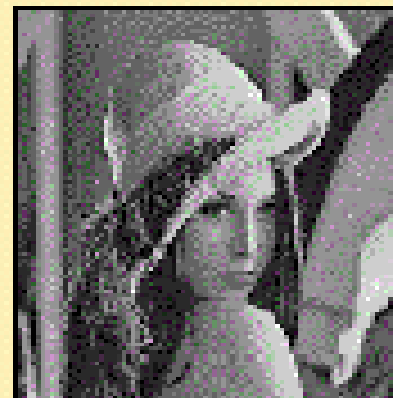
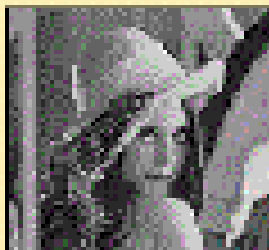
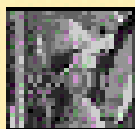
# Compression en ondelettes (JPEG2000)

## (2/2)

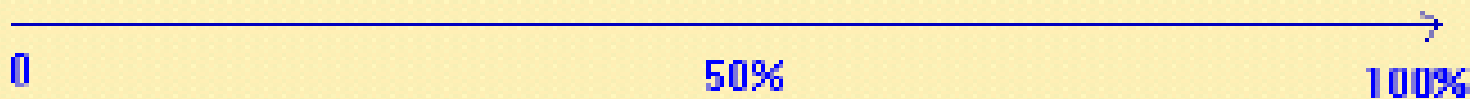


# Décompression progressive Mode 1

La taille de l'image augmente au fur et à mesure de la lecture du fichier compressé



Temps de transmission



# Décompression progressive

## Mode 2

La résolution de l'image augmente au fur et à mesure de la lecture du fichier compressé



Temps de transmission

0

50%

100%

# Compactage d'images : l'arbre quaternaire linéaire

Stocker **des images en noir et blanc** par un arbre quaternaire linéaire **en conservant la hiérarchie de l'arbre**

- Codage des nœuds noirs uniquement

- ① I. Gargantini (1982)

- ② D.J. Abel (1984)

- Codage de tous les nœuds

- ① *DF-Expression* de E. Kawagushi et T. Endo (1989)

- ② *Compact Improved Quadtree*

- de Y-H Yang, K-L Chung et Y-H Tsai (2000)

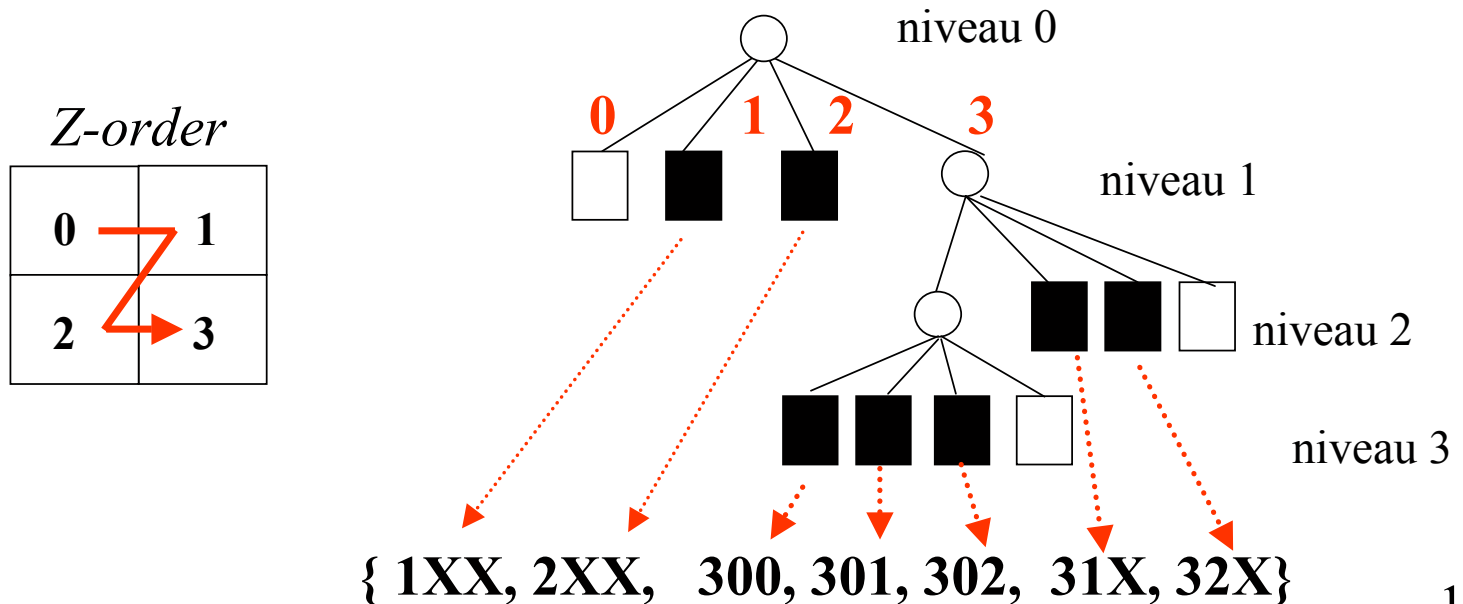
- ③ *Constant Bit-length Linear Quadtree Code*

- de T-W. Lin (1997)

# Codage des nœuds noirs

## I. Gargantini (1982)

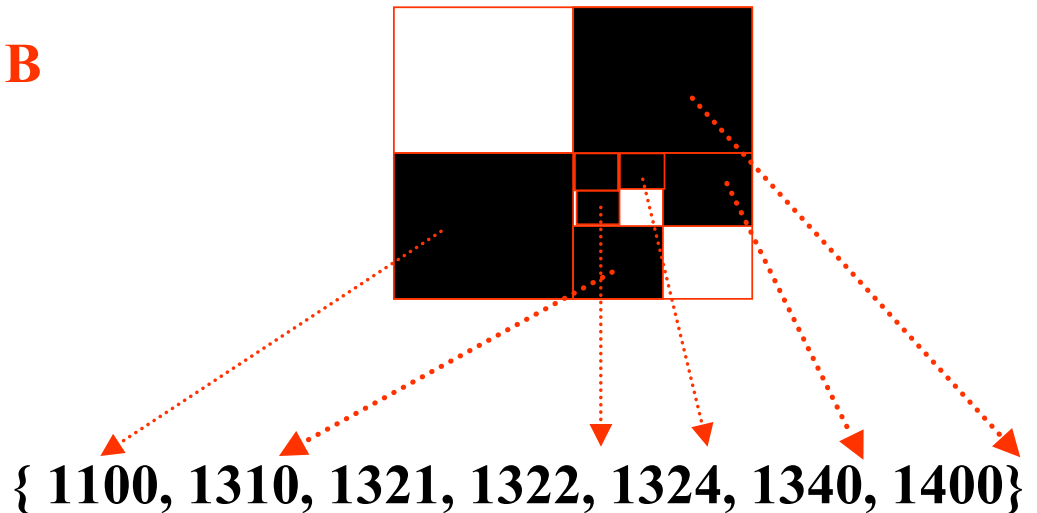
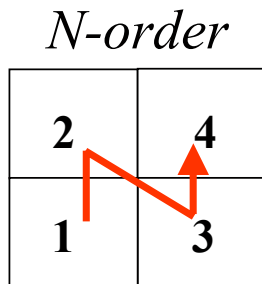
- Représentation par une **liste d'identificateurs de nœuds noirs**
- Prise en compte des directions pour générer l'identificateur d'un nœud (**ordre en Z**)
- Parcours de l'arbre quaternaire **en profondeur d'abord**
- Tous les **identificateurs** ont la **même taille** (ajout de X)



# Codage des nœuds noirs

D-J. Abel (1984)

- Représentation par une **liste d'identificateurs de nœuds noirs**
- Prise en compte des directions pour générer l'identificateur d'un nœud (**ordre en N**)
- La **racine est codée par 1** et tous les **identificateurs** ont la **même taille** (ajout de 0)
- Stockage dans un **arbre B**



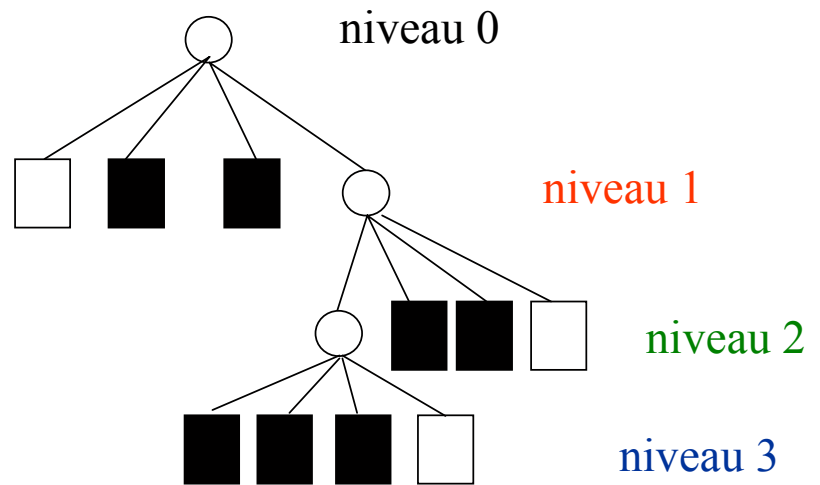
# *DF-Expression*

E. Kawagushi et T. Endo (1989)

- Codage de **tous les nœuds en profondeur d'abord**
- Codage d'un nœud noir par 1, d'un nœud blanc par 0 et d'un nœud interne par (

(011((1110110

(**0****1****1**((**1****1****1****0****1****1****0**



# Compact Improved Quadtree

de Y-H Yang, K-L Chung et Y-H Tsai (2000)

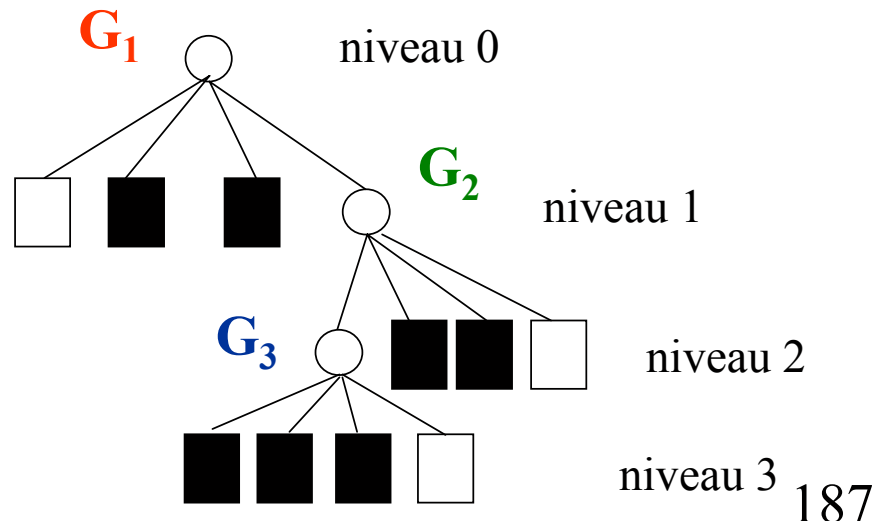
- Codage de **tous les nœuds en profondeur d'abord**
- Codage d'un nœud noir par 1, d'un nœud blanc par 0 et d'un nœud interne par 2
- $P = \langle P_1, P_2, \dots, P_n \rangle$  avec  $P_i = \sum_{j=0}^3 C_j(G_i) * 3^j$  avec  $C_j =$  codage couleur du fils situé à la  $j^{\text{ème}}$  direction du nœud gris  $G_i$

$\langle 66, 14, 13 \rangle$

$$66 = 0 * 3^0 + 1 * 3^1 + 1 * 3^2 + 2 * 3^3$$

$$14 = 2 * 3^0 + 1 * 3^1 + 1 * 3^2 + 0 * 3^3$$

$$13 = 1 * 3^0 + 1 * 3^1 + 1 * 3^2 + 0 * 3^3$$



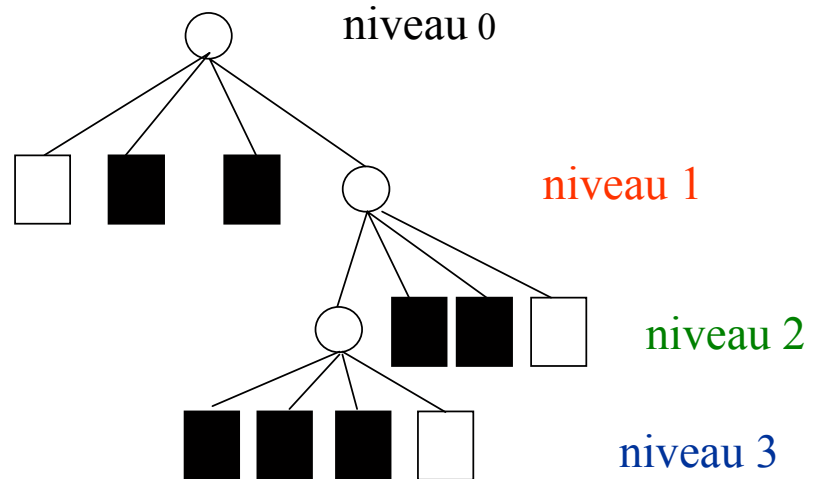
# *Constant Bit-length Linear Quadtree Code*

de T-W. Lin (1997)

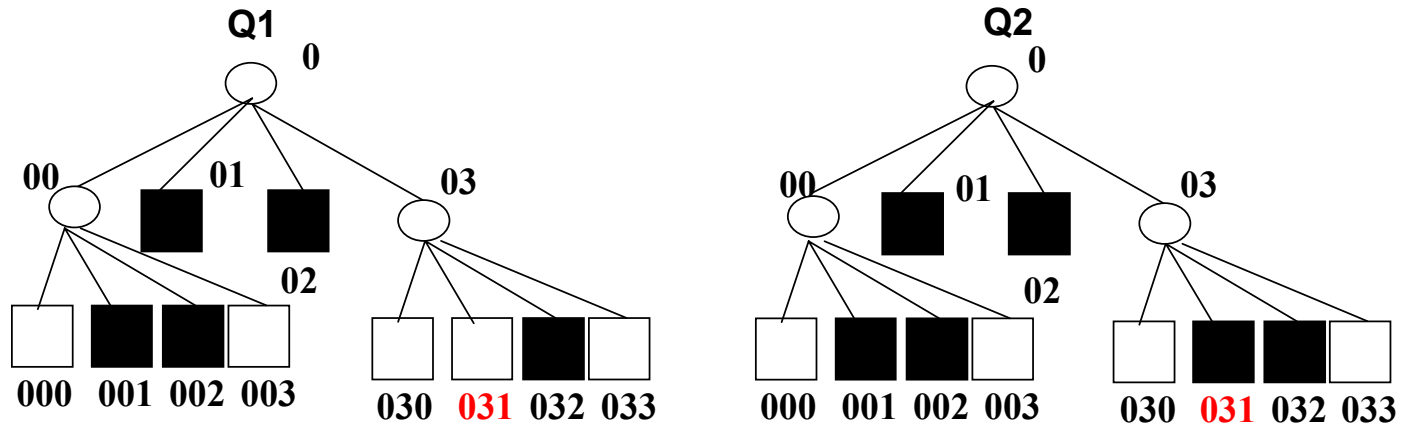
- Codage de **tous les nœuds en largeur d'abord**
- Codage d'un nœud noir par 1, d'un nœud blanc par 0 et d'un nœud interne par 2 s'il a un nœud fils interne ou par 3 sinon

0112 3110 1110

0**11**2 3**11**0 1**11**0

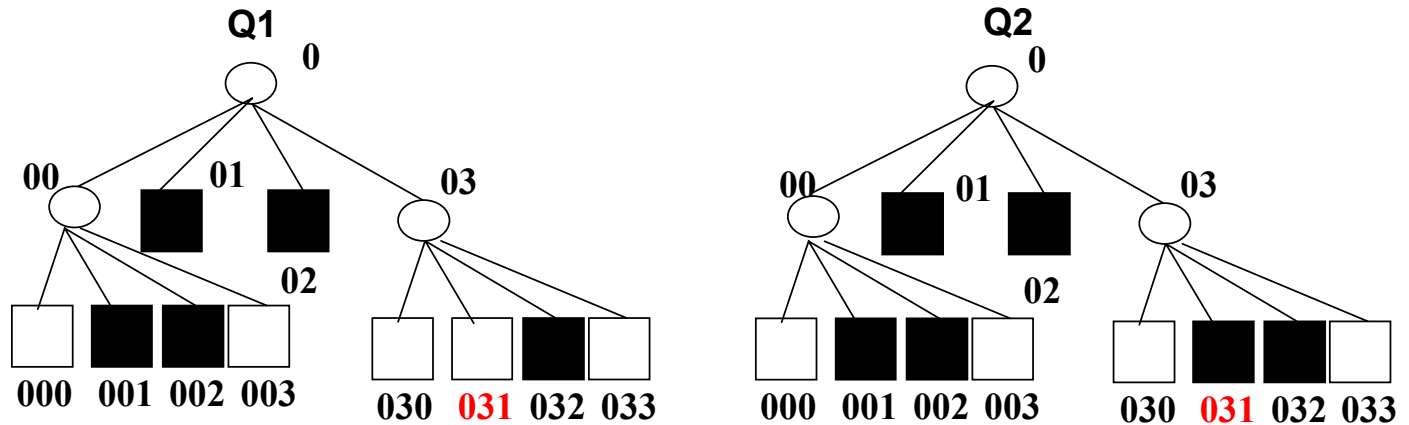


# Chevauchement d'arbres quaternaires

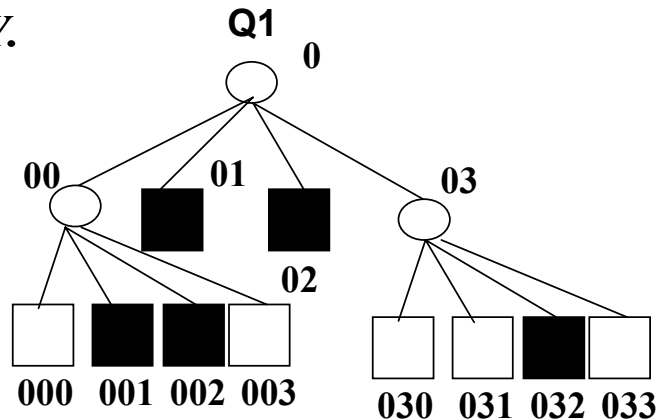


*M. Vassilakopoulous, Y.  
Manolopoulous et al.  
(1993)*

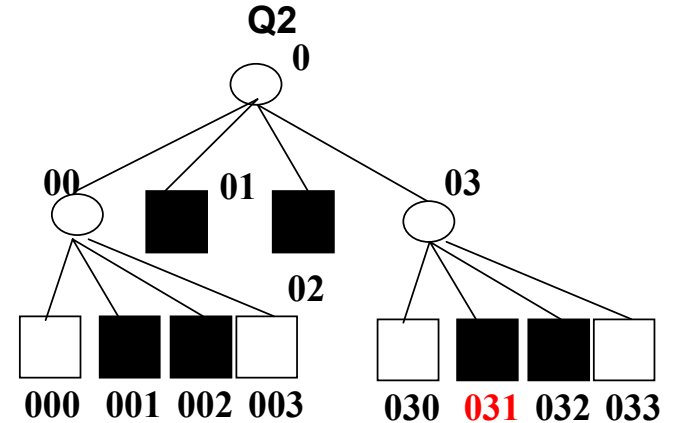
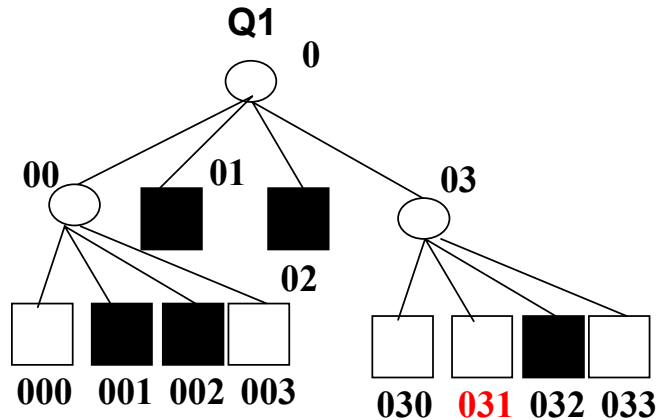
# Chevauchement d'arbres quaternaires



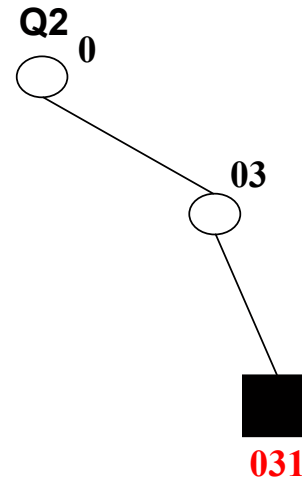
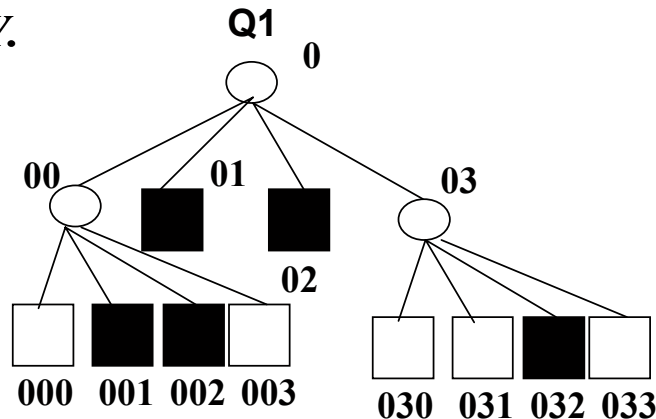
*M. Vassilakopoulos, Y.  
Manolopoulos et al.  
(1993)*



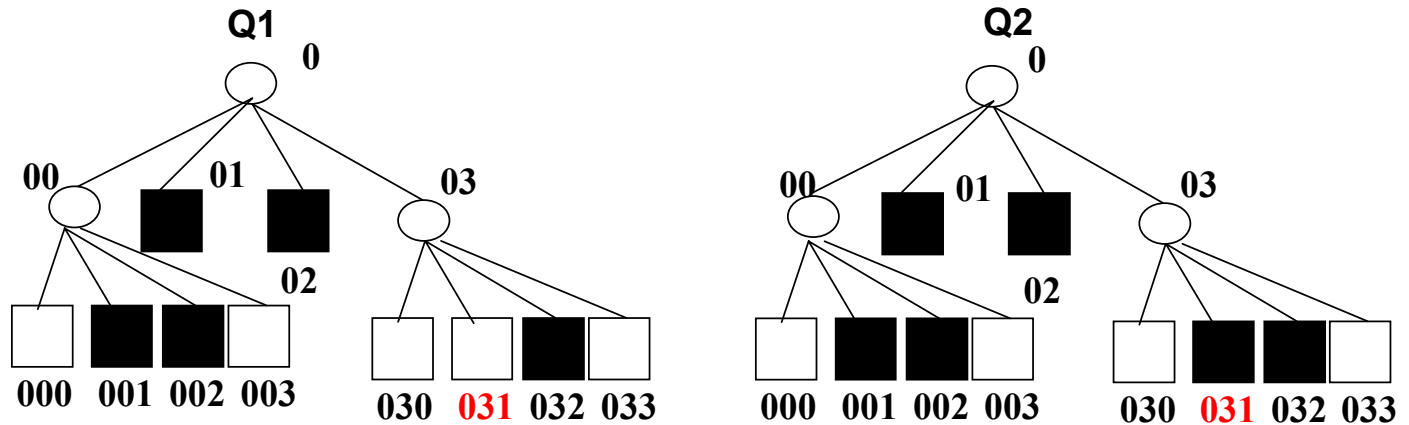
# Chevauchement d'arbres quaternaires



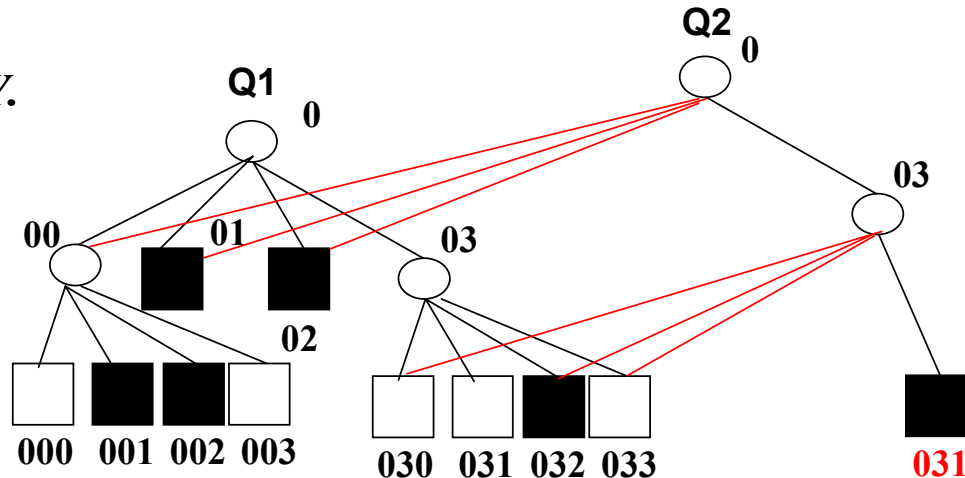
*M. Vassilakopoulous, Y.  
Manolopoulous et al.  
(1993)*



# Chevauchement d'arbres quaternaires



*M. Vassilakopoulous, Y.  
Manolopoulous et al.  
(1993)*



# Chevauchement d'arbres quaternaires linéaires

Code CBLQ chevauché ou *Overlapped Content Bit lenght Linear Quadtree* T-W. Lin (1997)

- Codage de la **première** et de la **dernière image** de la séquence par le **code CBLQ**
- Codage de toutes les images, sauf la première, par le **code CBLQ chevauché**

0110

AAAB ADDA

AAAB DAAD

0111

Image 1

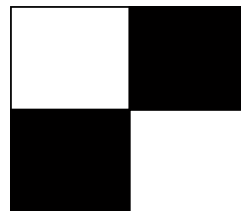


Image 2

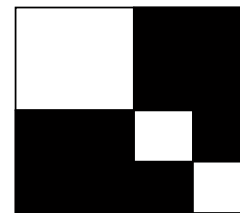
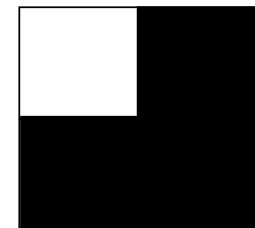


Image 3



# Chevauchement d'arbres quaternaires linéaires

|                           |          |          |          |          |          |          |
|---------------------------|----------|----------|----------|----------|----------|----------|
| Quadrant de la 1ère image | B        | B        | G        | G        | N        | N        |
| Quadrant de la 2ème image | G        | N        | N        | B        | B        | G        |
| Code chevauché            | <b>C</b> | <b>D</b> | <b>C</b> | <b>D</b> | <b>C</b> | <b>D</b> |

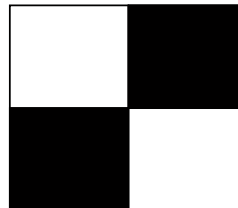
**0110**

**AAAB ADDA**

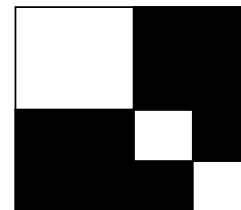
**AAAB DAAD**

**0111**

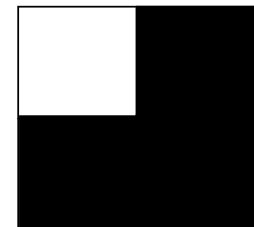
**Image 1**



**Image 2**



**Image 3**



# **Arbre Quaternaire Générique**

[JMR99, JMR00, Man00]

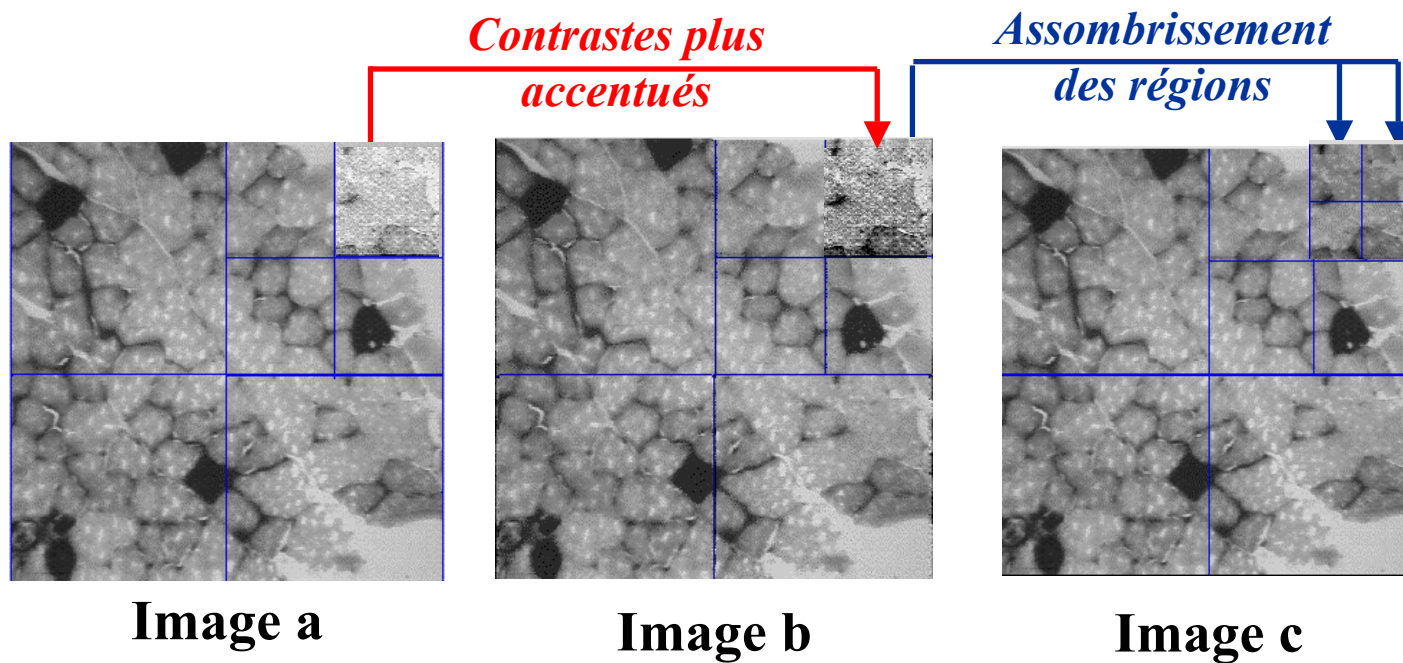
- **Contexte**
- **Arbre Quaternaire Générique**
- **Comparatif avec les travaux proches**

# Contexte (1/2)

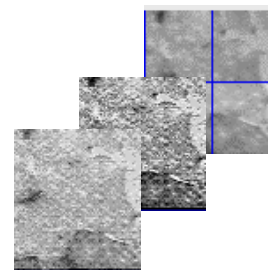
## Un système de traitement d'images :

- l'application d'une série d'opérations particulières sur une image ou sur des parties d'images,
- la gestion, la manipulation et l'accès des images résultant de traitements,
- le retour à une image précédente lorsque le traitement a trop modifié l'image originale,
- le suivi de l'évolution d'une région d'image après différents traitements ...

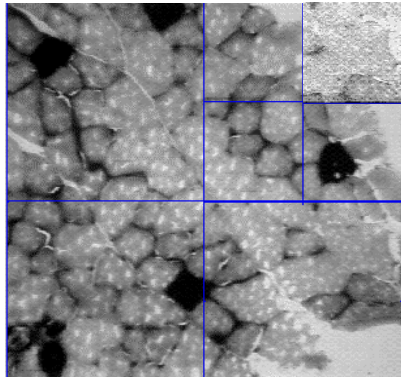
# Contexte (2/2)



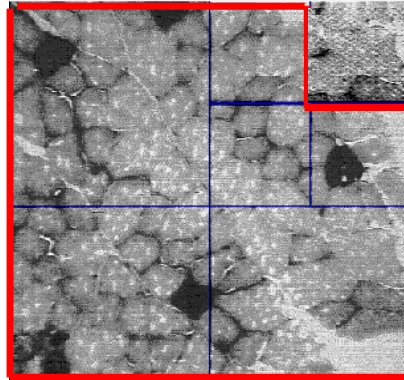
*Résultats des traitements  
appliqués à la même région dans  
les images a, b et c :*



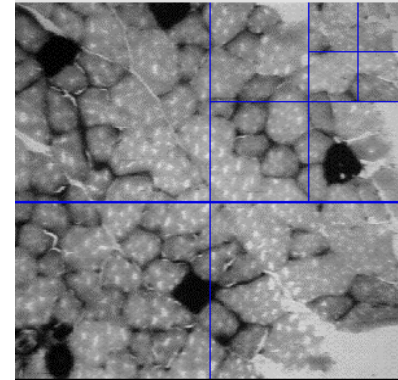
# Partage entre images



**Image a**



**Image b**



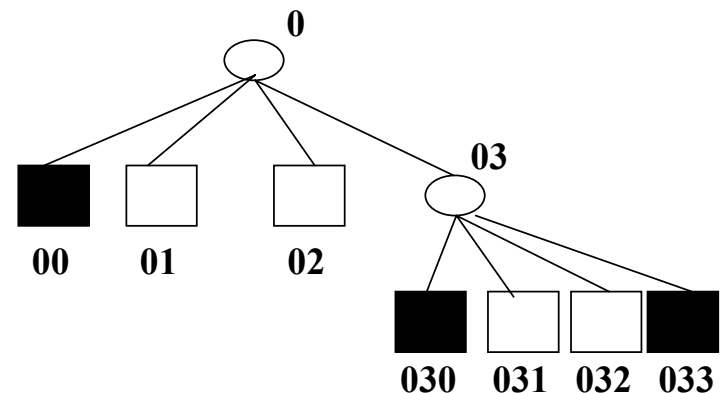
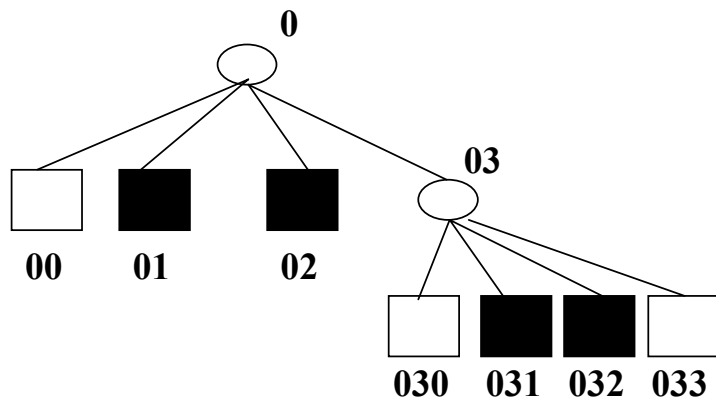
**Image c**

- **Stockage des différences**
- **Partage des régions communes**

# Opérations sur les arbres quaternaires

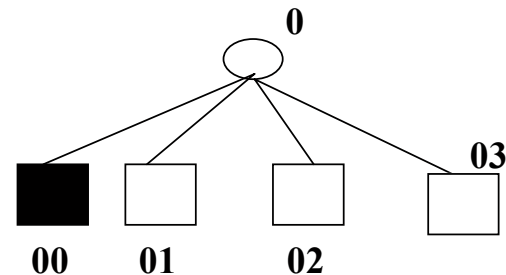
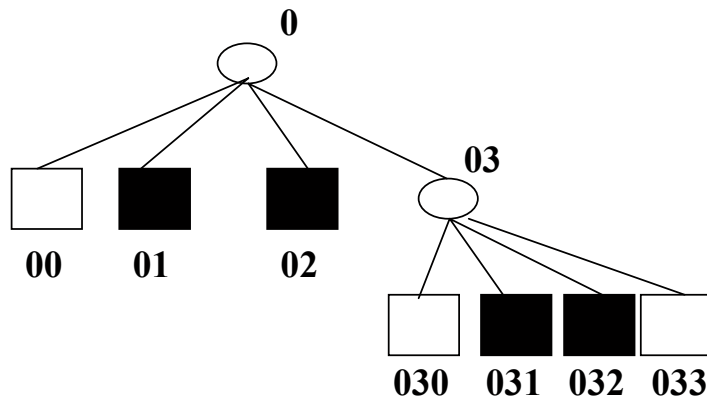
## ① Opération sur un arbre

Ex. complémentaire



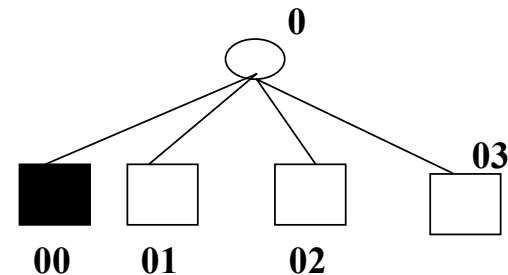
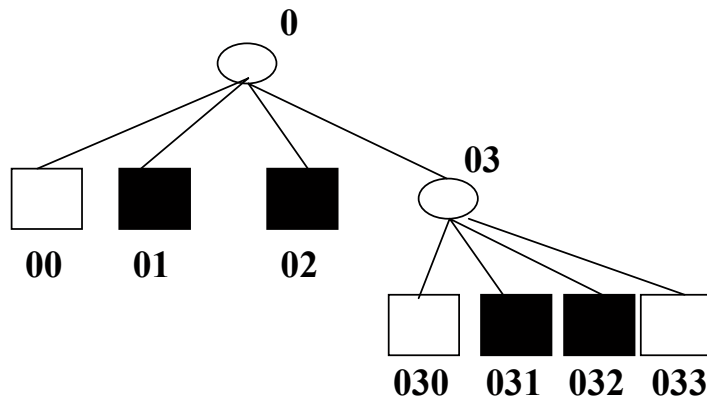
## ② Opération sur deux arbres

**Nœuds homologues** : nœuds de même identificateur dans différents arbres quaternaires



## ② Opération sur deux arbres

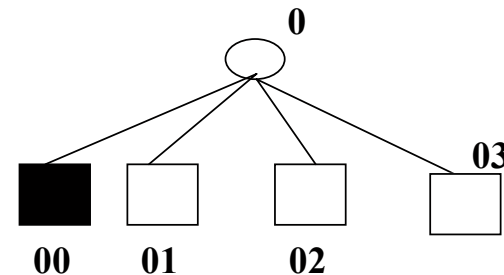
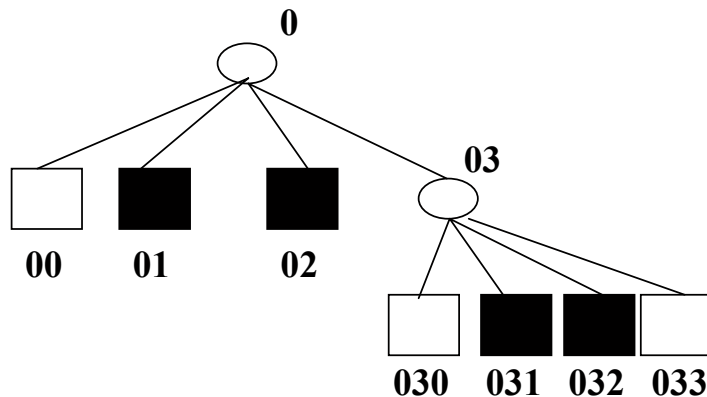
**Nœuds homologues** : nœuds de même identificateur dans différents arbres quaternaires



Les nœuds  $03x$  ont pour valeur  $\perp$  (*n'existe pas*)

## ② Opération sur deux arbres

**Nœuds homologues** : nœuds de même identificateur dans différents arbres quaternaires

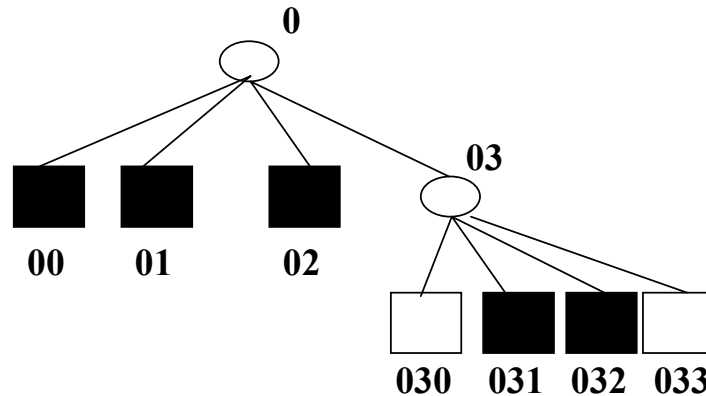
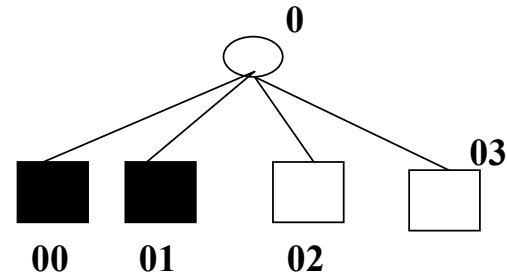
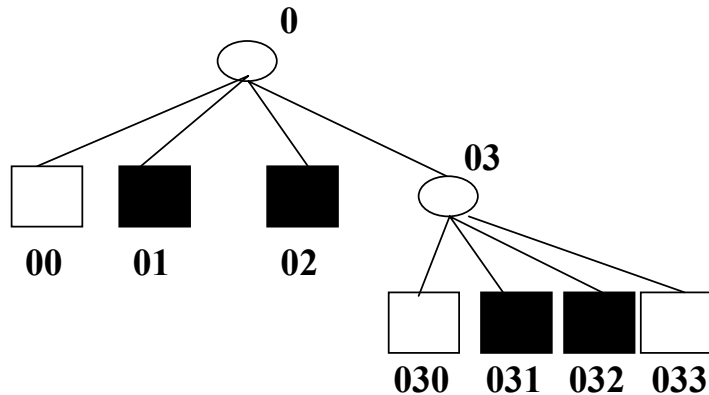


Les nœuds  $03x$  ont pour valeur  $\perp$  (*n'existe pas*)

**Opérations élémentaires** : union, intersection, différence, comparaison

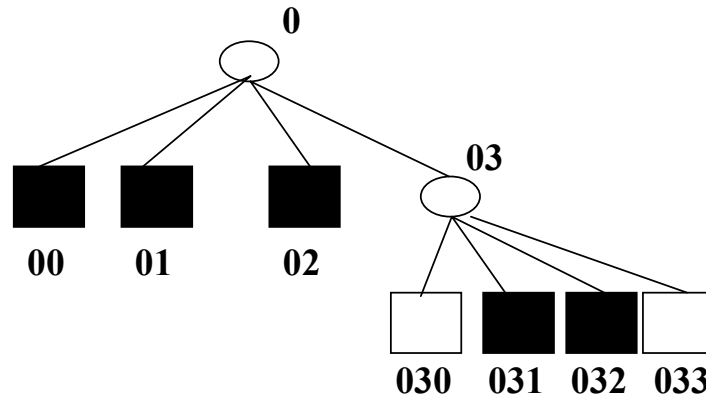
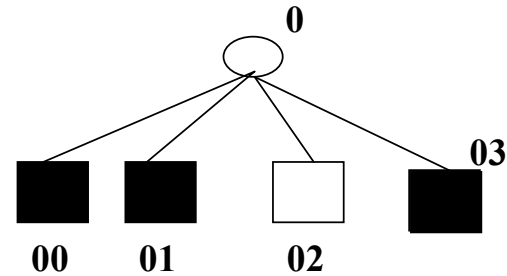
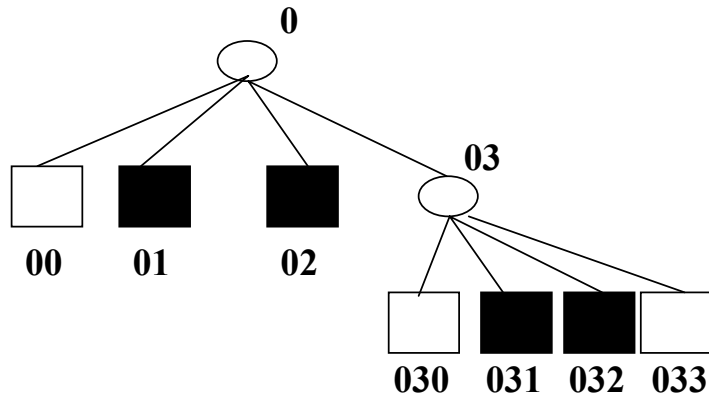
## ② Opération sur deux arbres

**Union :**



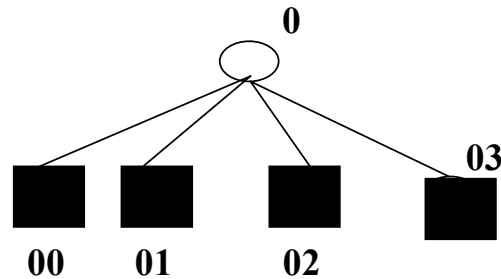
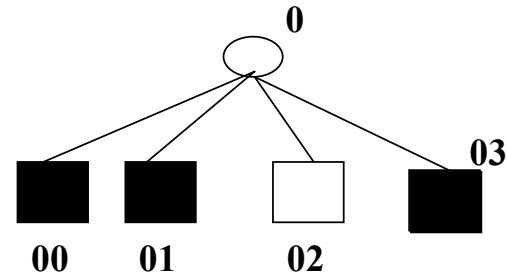
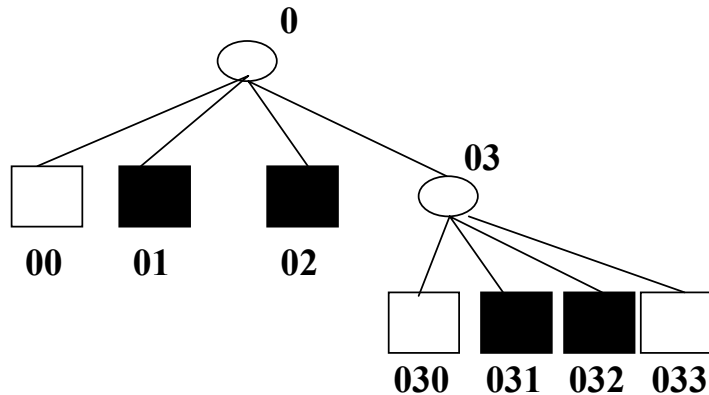
## ② Opération sur deux arbres

**Union :**

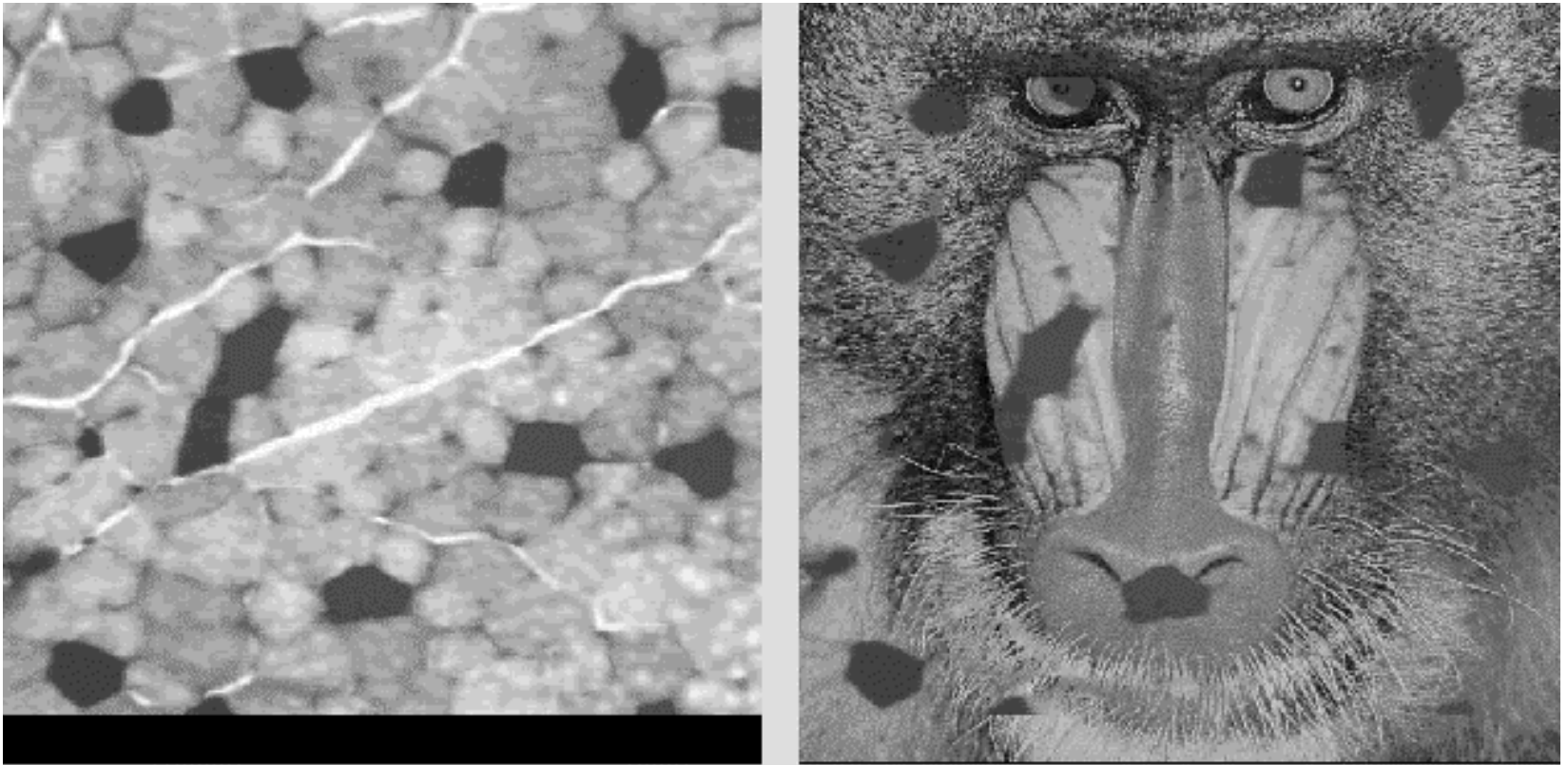


## ② Opération sur deux arbres

**Union :**

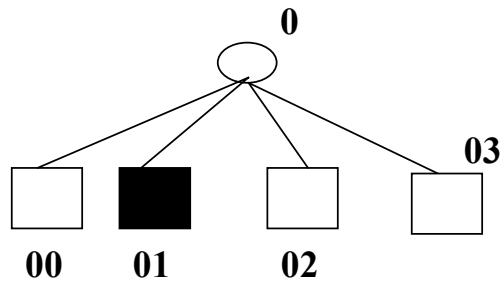
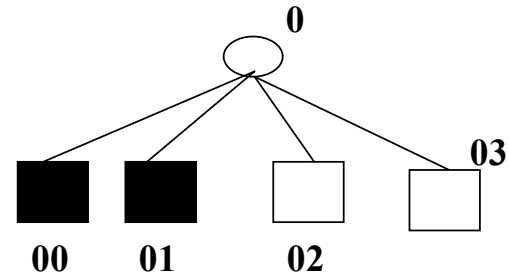
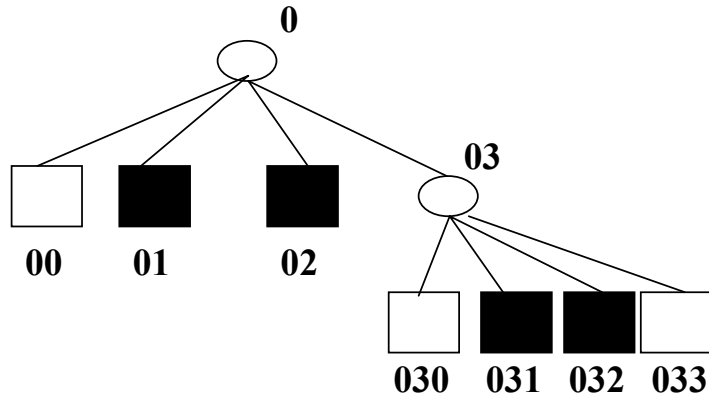


# Exemple d'union de deux images



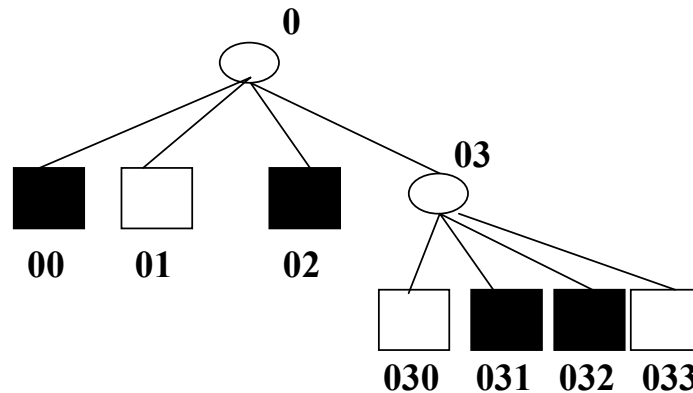
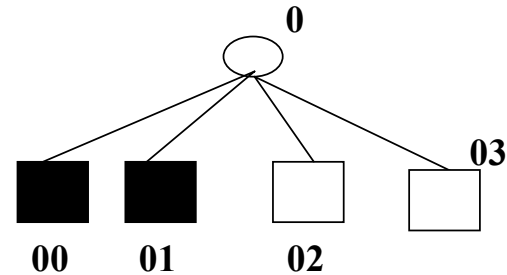
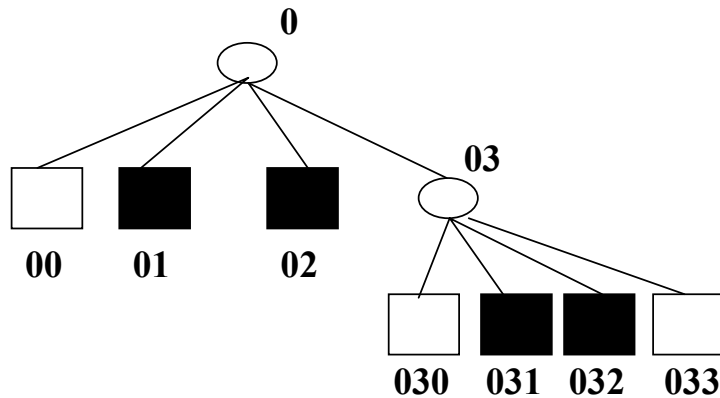
## ② Opération sur deux arbres

### Intersection :



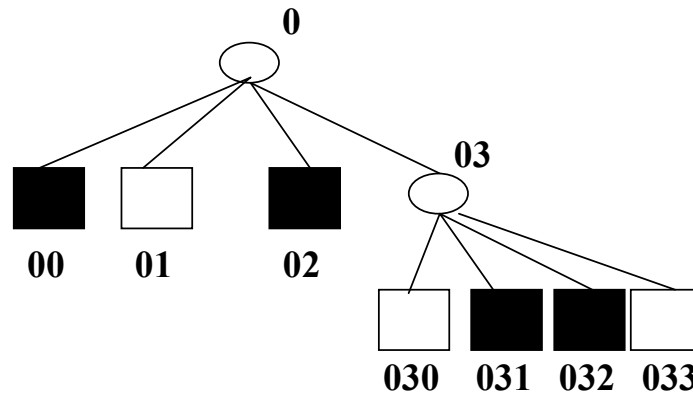
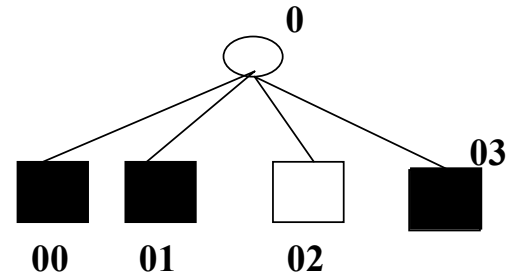
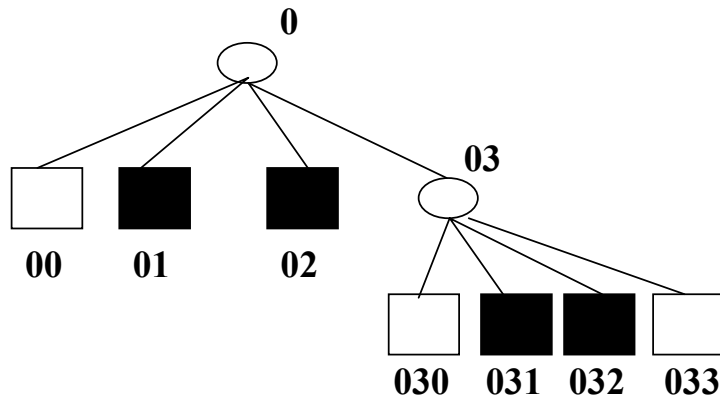
## ② Opération sur deux arbres

**Différence :**



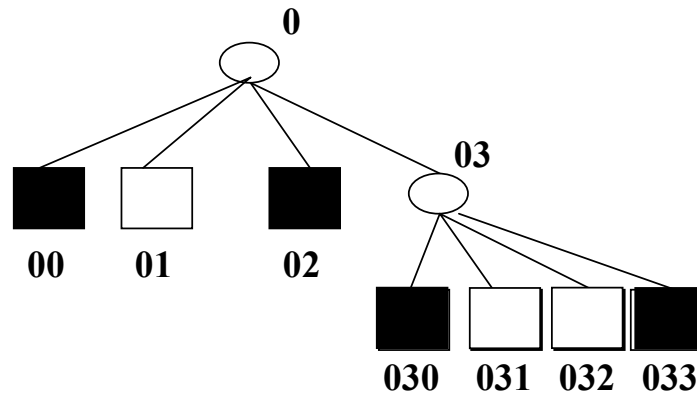
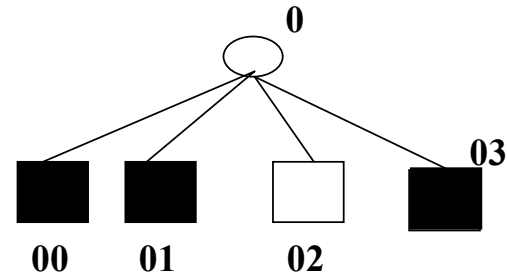
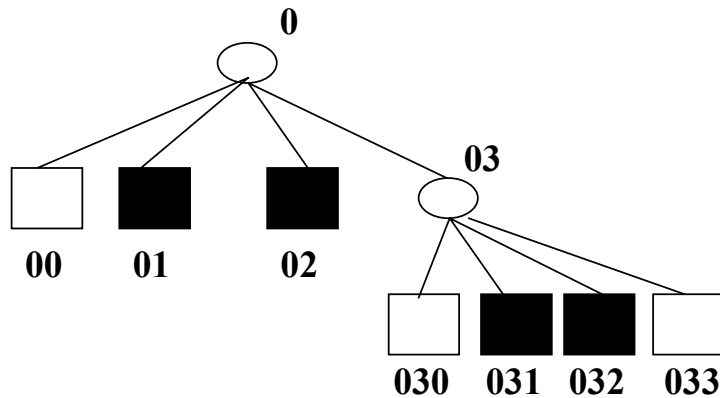
## ② Opération sur deux arbres

**Différence :**



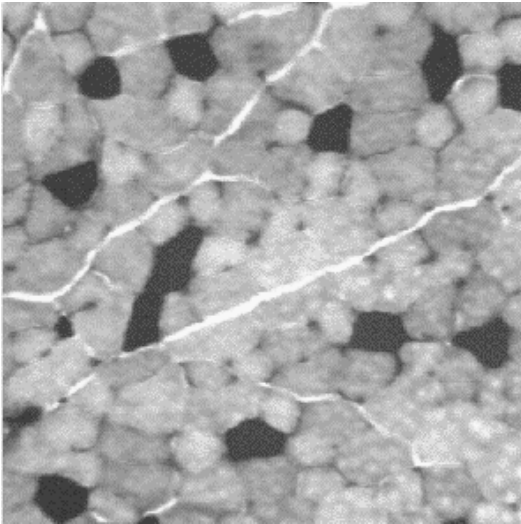
## ② Opération sur deux arbres

**Différence :**

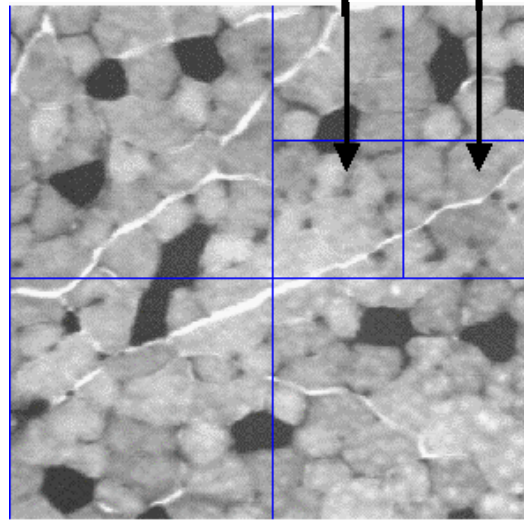


# Exemple de différence entre deux images

Dilatation des  
quadrants 02 et 03



**Image initiale**



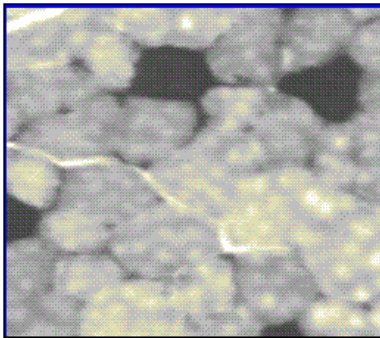
**Image après traitement**



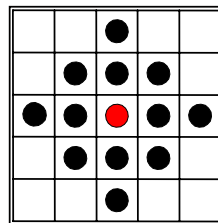
**Différence entre  
les deux images**

# Exemple d'érosion

$$(A \ominus B) = \left[ \bigcup_{b \in -B} \{c / c = a+b, a \in A^c\} \right]^c$$

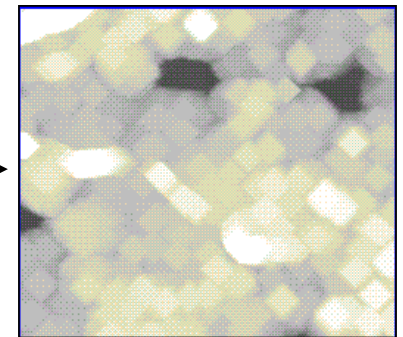


**Image A**



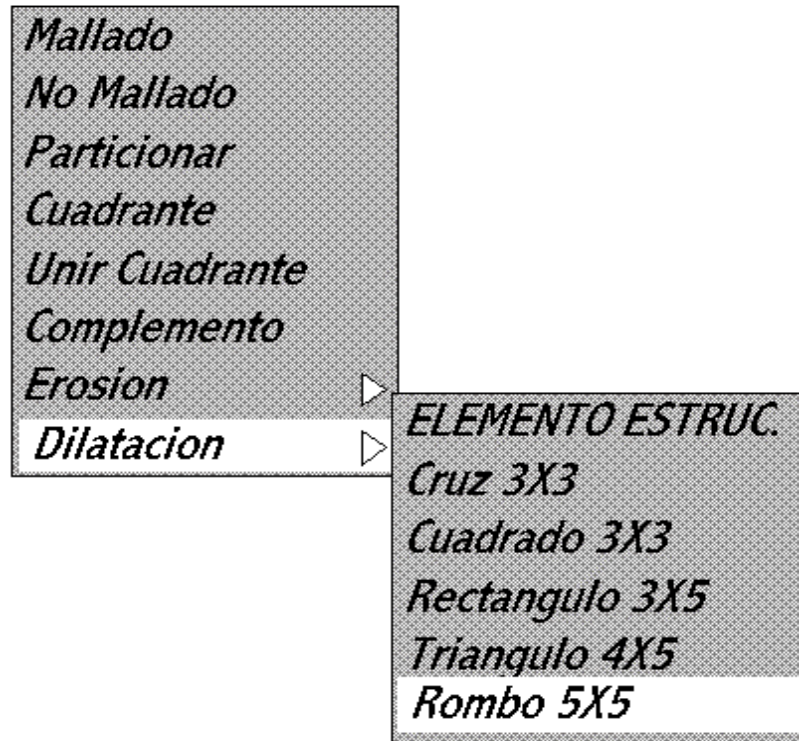
**Patron B**

$(A \ominus B)$



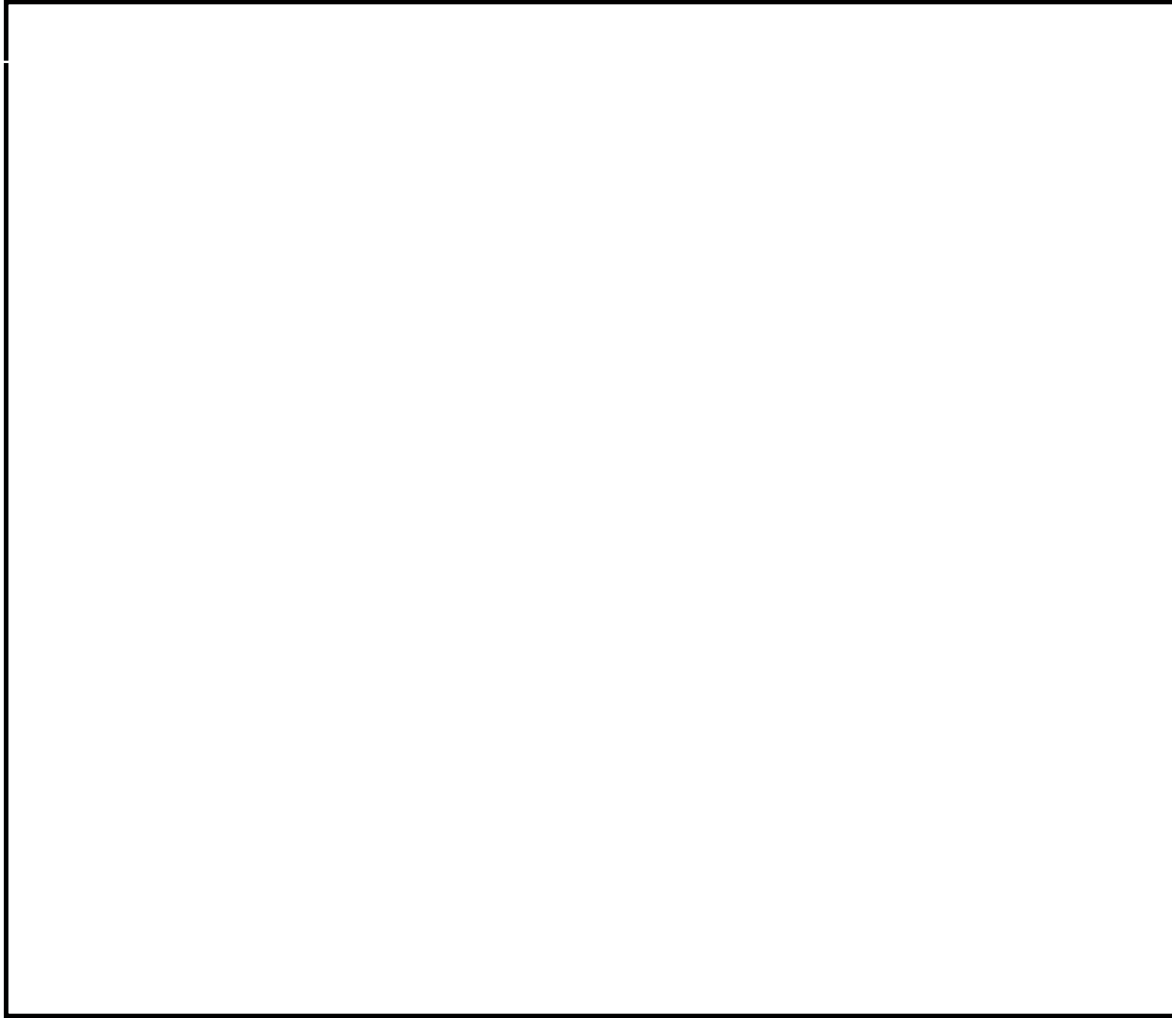
**Image A  
après érosion**

# Erosion et dilatation



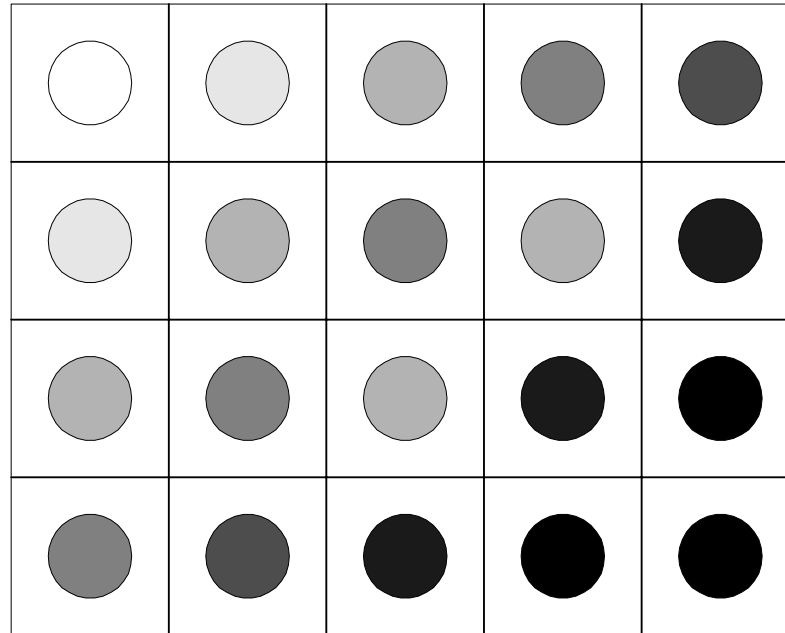


# Erosion



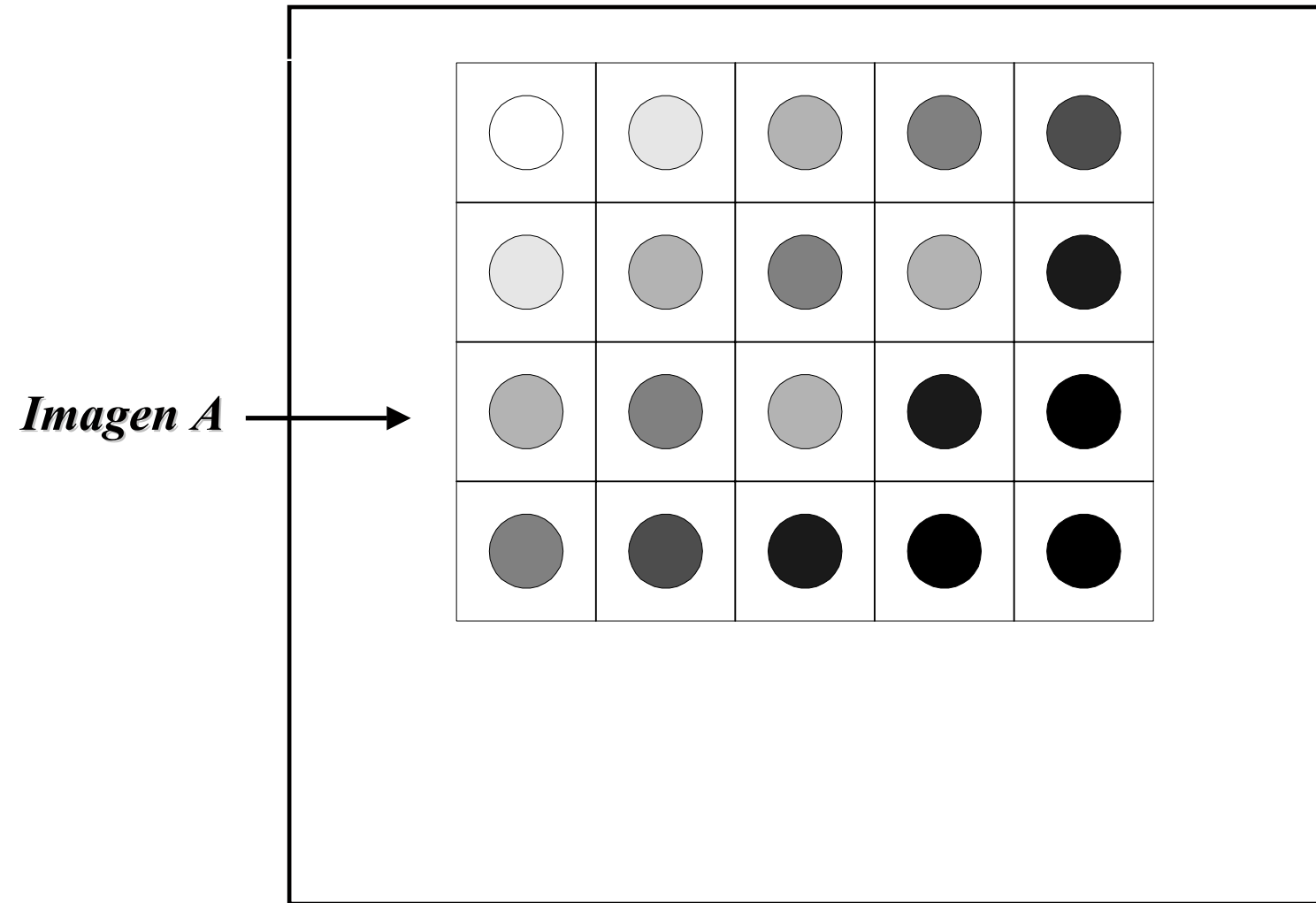


# Erosion



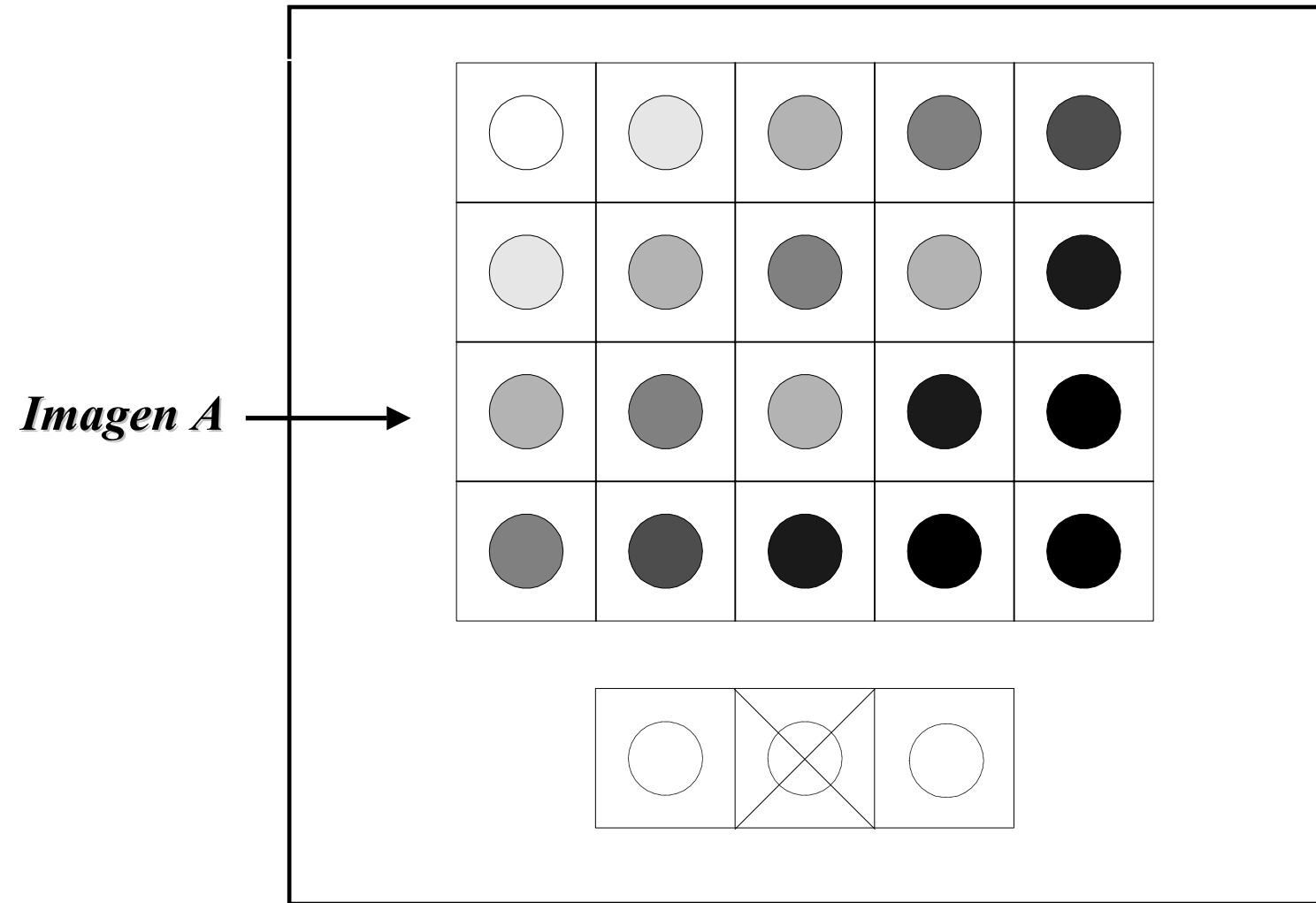


# Erosion



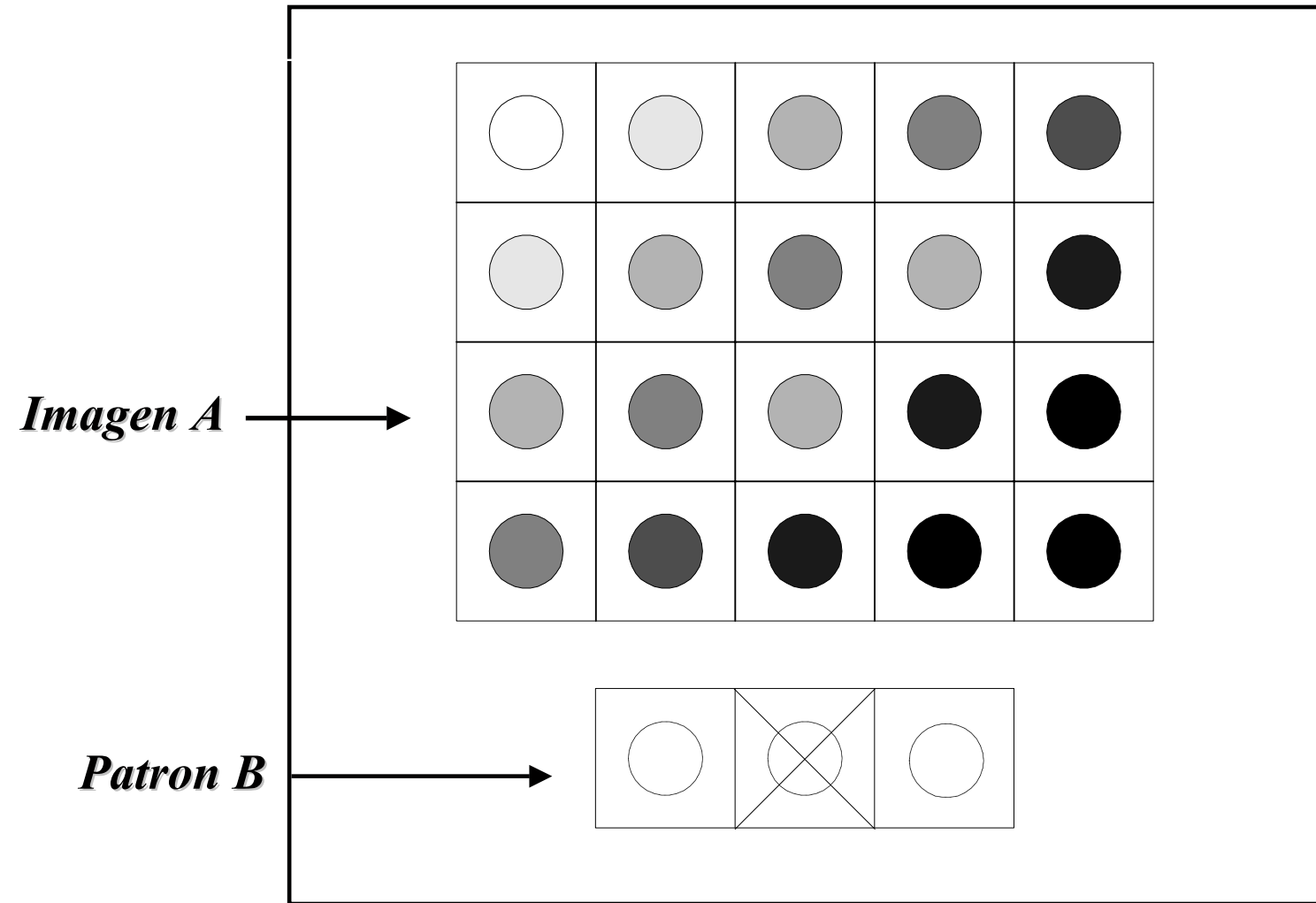


# Erosion



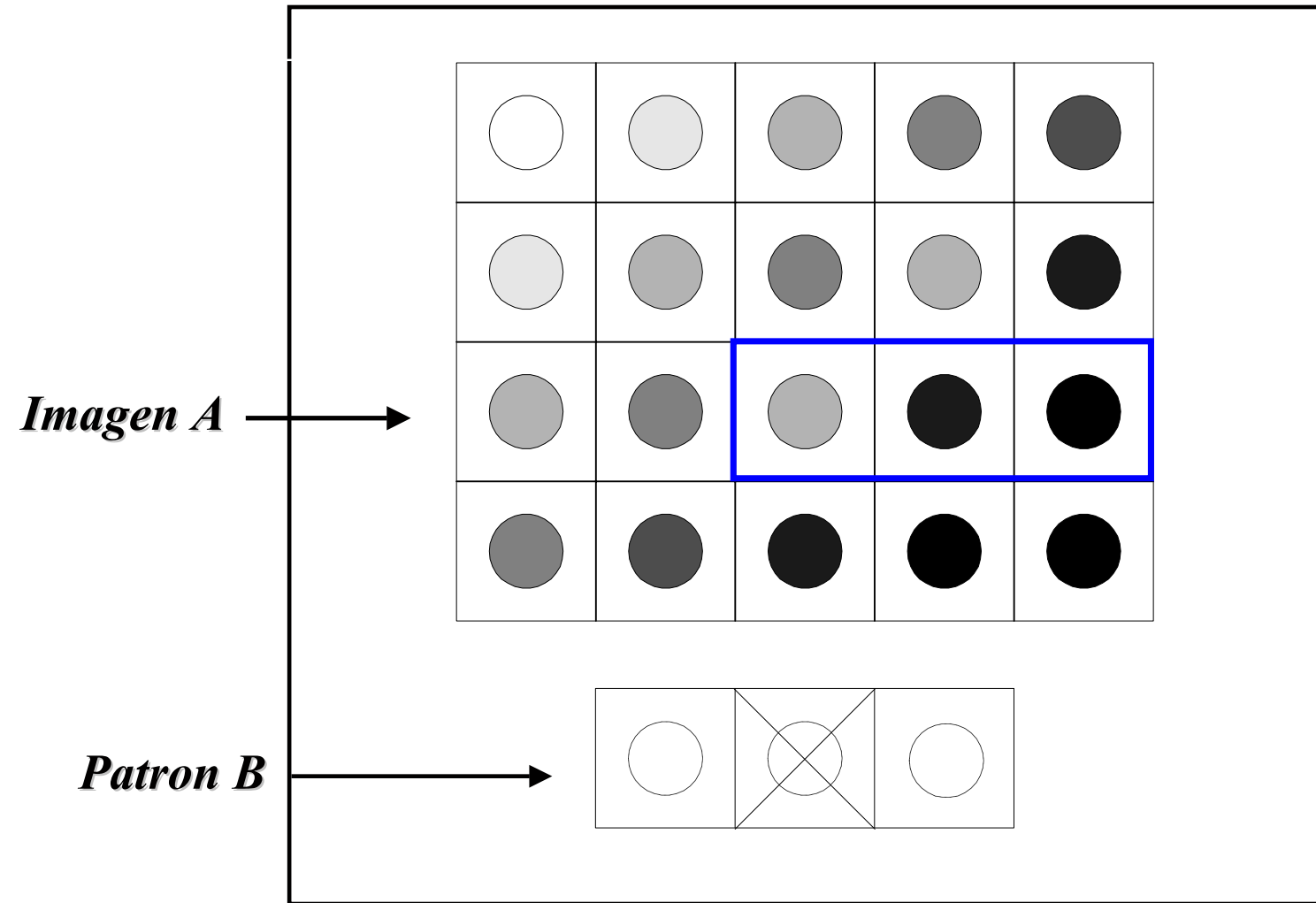


# Erosion



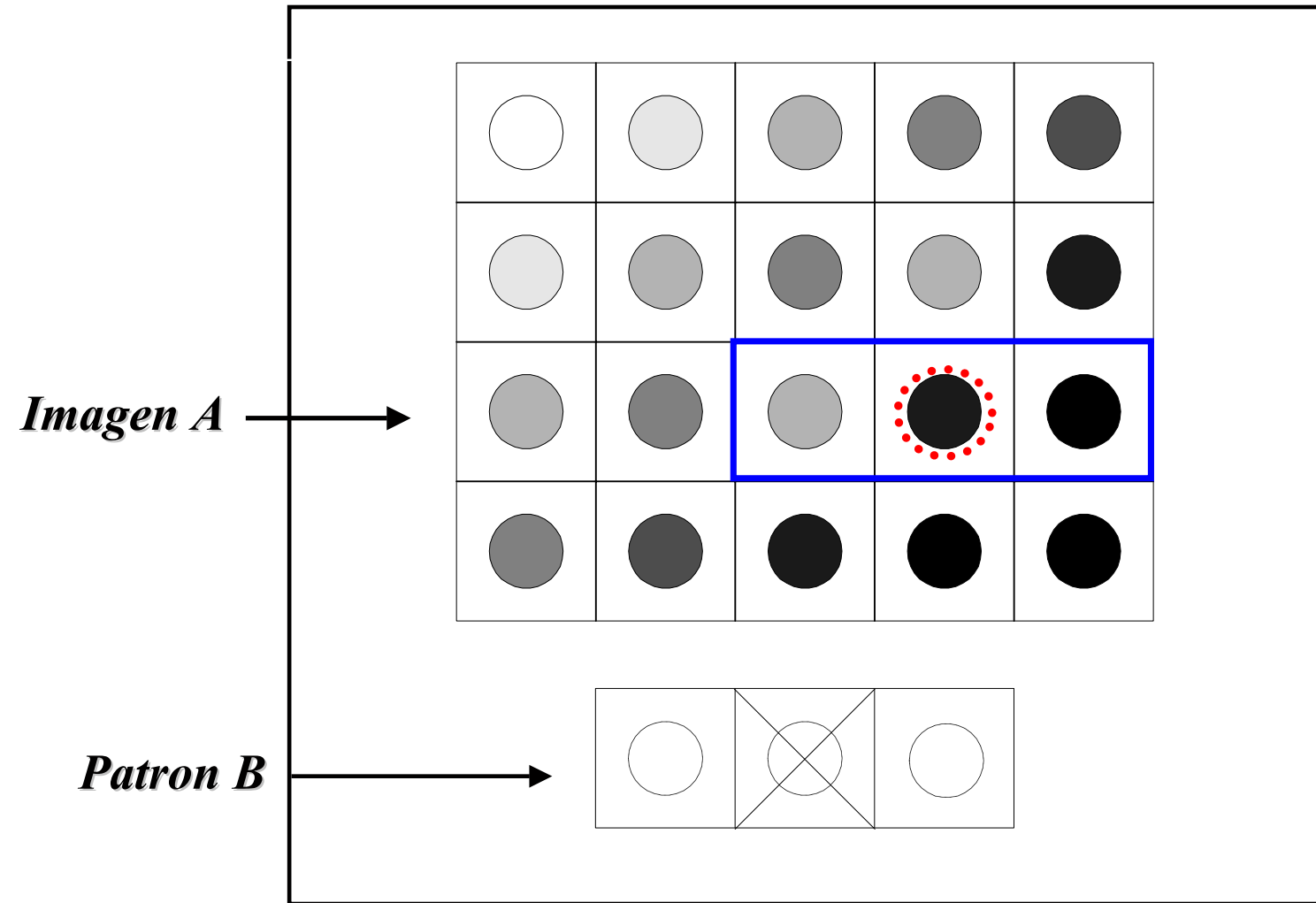


# Erosion



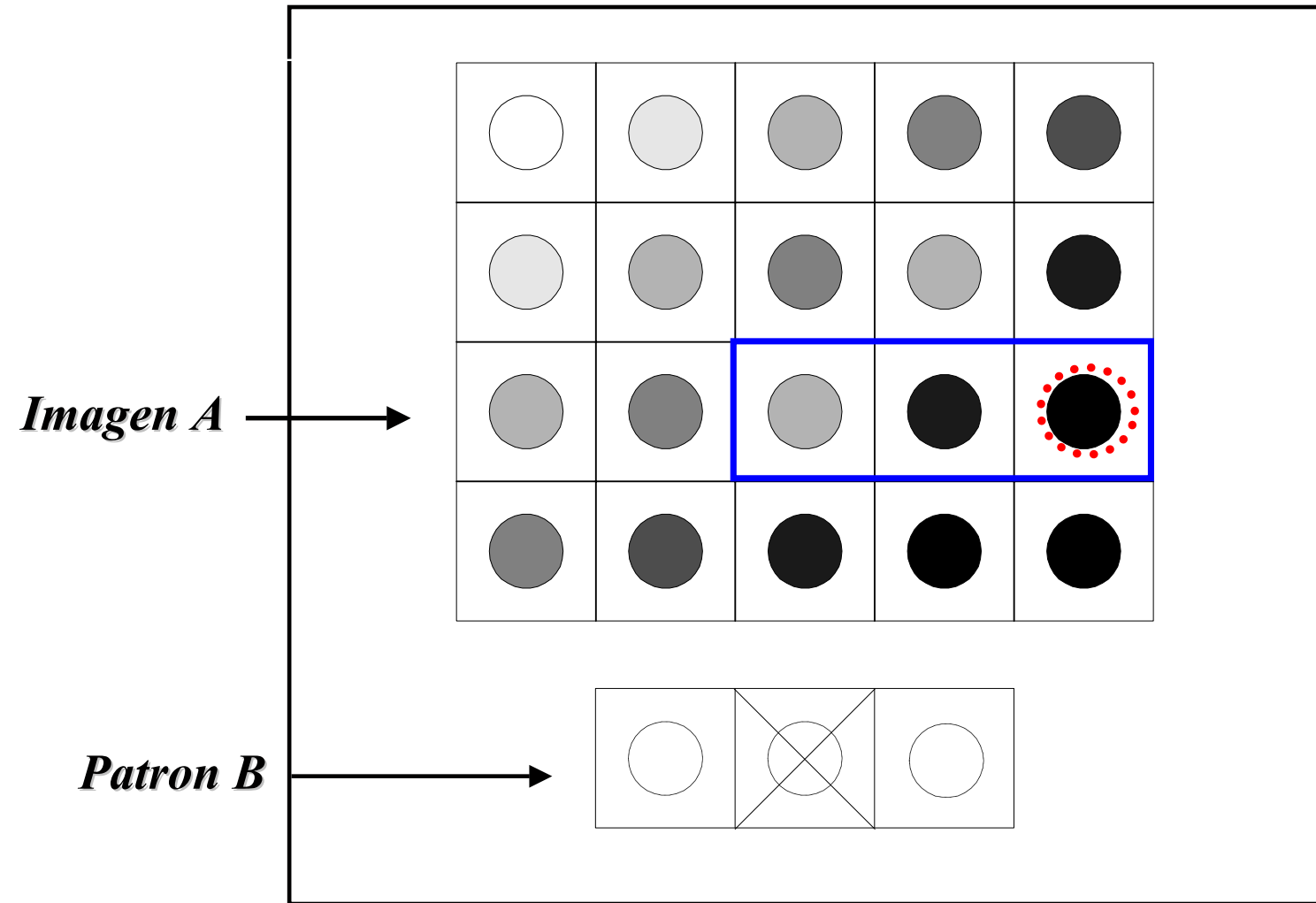


# Erosion



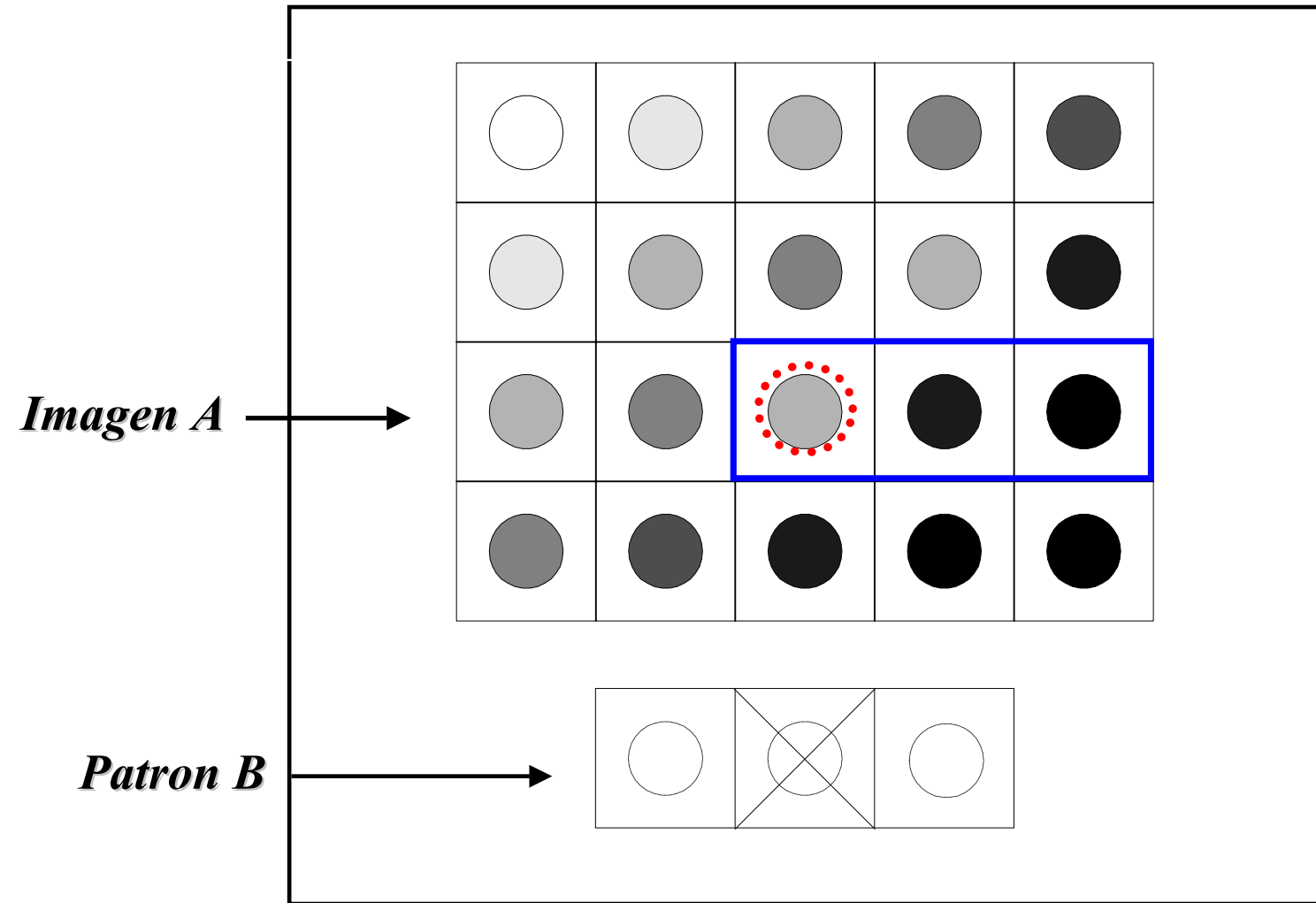


# Erosion



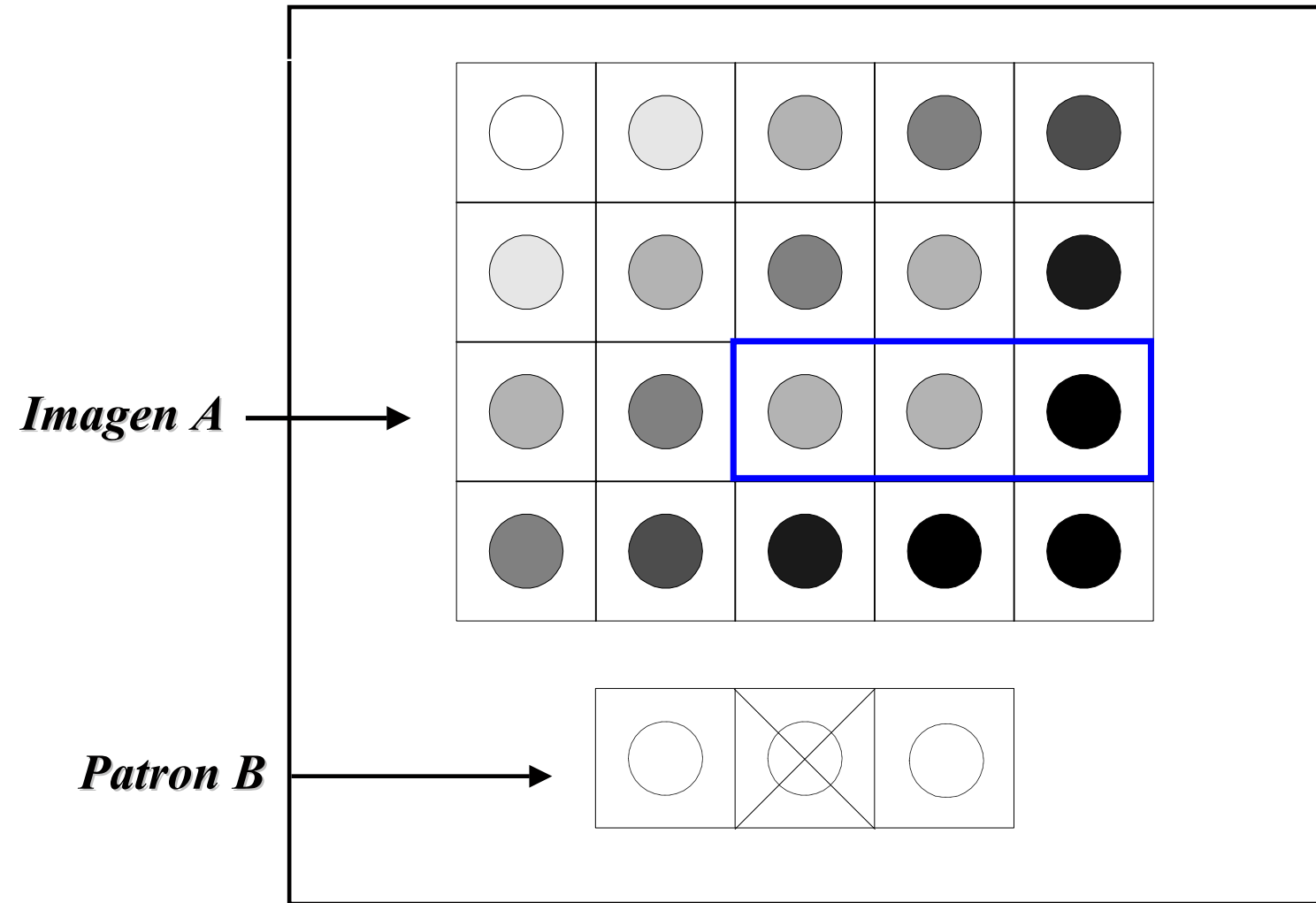


# Erosion



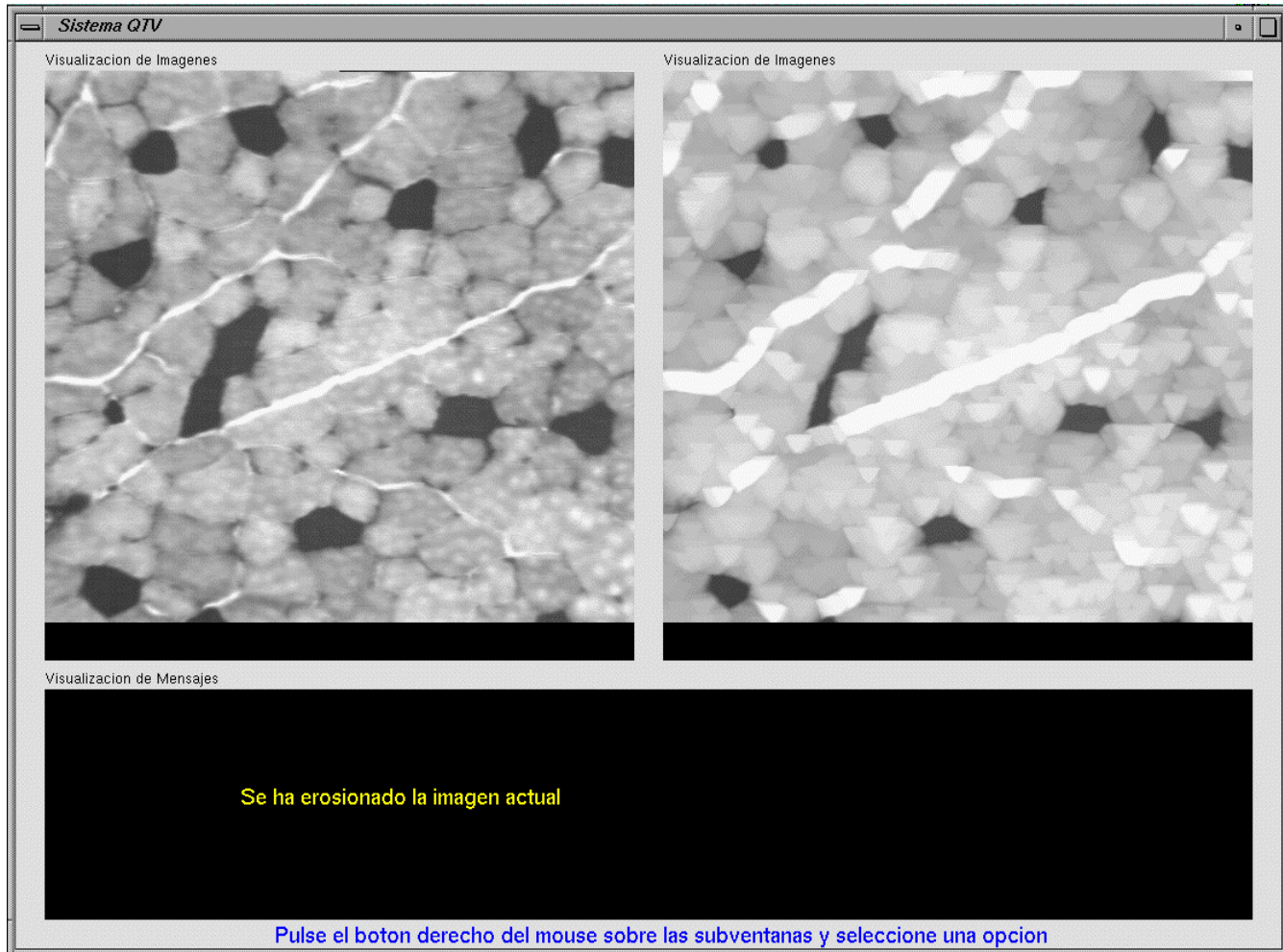


# Erosion

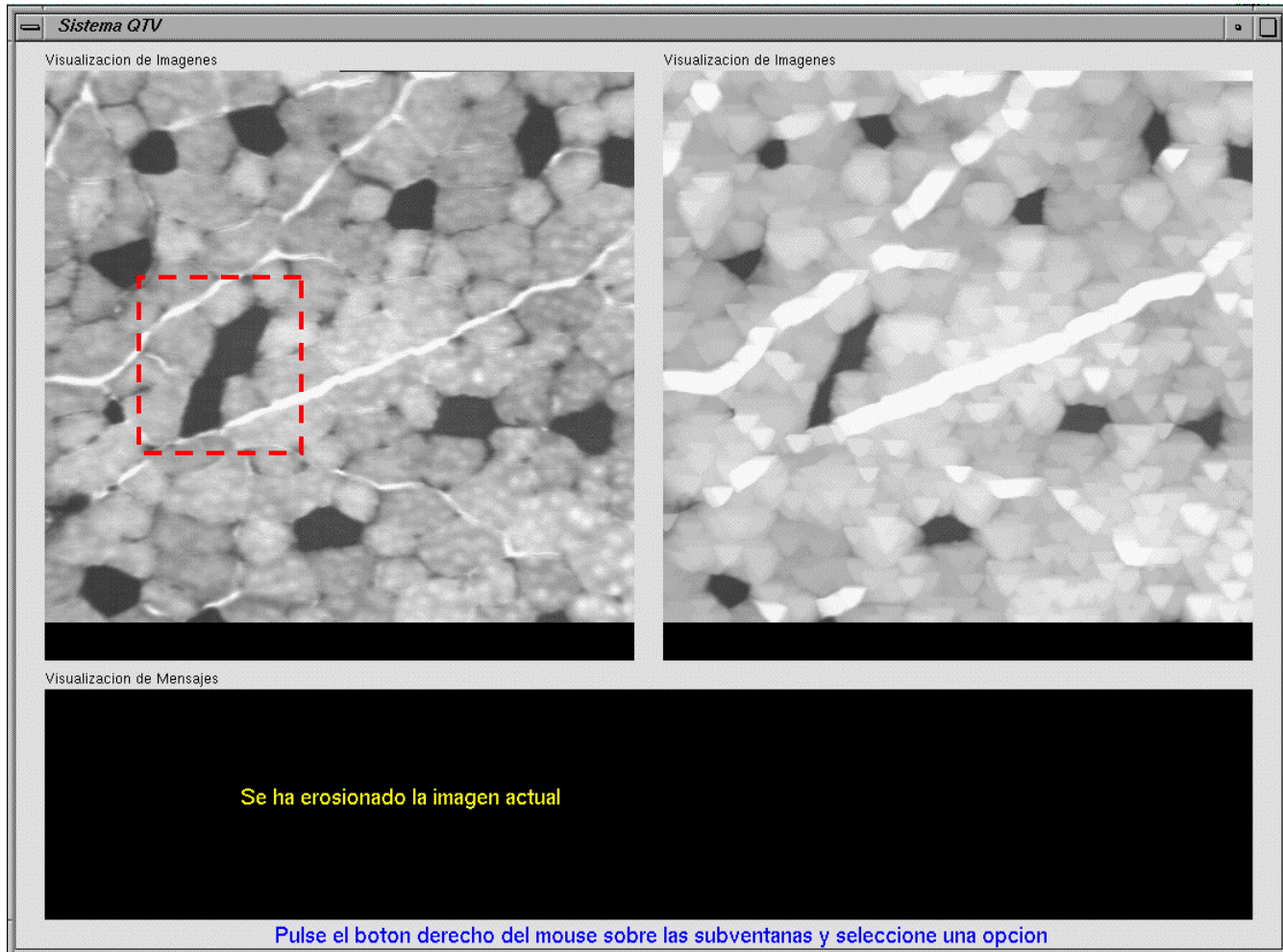


# Exemple d'érosion

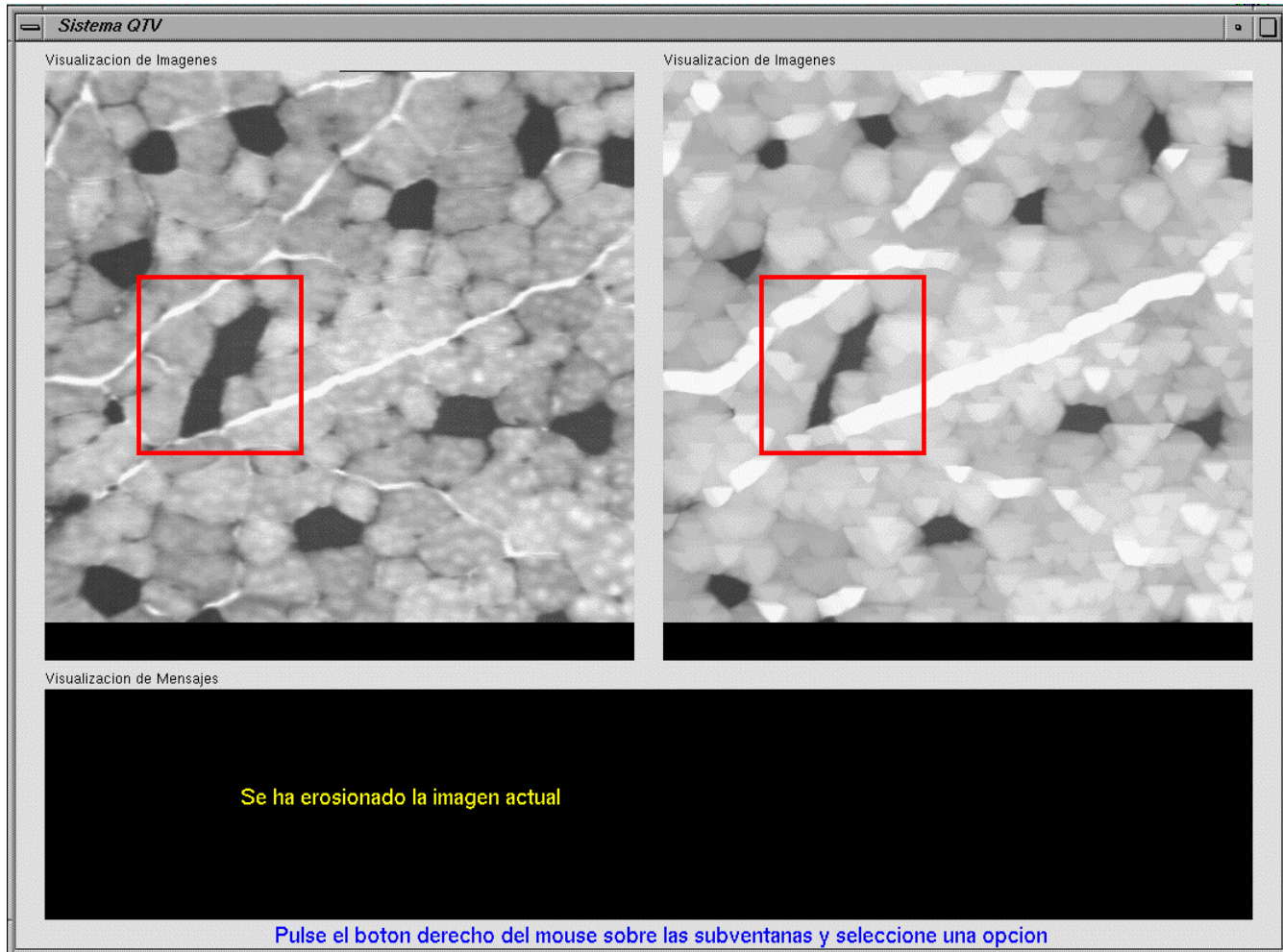
# Exemple d'érosion



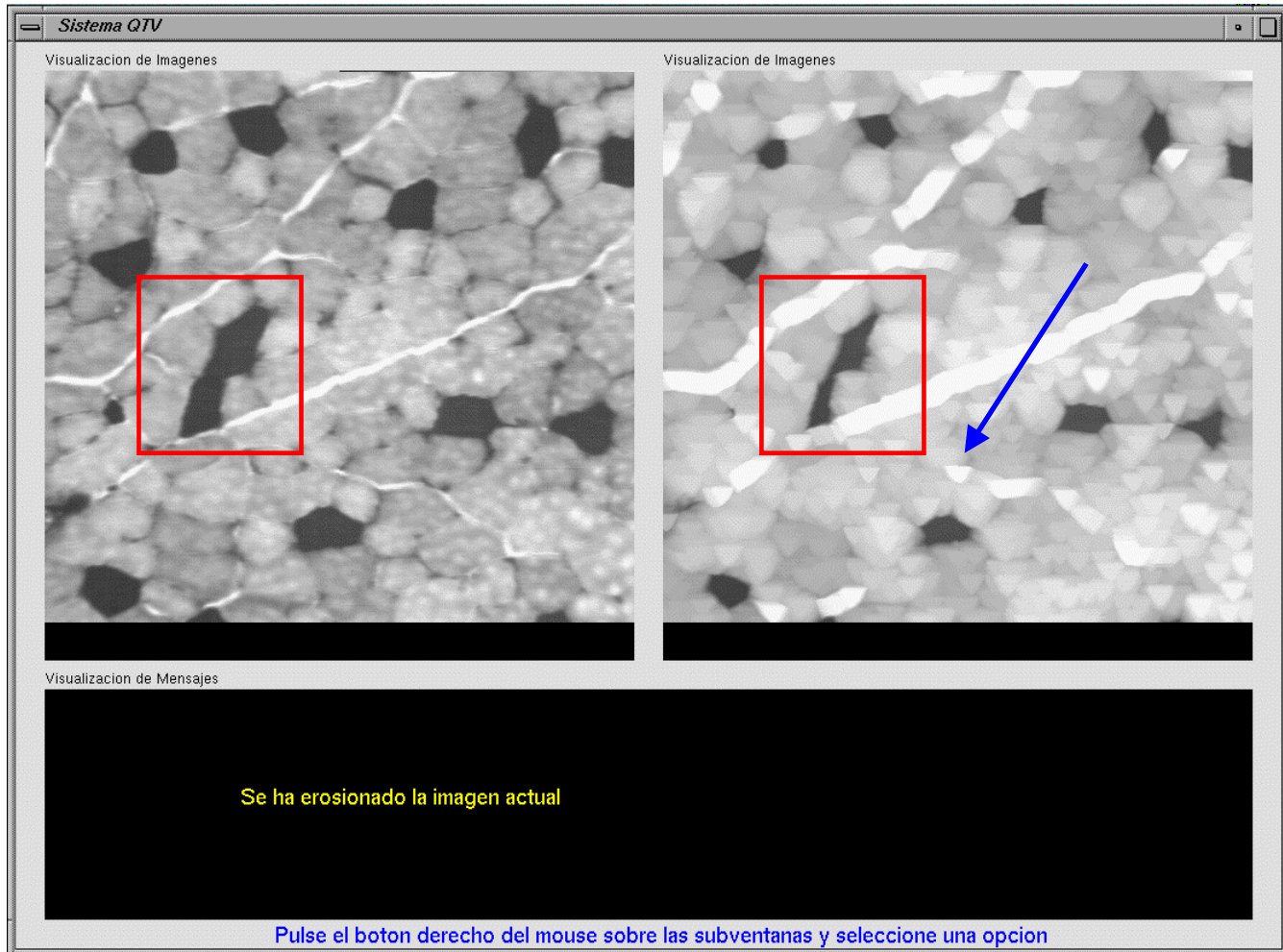
# Exemple d'érosion



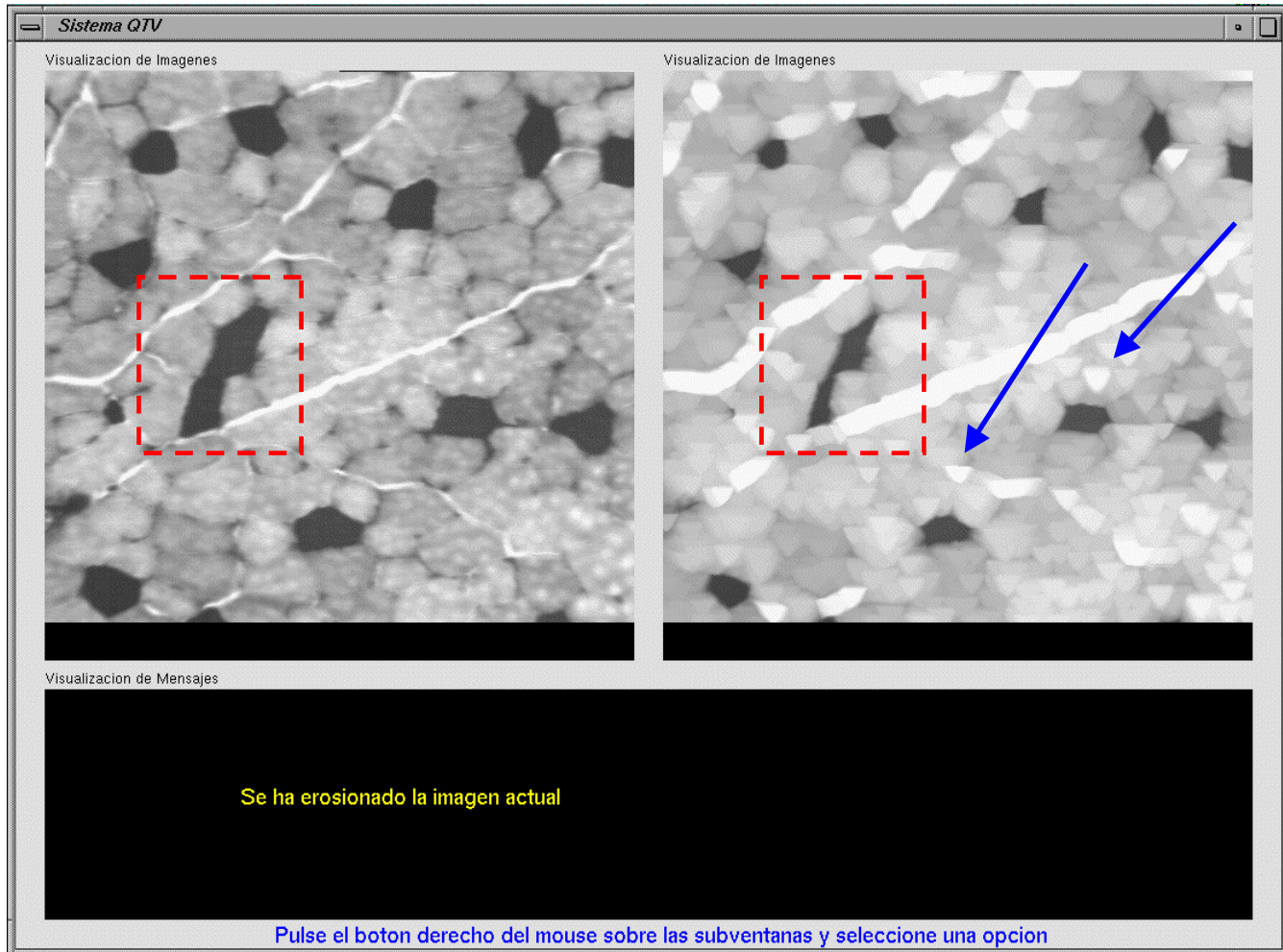
# Exemple d'érosion



# Exemple d'érosion

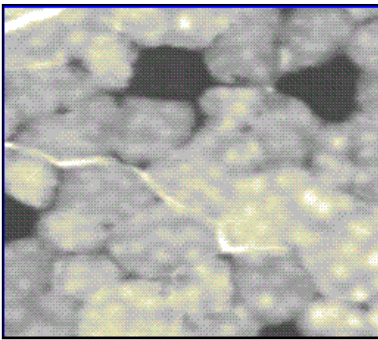


# Exemple d'érosion

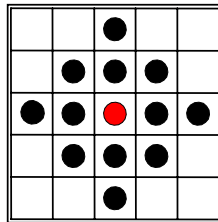


# Exemple de dilatation

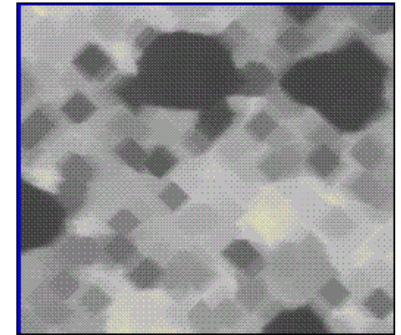
$$(A \oplus B) = \left[ \bigcup_{b \in B} \{ a+b, a \in A \} \right]$$



**Image A**

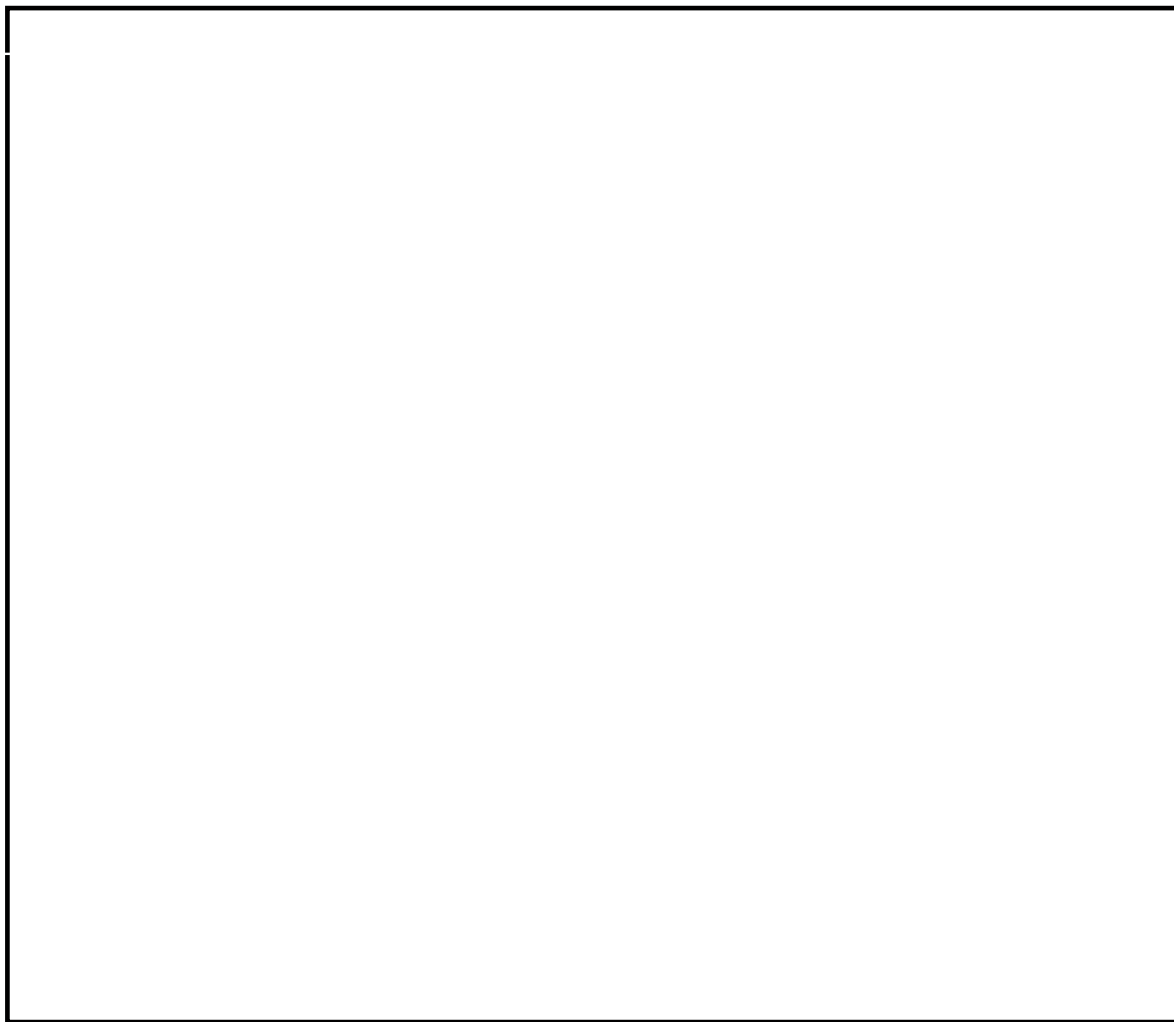


**Patron B**

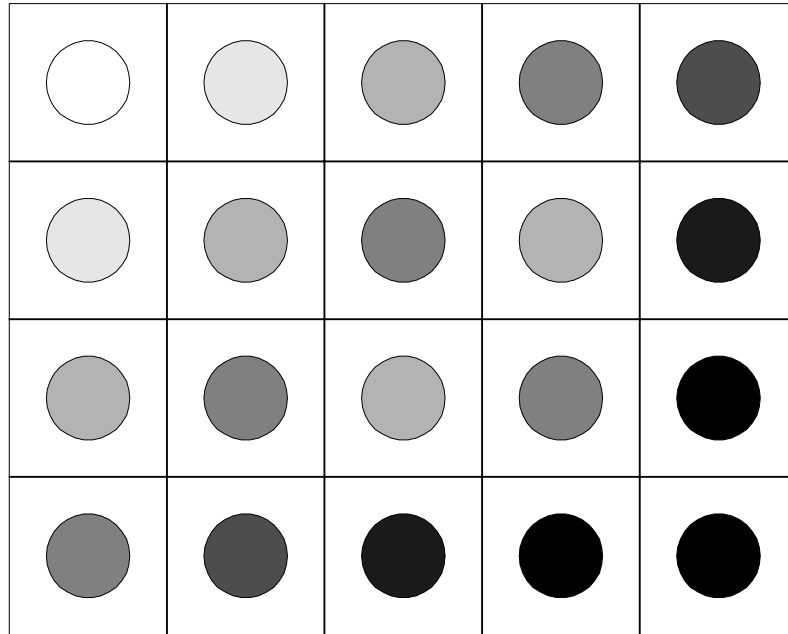


**Image A  
après dilatation**

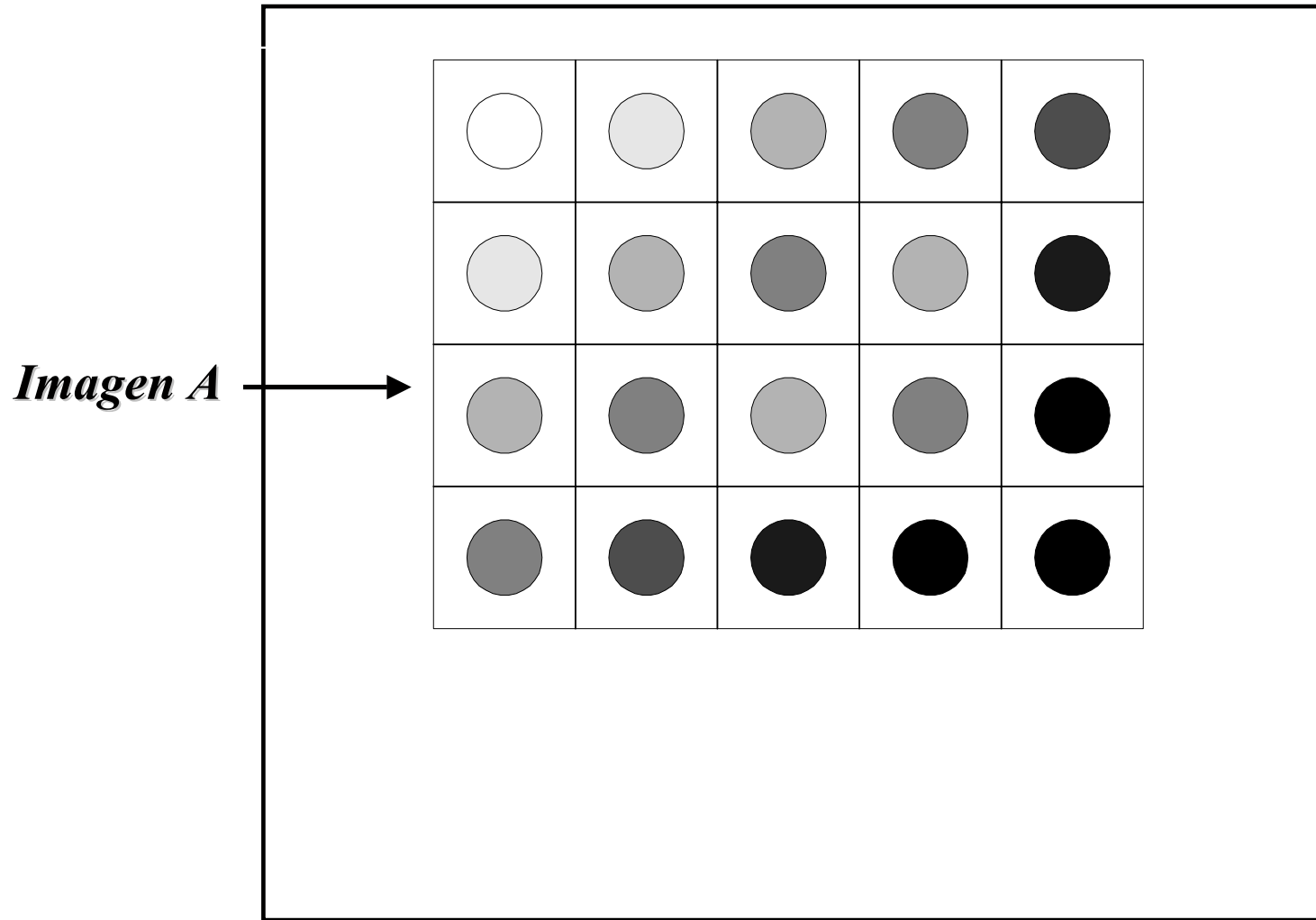
# Dilatation



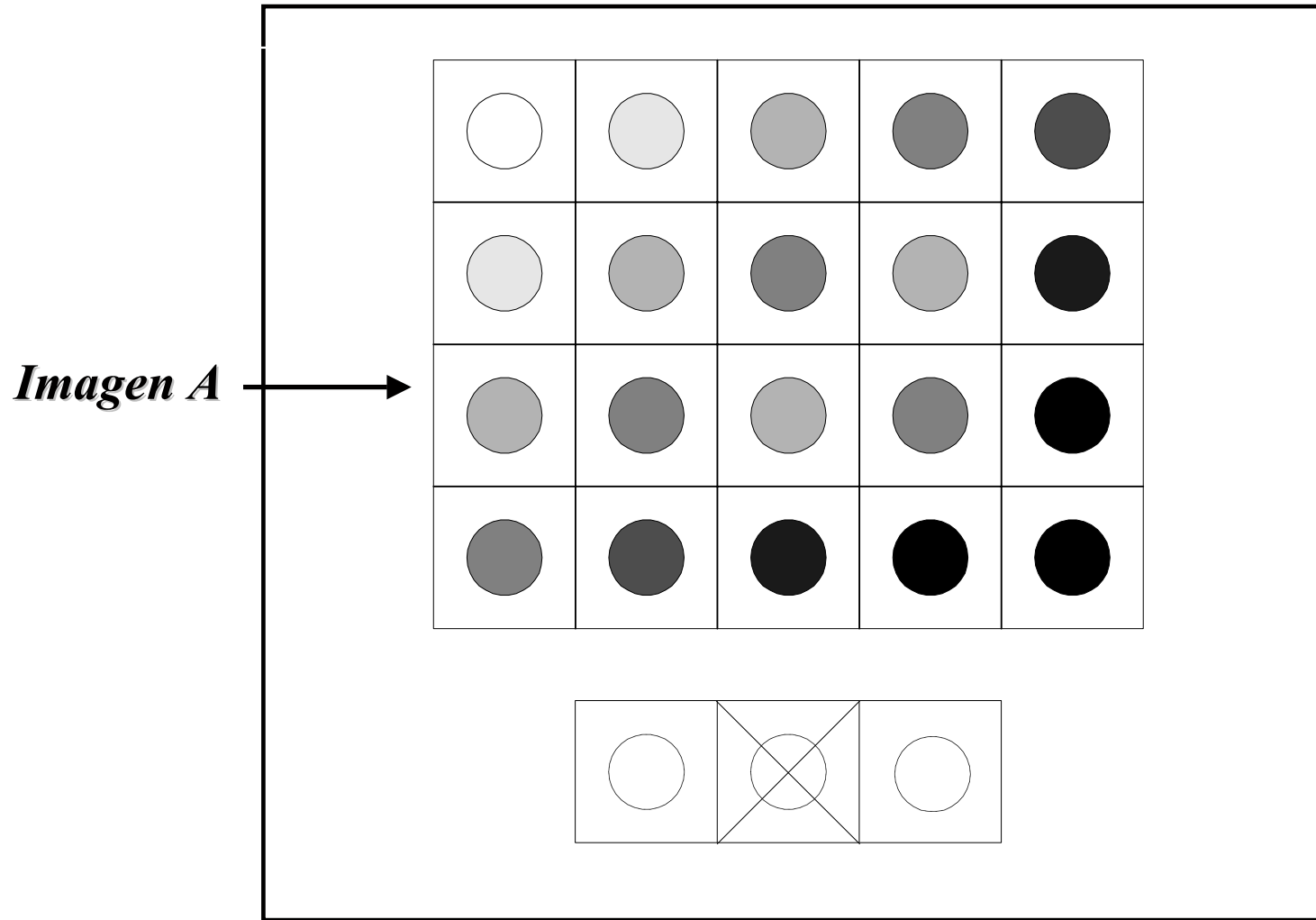
# Dilatation



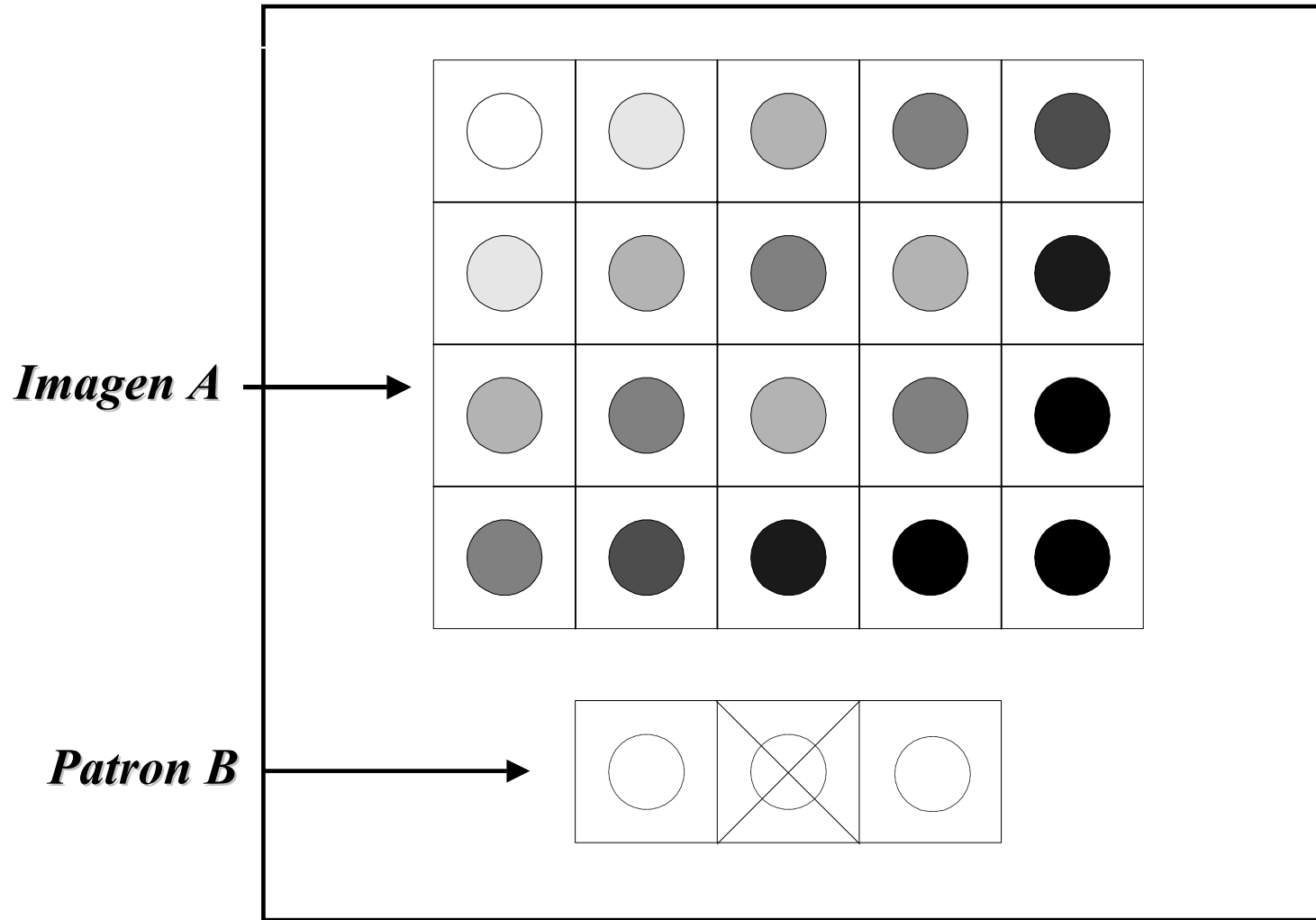
# Dilatation



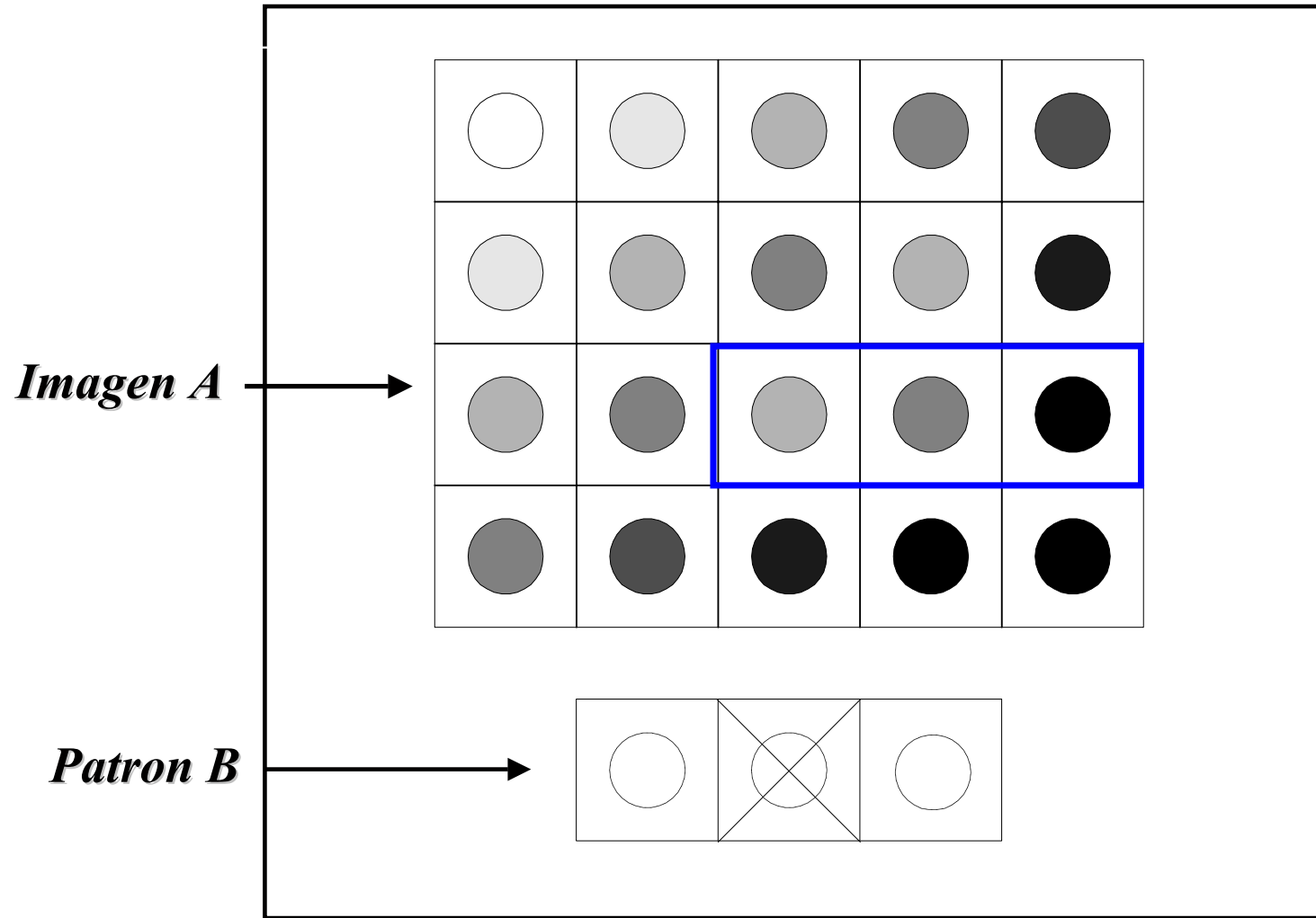
# Dilatation



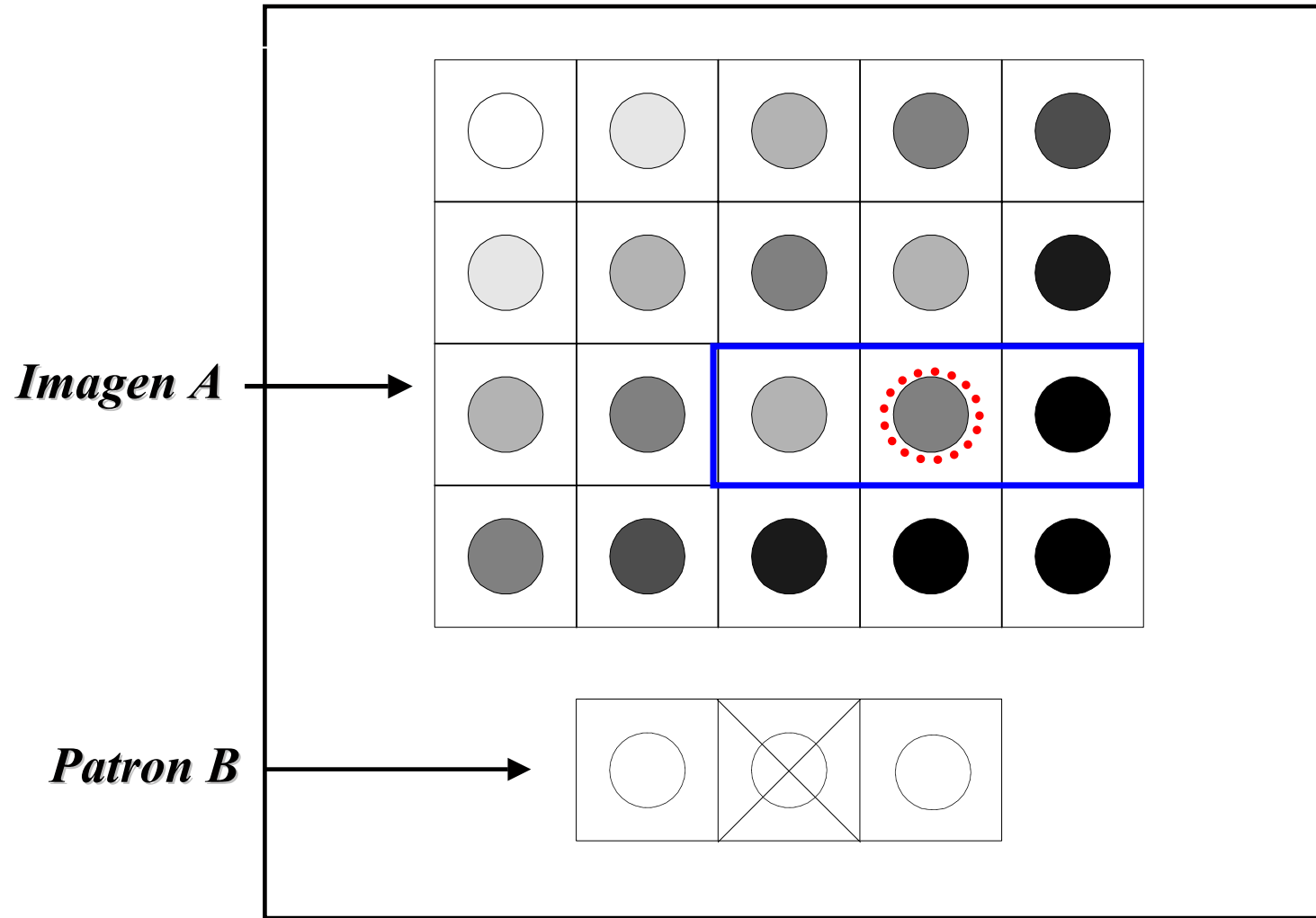
# Dilatation



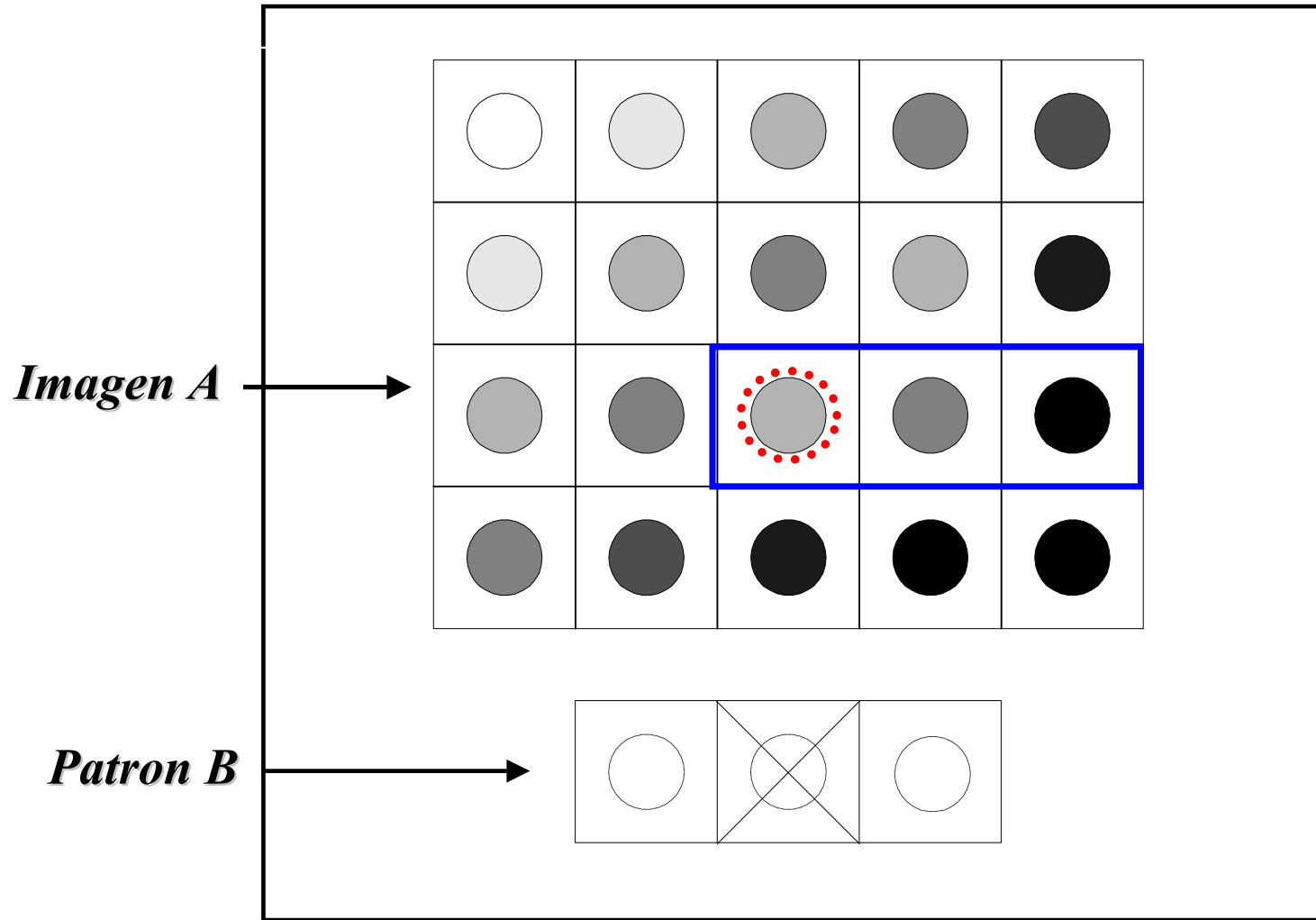
# Dilatation



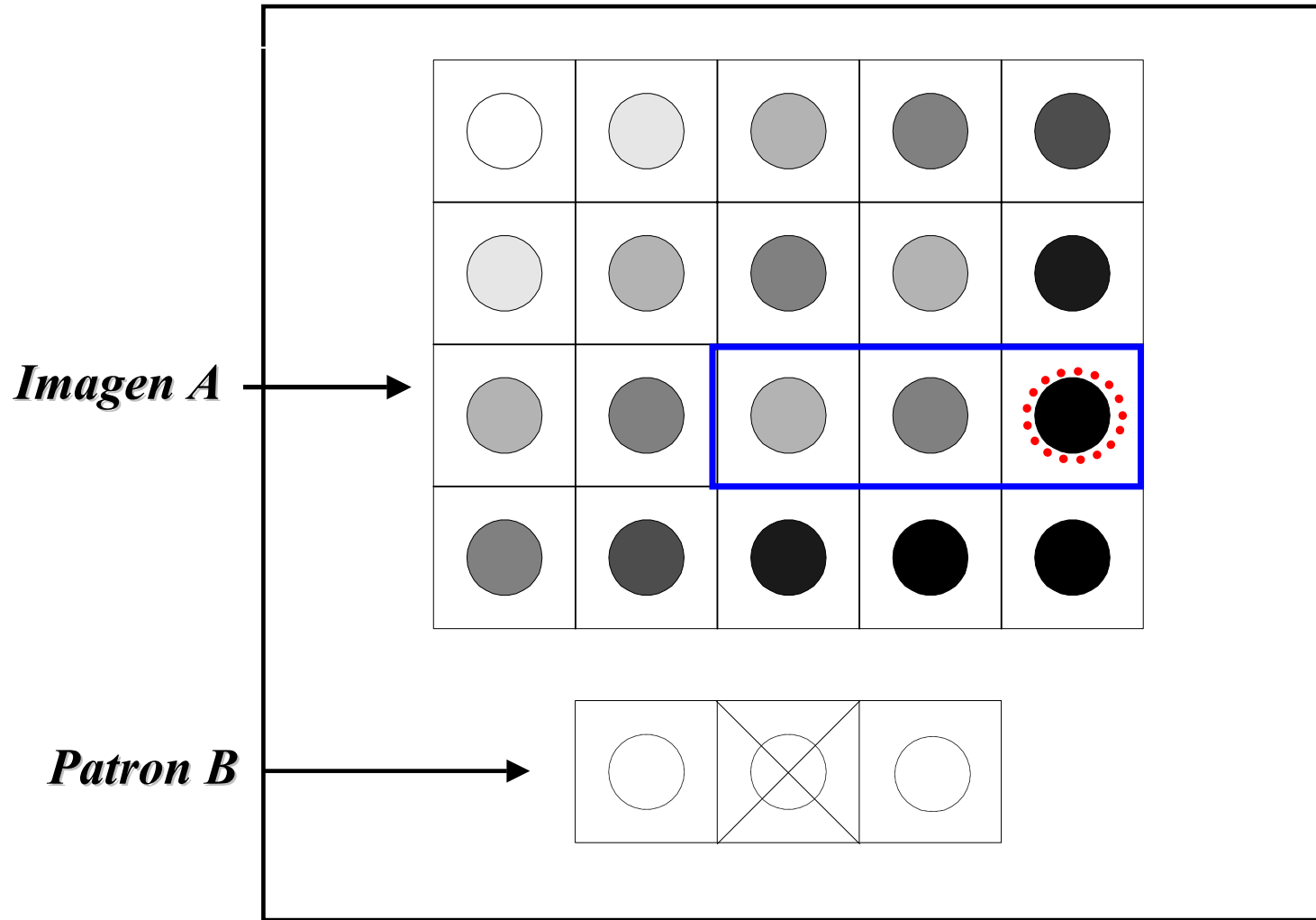
# Dilatation



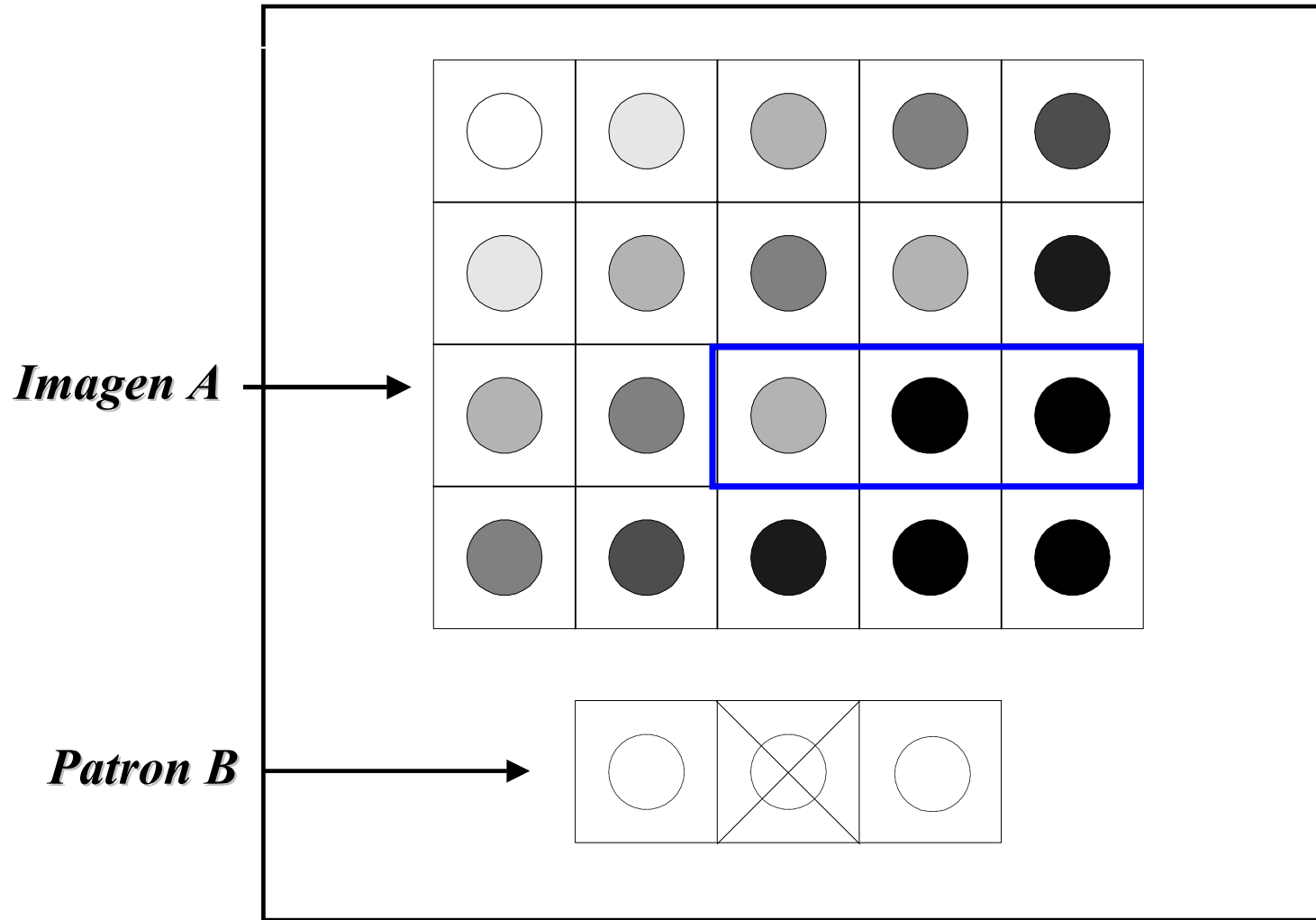
# Dilatation



# Dilatation

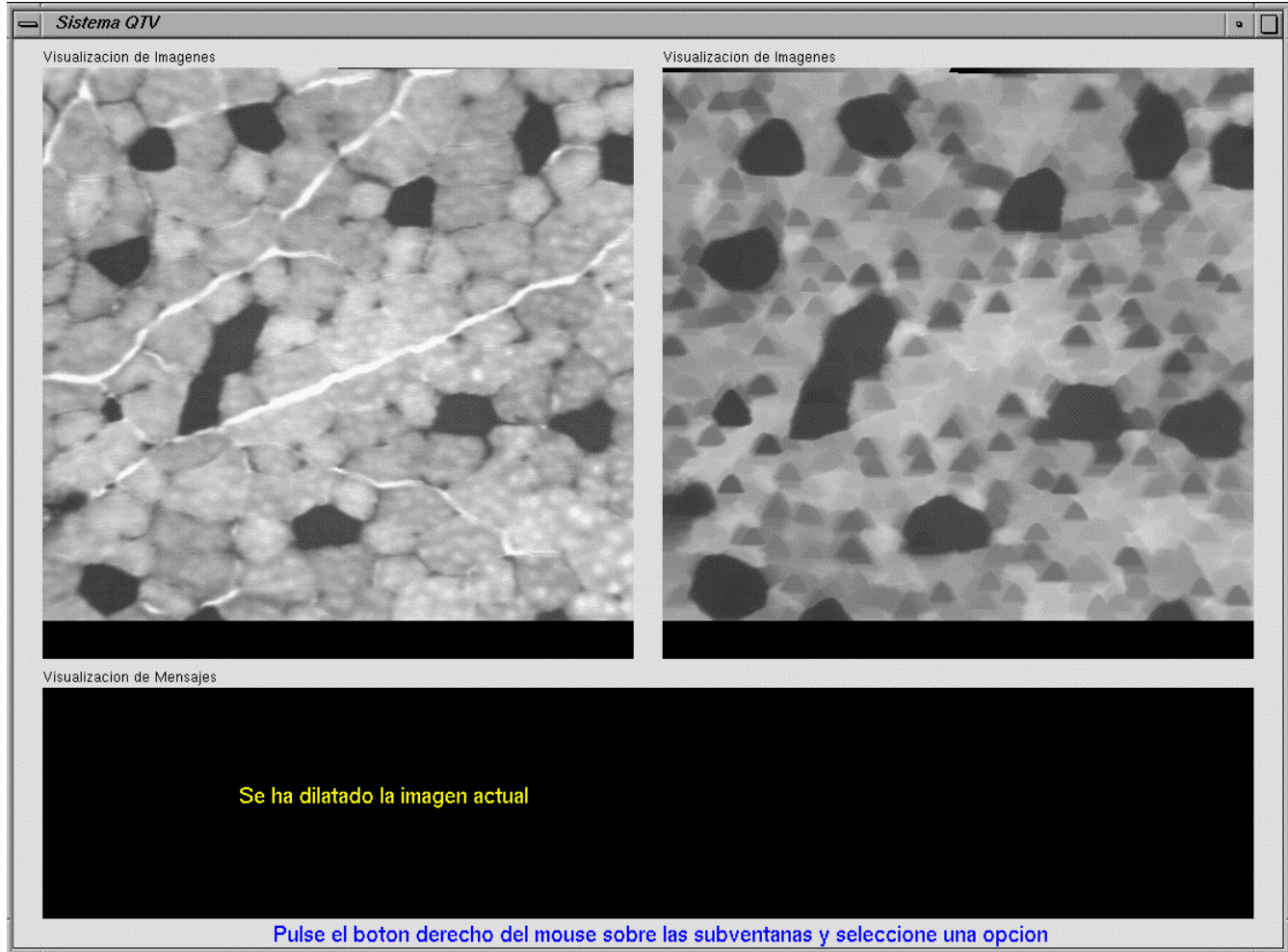


# Dilatation

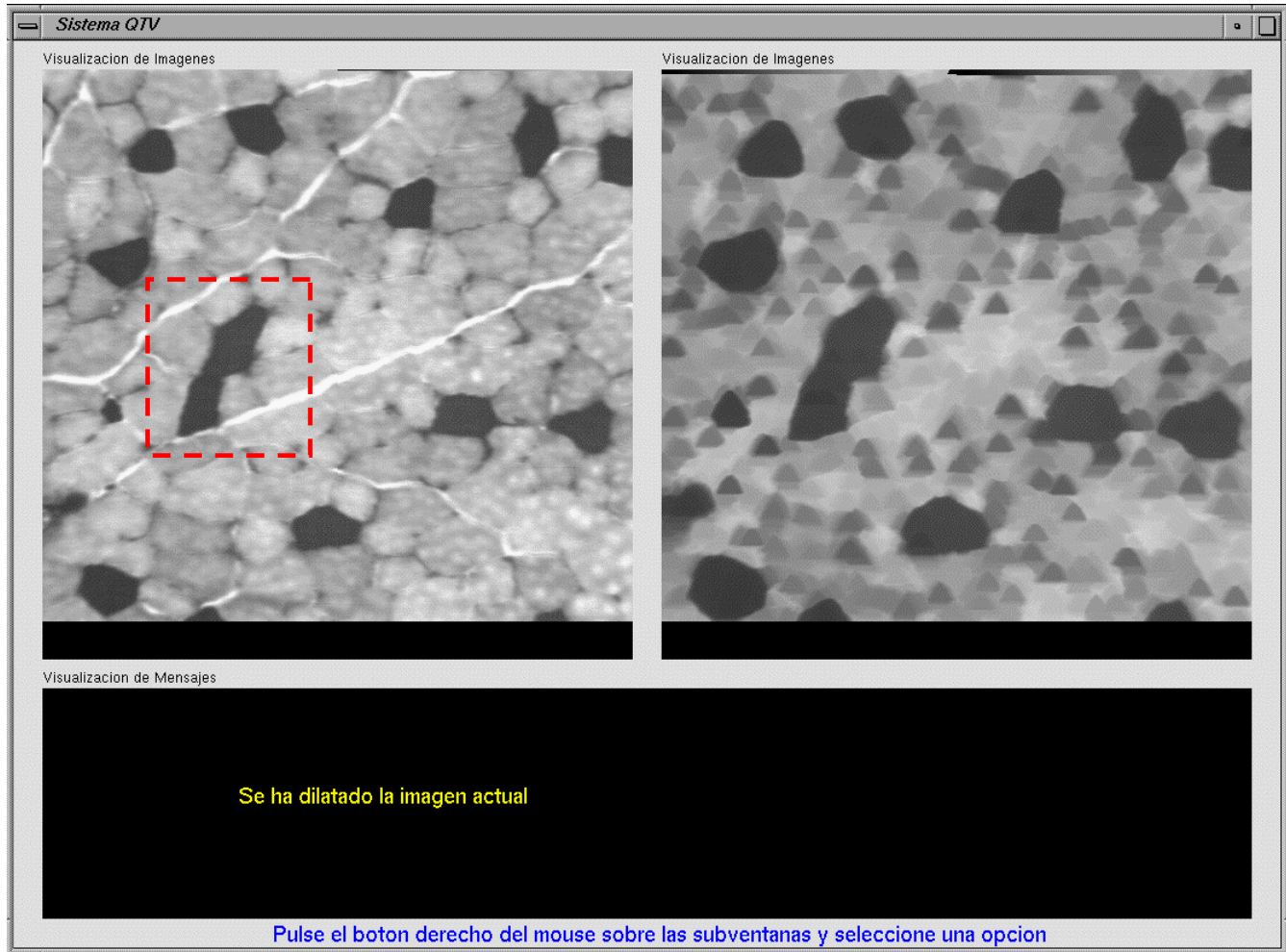


# Exemple de dilatation

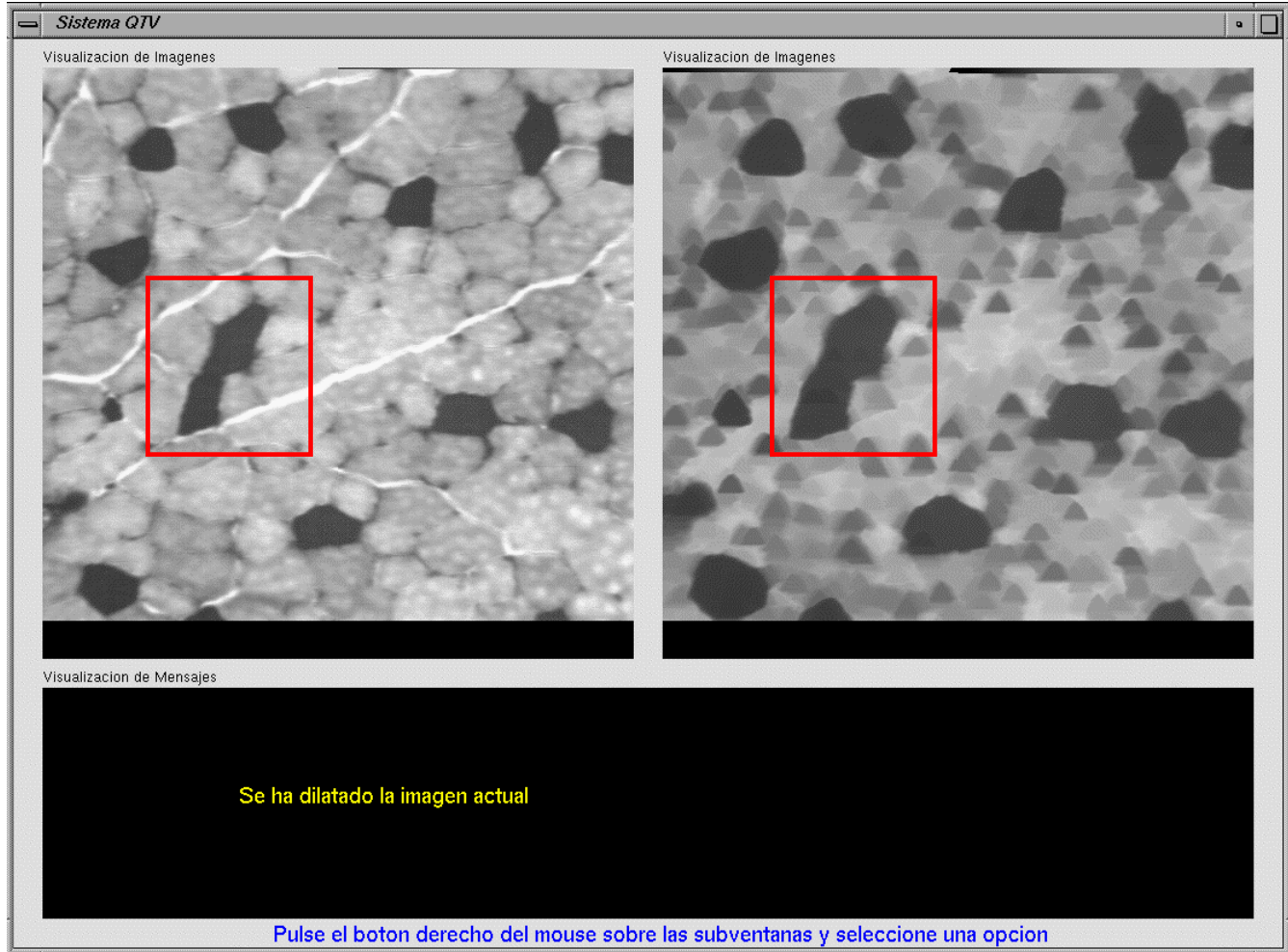
# Exemple de dilatation



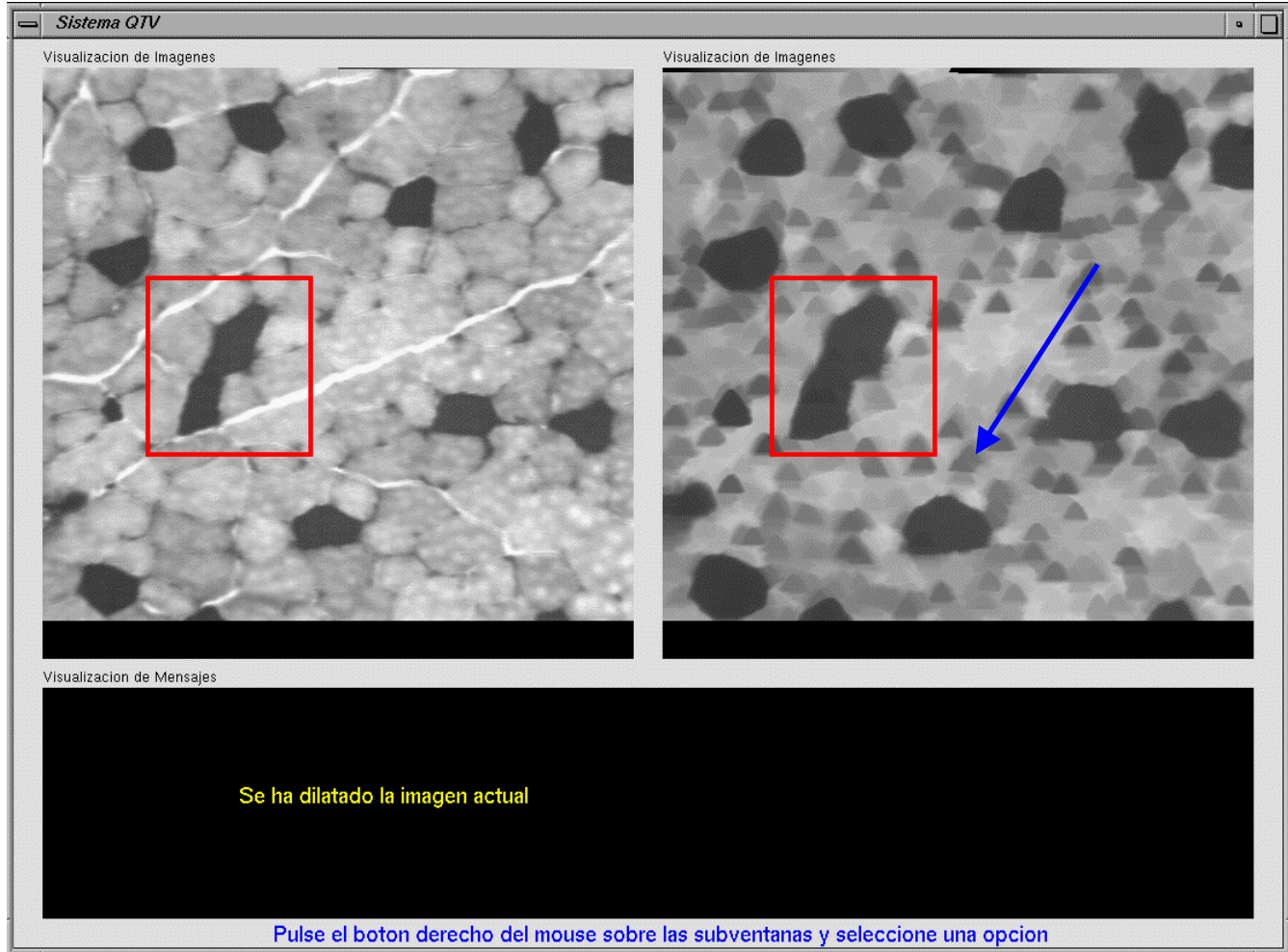
# Exemple de dilatation



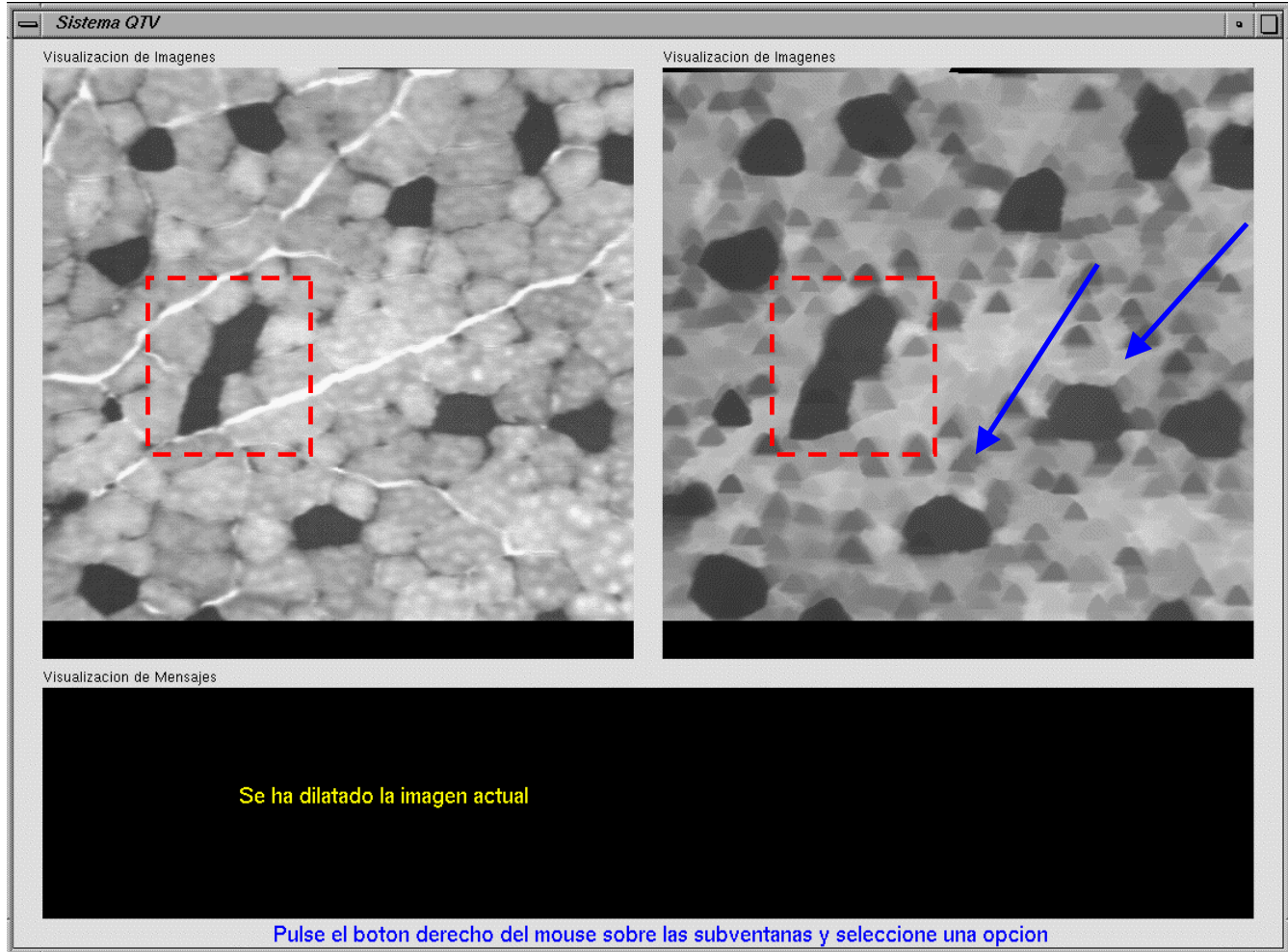
# Exemple de dilatation



# Exemple de dilatation



# Exemple de dilatation



# Arbre Quaternaire Générique

Image A

|    |    |
|----|----|
| 00 | 01 |
| 02 | 03 |

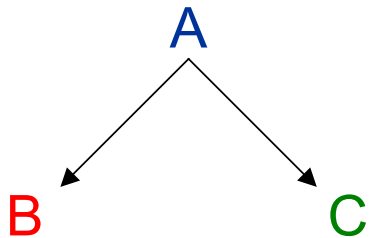
Image B

|    |    |
|----|----|
| 00 | 01 |
| 02 |    |

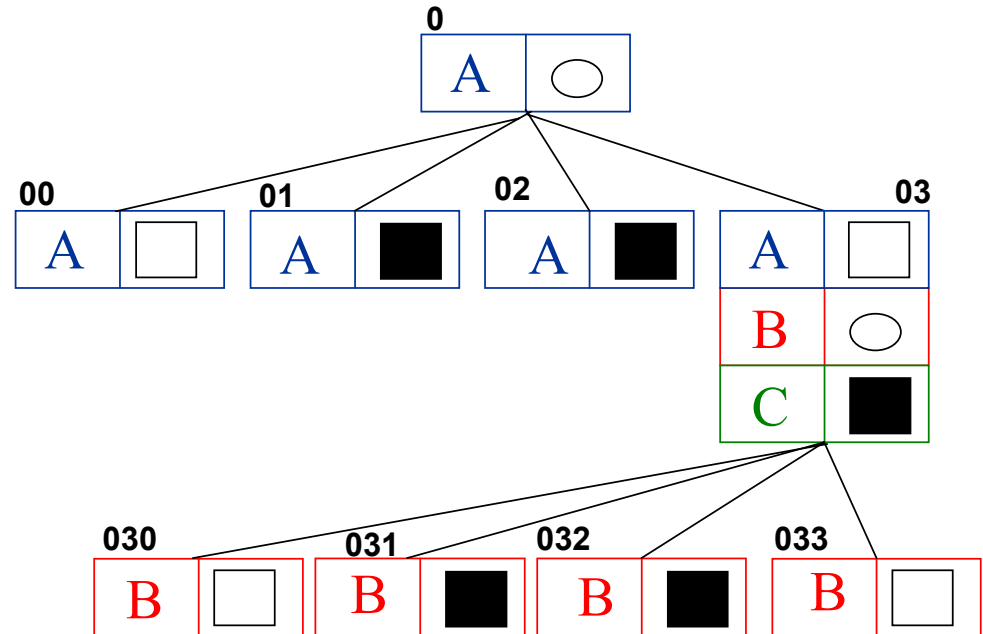
Image C

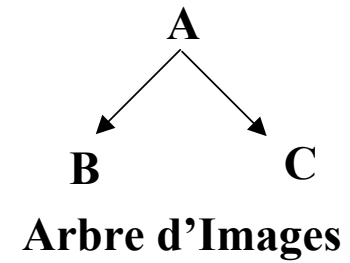
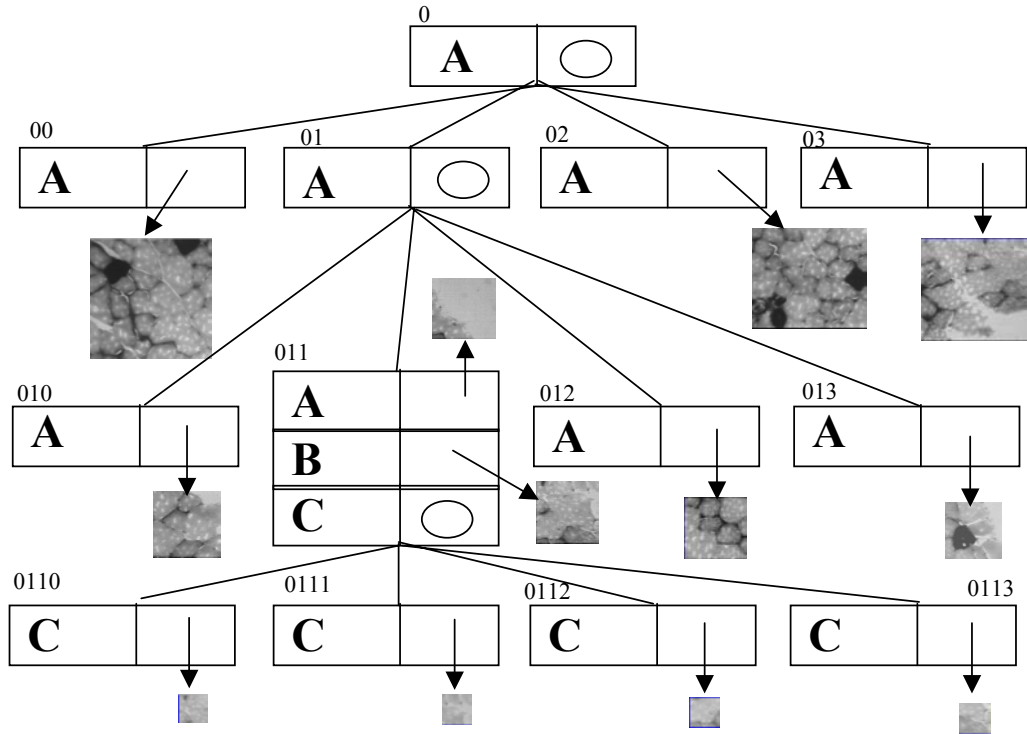
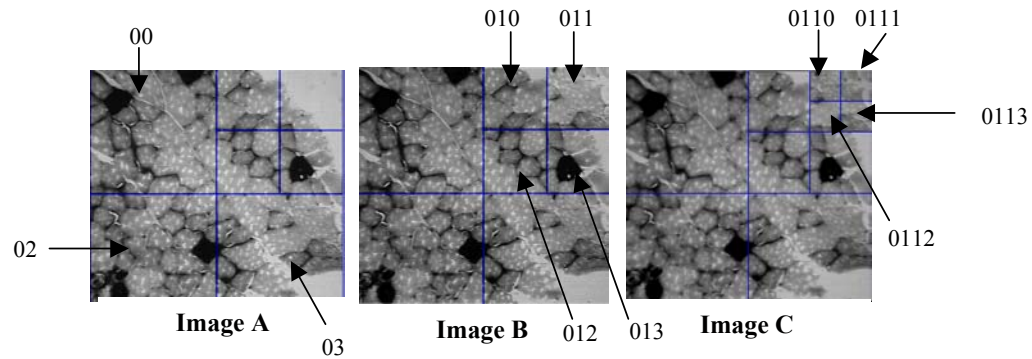
|    |    |
|----|----|
| 00 | 01 |
| 02 | 03 |

Arbre d'Images



Arbre Quaternaire Générique

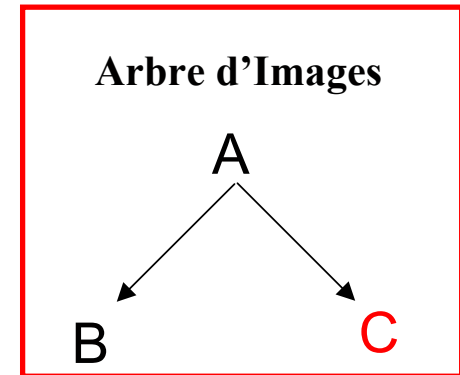
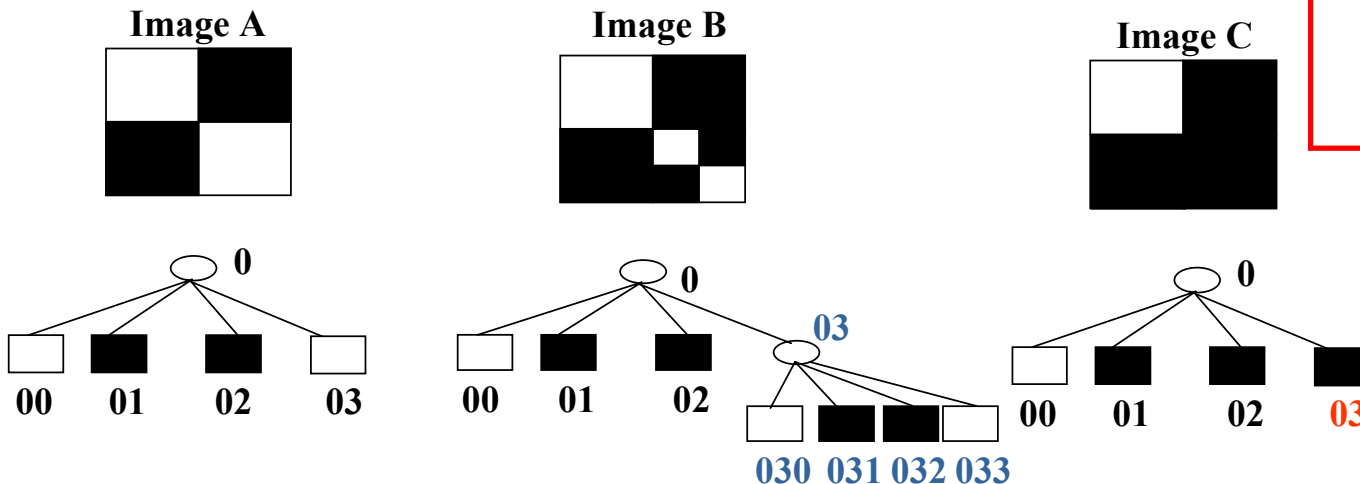




**Arbre Quaternaire Générique**

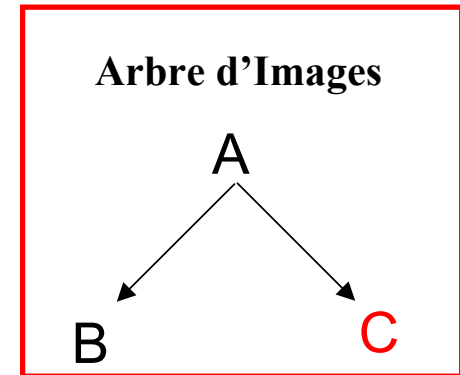
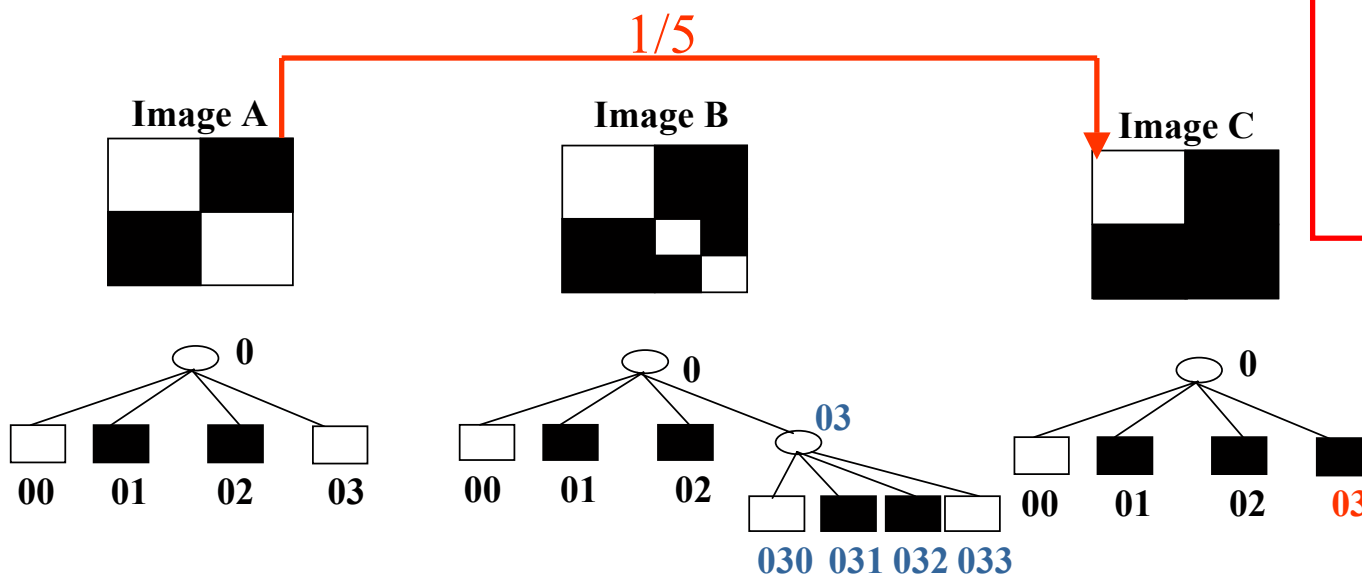
# Arbre d'Images

- Les images sont organisées en arbre
- Une image  $i$  est insérée dans l'Arbre d'Images comme fille de l'image  $i'$  telle que  $d(i, i')$  soit la plus proche de 0



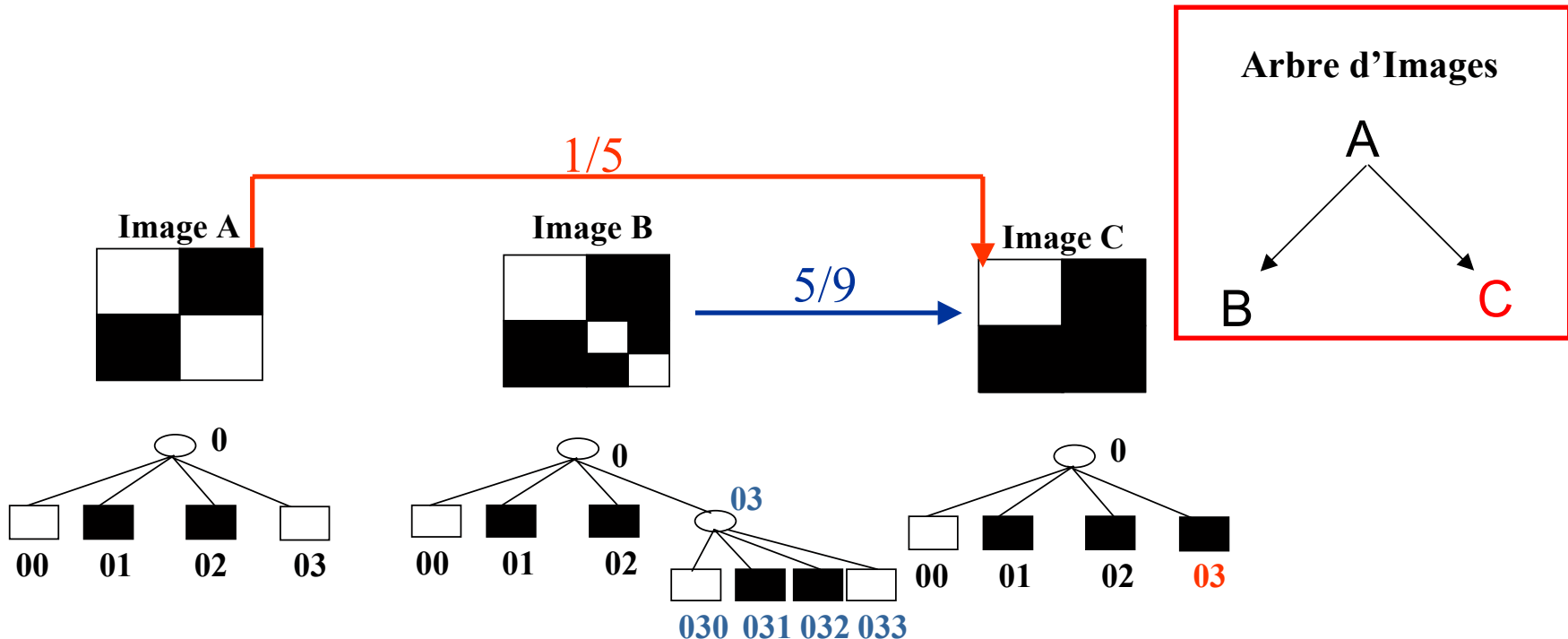
# Arbre d'Images

- Les images sont organisées en arbre
- Une image  $i$  est insérée dans l'Arbre d'Images comme fille de l'image  $i'$  telle que  $d(i, i')$  soit la plus proche de 0



# Arbre d'Images

- Les images sont organisées en arbre
- Une image  $i$  est insérée dans l'Arbre d'Images comme fille de l'image  $i'$  telle que  $d(i, i')$  soit la plus proche de 0



# Opérations

## Opérations sur les images stockées dans l'Arbre Quaternaire Générique :

- Lecture d'une ou plusieurs images
- Insertion / suppression d'une image
- Modification d'une image
- Comparaison d'images
- Suivi de l'évolution d'une région

# Arbre Quaternaire Générique

Image 1

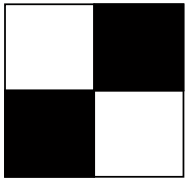


Image 2

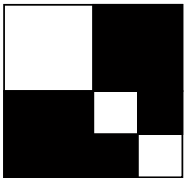


Image 3

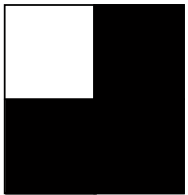
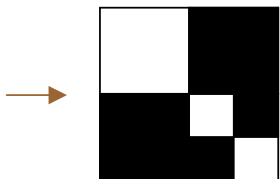
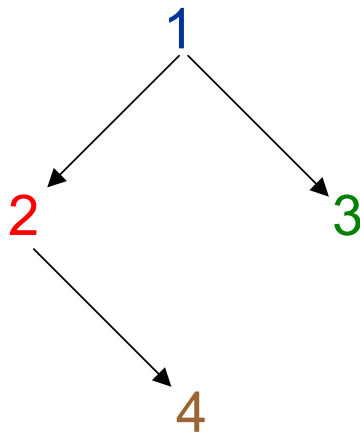


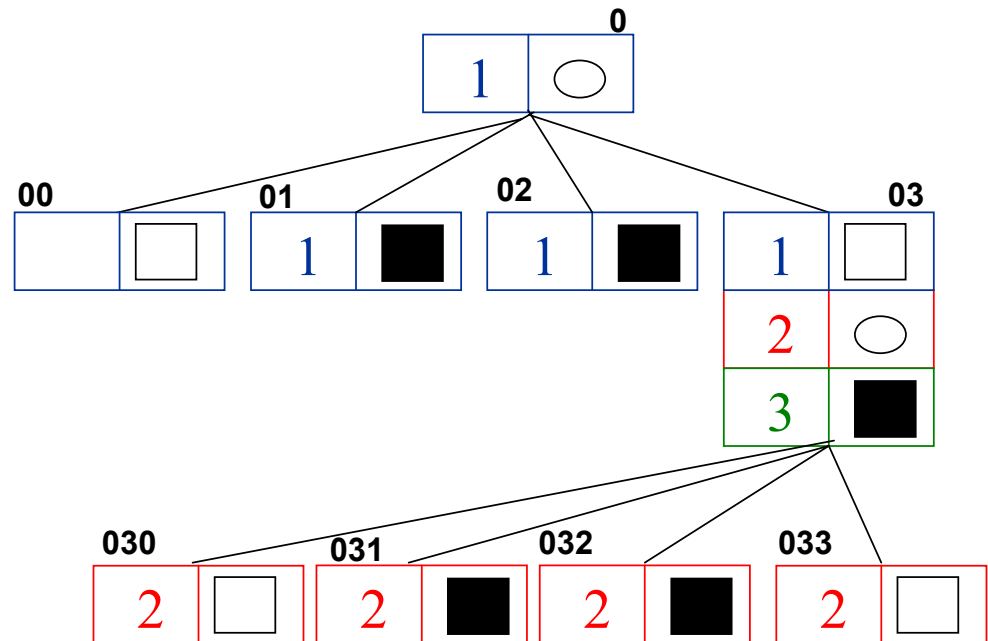
Image 4



Arbre d'images



Arbre Quaternaire Générique



# Arbre Quaternaire Générique

Image 1

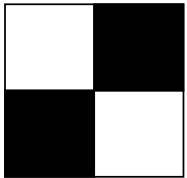


Image 2

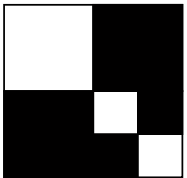


Image 3

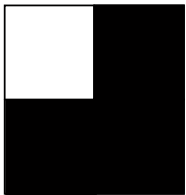
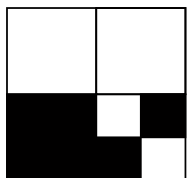
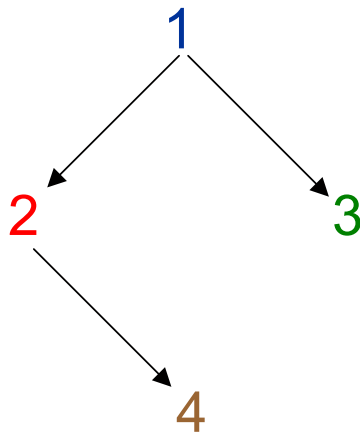


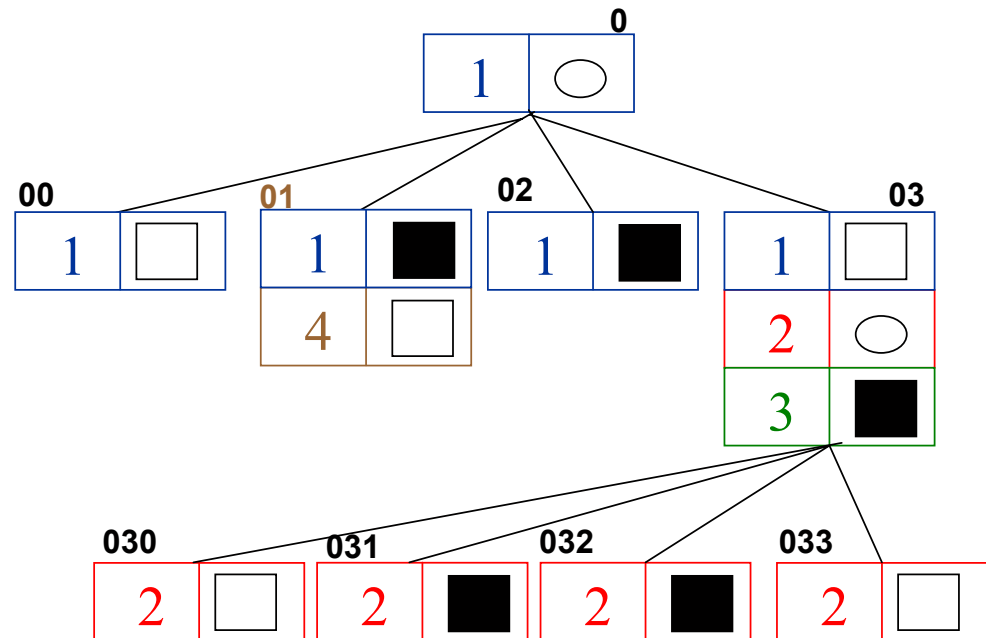
Image 4



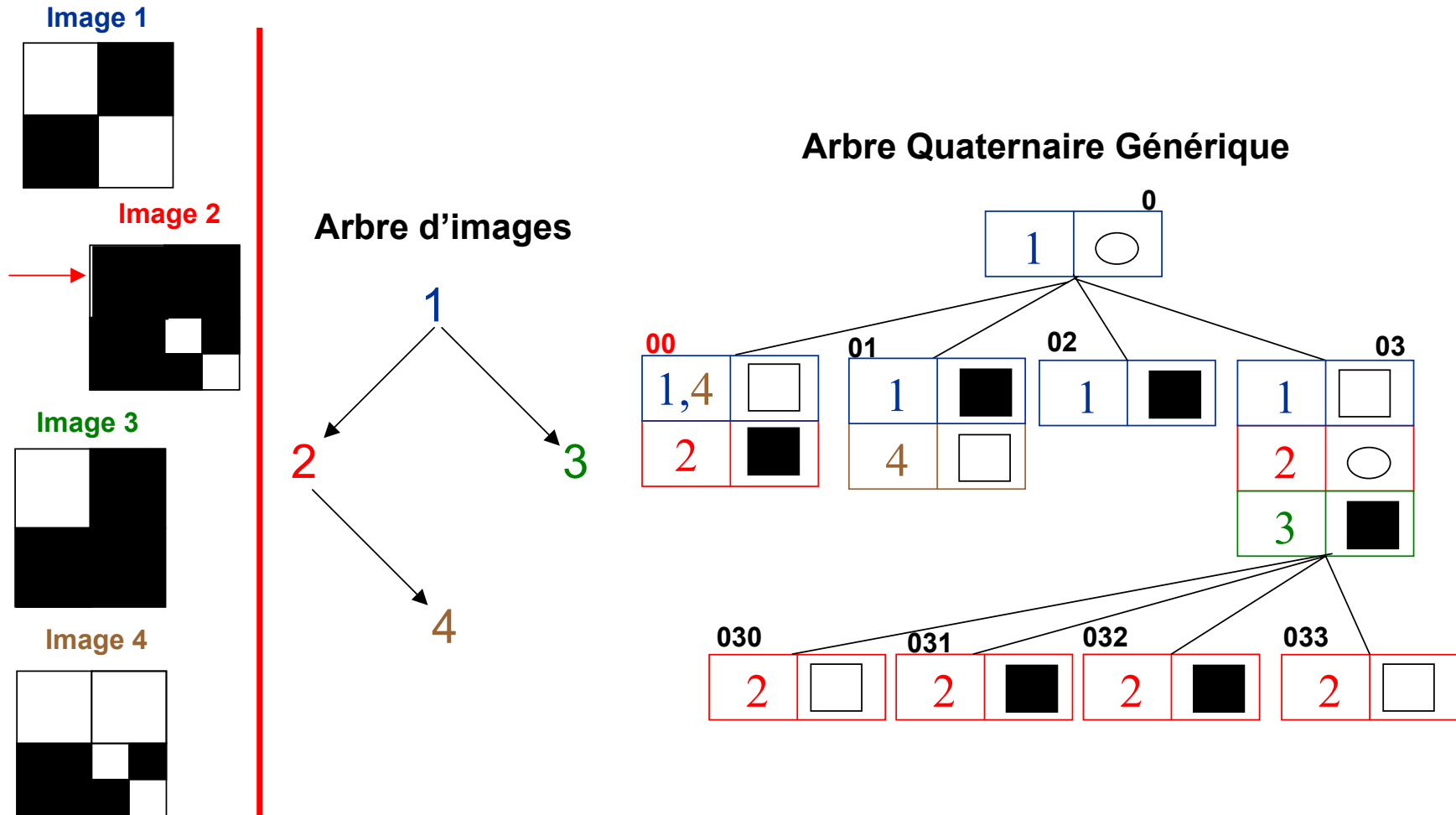
Arbre d'images



Arbre Quaternaire Générique



# Arbre Quaternaire Générique



# Travaux proches

| <b>Chevauchement</b>                                                                       | <b>Arbre Quaternaire Générique</b>                            |
|--------------------------------------------------------------------------------------------|---------------------------------------------------------------|
| <b>Création d'une image supplémentaire à chaque groupe de mises à jour</b>                 | <b>Mise à jour directe sur n'importe quelle image</b>         |
| <b>Copie de tout un chemin à chaque ajout d'image</b>                                      | <b>Pas de redondance</b>                                      |
| <b>Pas de suppression d'image</b>                                                          | <b>Suppression logique ou physique</b>                        |
| <b>Lecture de plusieurs chemins pour accéder à une même région dans différentes images</b> | <b>Accès direct à une même région dans différentes images</b> |
| <b>⇒ stockage de séquences d'images non modifiables<br/>(séquences d'images vidéo)</b>     | <b>⇒ élément d'un atelier de traitement d'images</b>          |

# Récapitulatif

## Utilisation de l'arbre quaternaire (*quadtree*) pour les images

- Filtrage multi-niveaux (*coarse to fine*) :
  - ◆ Chaque image est représentée par un *Quadtree-based feature vector*
  - ◆ *On compare les images niveau par niveau dans l'arbre*
  - ◆ *On peut utiliser des index (arbre R, hachage ...) au dessus des arbres quaternaires*
- Compactage/Compression
- Stockage de séquences d'images similaires

# États de l'art sur l'utilisation des arbres quaternaires dans le domaine des images

- M. Manouvrier, M. Rukoz et G. Jomier, *Quadtree representations for storage and manipulation of clusters of images*, Image and Vision Computing (Elsevier Science Journal), May 2002, vol 20 issue 7, pp 513-527
- M. Manouvrier, M. Rukoz, G. Jomier, *Quadtree-Based Image Representation and Retrieval*, chapitre à paraître dans Spatial Databases : Technologie, Techniques and Trends, Y. Manolopoulos, A. Papadopoulos and M. Vassilakopoulos (Eds), 2005

# 5. Exemple de domaine d'application

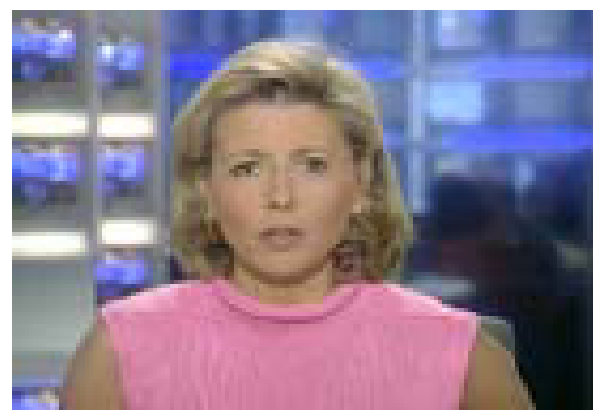
- **Annotation : propagation de mots-clés à partir du contenu visuel des images**
- **Signatures spécifiques (ex. détection de visages)**
- **Recherche par points d'intérêt**
- **Imagerie médicale**

## *Indexation texte-image*

### *Propagation d'annotation fondée sur la similarité visuelle*

---

- Propagation de mots-clés :



■ Similarité visuelle ➡



Mots-clés ? Claire Chazal, gp ← Propagation ■ Mots-clés : Claire Chazal, gp.

- ✓ Indexation textuelle automatique,
- ✓ Auto-complétion,



# Protocole pour associer automatiquement des mots à une image

Corpus d'images

(indexées textuellement et visuellement)

Base de test

Base de référence



ciel mer fleur arbre



ciel arbre



arbre fleur route

|       | Arbre | Ciel | Fleur | Mer |
|-------|-------|------|-------|-----|
| Rouge | 2     | 1    | 2     | 1   |
| Vert  | 3     | 2    | 2     | 1   |
| Bleu  | 2     | 2    | 1     | 1   |

Classes visuelles

Estimation des probabilités

arbre (5fois), fleur (4 fois)

fusion

# Calcul du score de la classification

Image de la base de test

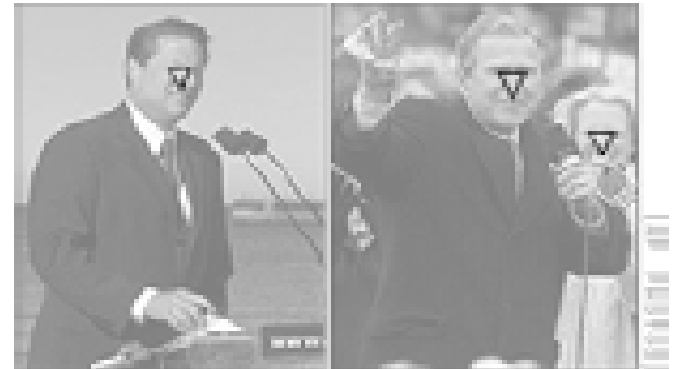


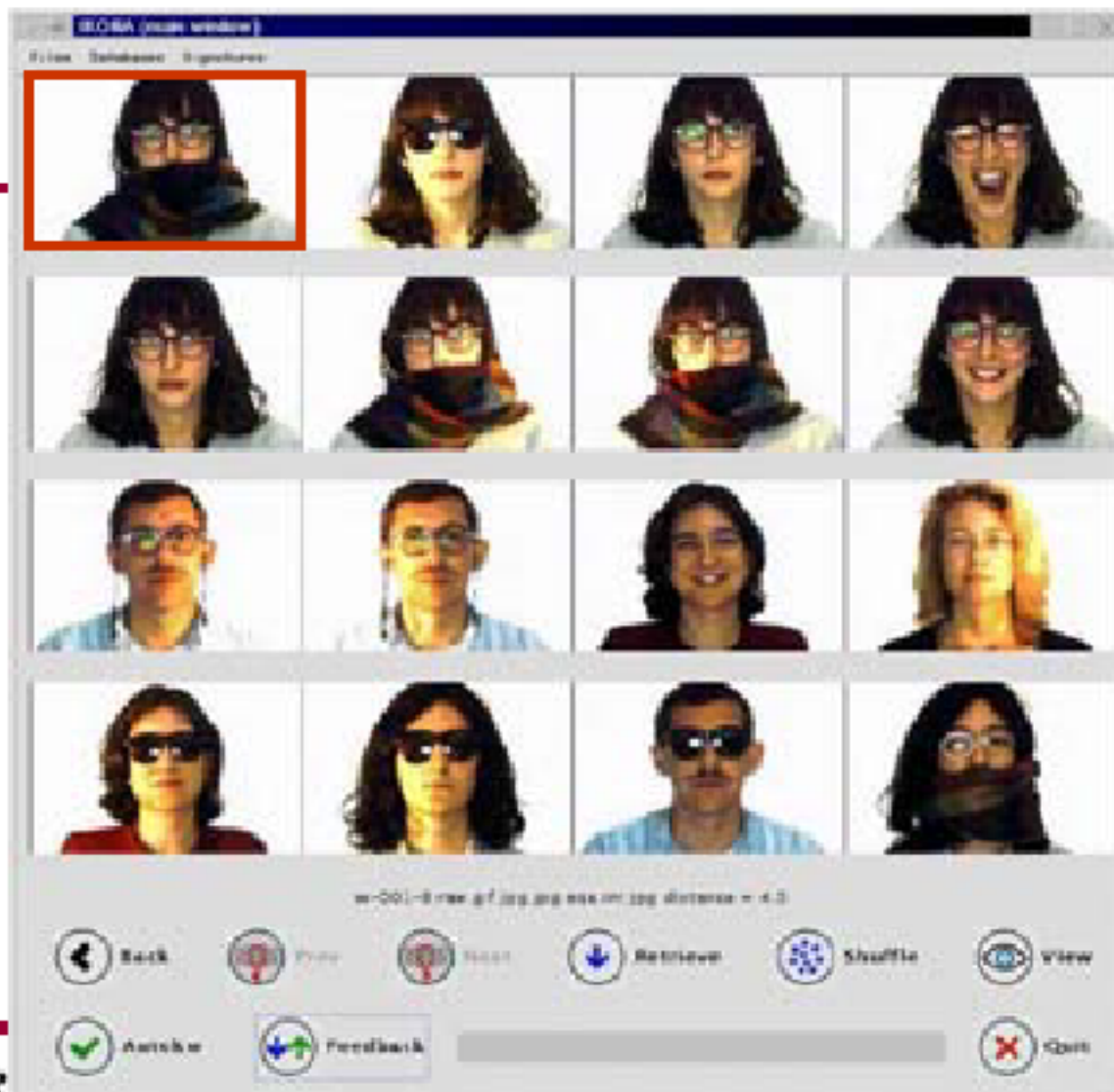
| Mots de références<br>de l'image | Mots trouvés par le<br>système |
|----------------------------------|--------------------------------|
| fleur arbre lampadaire           | arbre fleur ciel               |

Score NS=2/3-1/200

# Détection de Visages

---





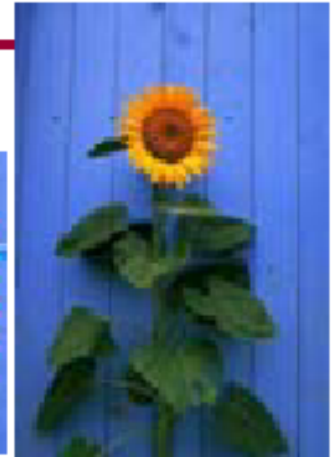
# *Description locale des couleurs*

- Extraction des points d'intérêt:

## Détecteur de Harris couleur

=> Caractéristiques :

- Bonne répétabilité,
- Dérivées de premier ordre seulement



- Caractérisation des points d'intérêt :

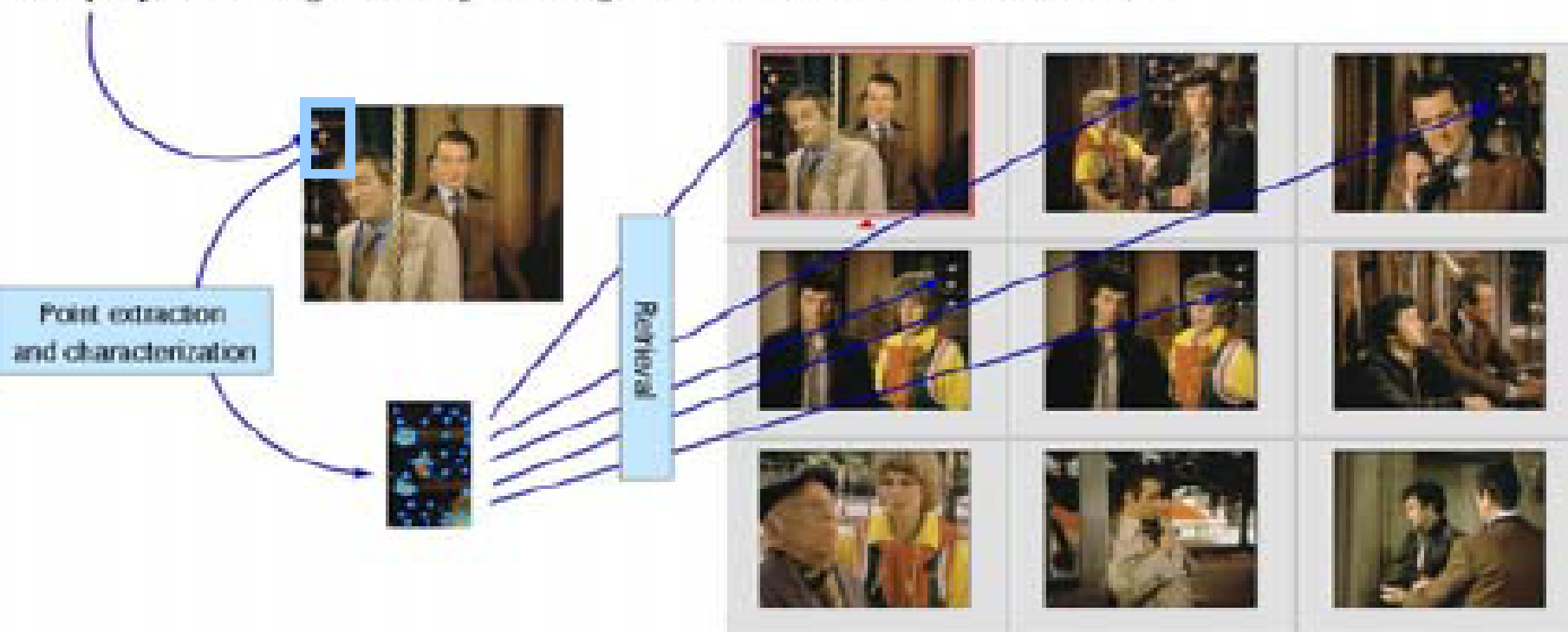
## Invariants différentiels couleur

=> Caractéristiques :

- Invariance aux transformations géométrique/affine, changement d'illumination, (changement d'échelle – images niveau de gris)
- Robuste aux changement de point de vue,

## Recherche précise par description locale

The query : "I am looking for the images involving the room where there is this wine store".



➔ Contrat de recherche avec la Direction Centrale de la Police Judiciaire

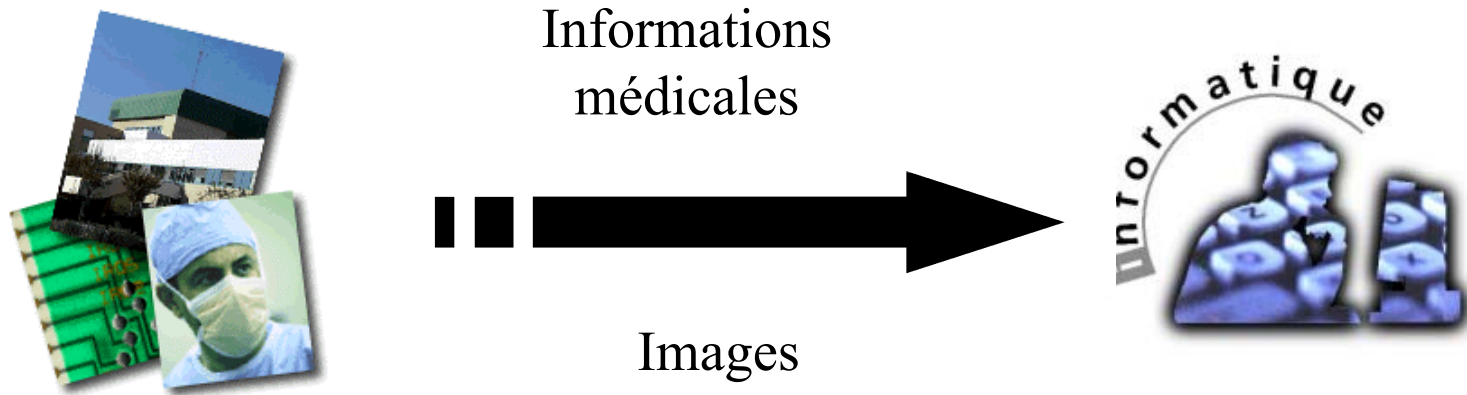


# Exemple de domaine d'application : **images médicales**

- Recherche par similarité d'images médicales
- Format d'image standard : DICOM
- Contenu physique des images médicales
- Système MIMS
- Contenu sémantique des images médicales
- Des images aux bases de caractéristiques
- Exemple d'interface utilisateur
- Aspect évolutif des images médicales

Depuis 2003 : **GDR STIC Santé** regroupant des chercheurs en traitement d'images, en informatique et du domaine médical - <http://stic-sante.org/>

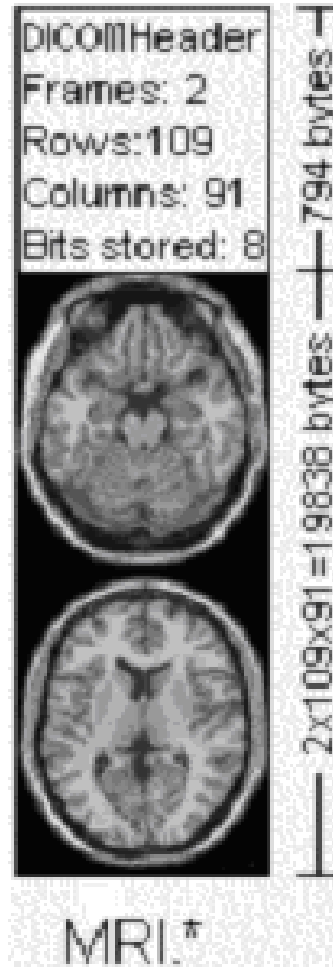
# Recherche par similarité d'images médicales



- Hôpitaux, médecins, INSERM
- Images médicales : IRM, scanner, ultrasonore, échographie, scintigraphie ...
- Standards : PACS, DICOM
- Système de recherche par le contenu d'images/séquences d'images :
  - Aide au diagnostic
  - Prévision

# Format d'image standard : DICOM

*(Digital Imaging and Communications in Medicine)*

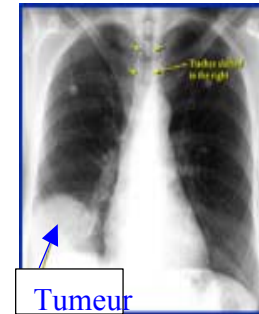


# Contenu physique des images médicales

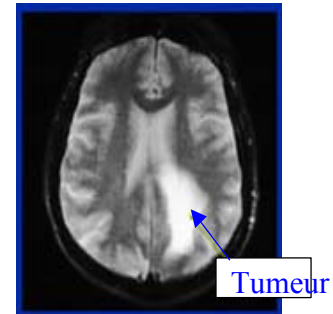
- Information pertinente : existence d'une anomalie ou d'un signe pathologique

**Régions d'intérêt (ROI : *Region Of Interest*)**

- Recherche par régions d'images
  - Limiter les régions : spécialiste
  - Segmentation
  - Extraction des caractéristiques :
    - Forme, taille, périmètre [EF97, CHI+98]
    - Texture [BKD99]
    - Couleur [BCT+02]



*Radiographie  
Poumons*



*MRI Cerveau*

# Contenu physique des images médicales

**Autres Informations pertinentes : relation spatiale entre une anomalie et une région pathologique, ou entre deux anomalies**

- Distance, position relative, montant de recouvrement
- Relations topologiques : couvre, touche, contient, etc
- Relations directionnelles : Nord, est, ouest, etc

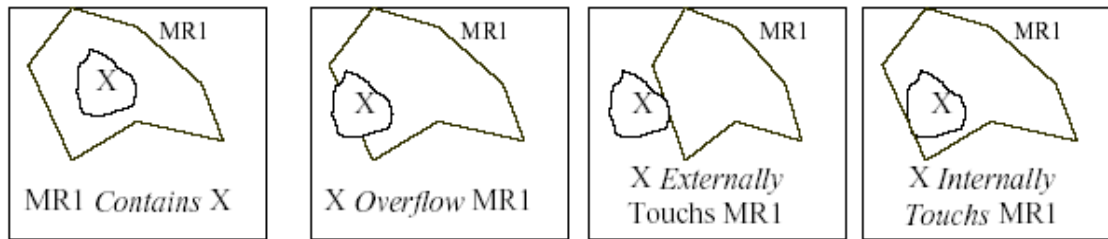


Région pathologique

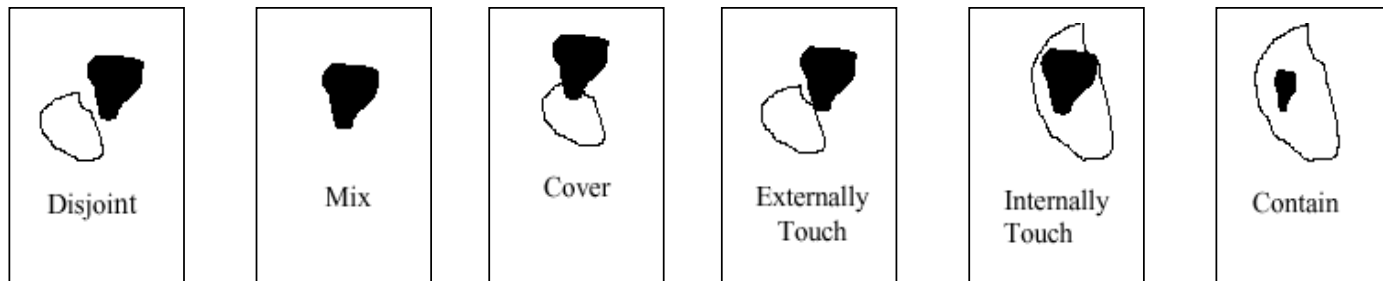
Anomalie

# Relations spatiales dans le système MIMS [CAF00]

- Relations topologiques entre une anomalie et une région pathologique



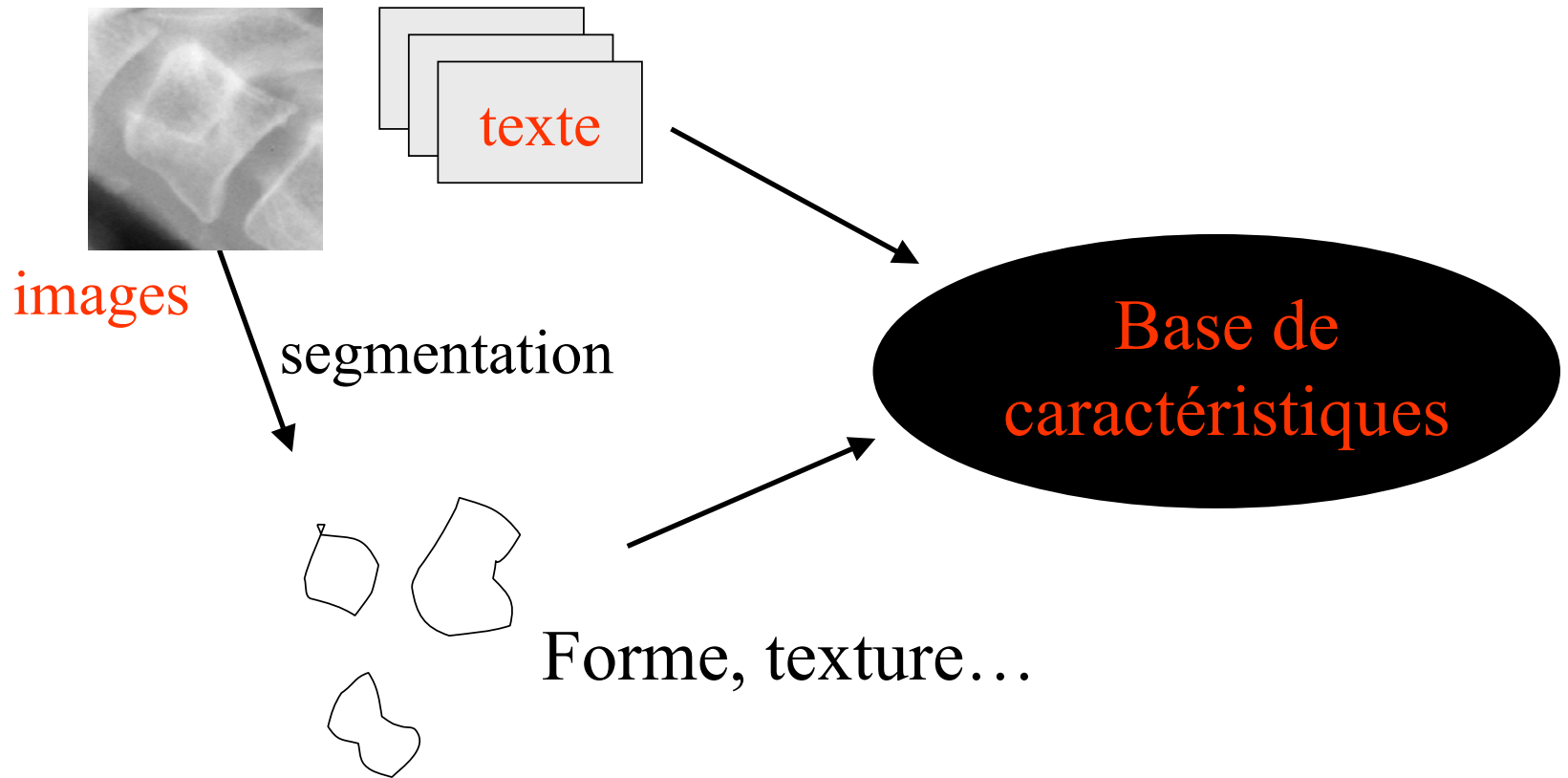
- Relations topologiques entre deux anomalies



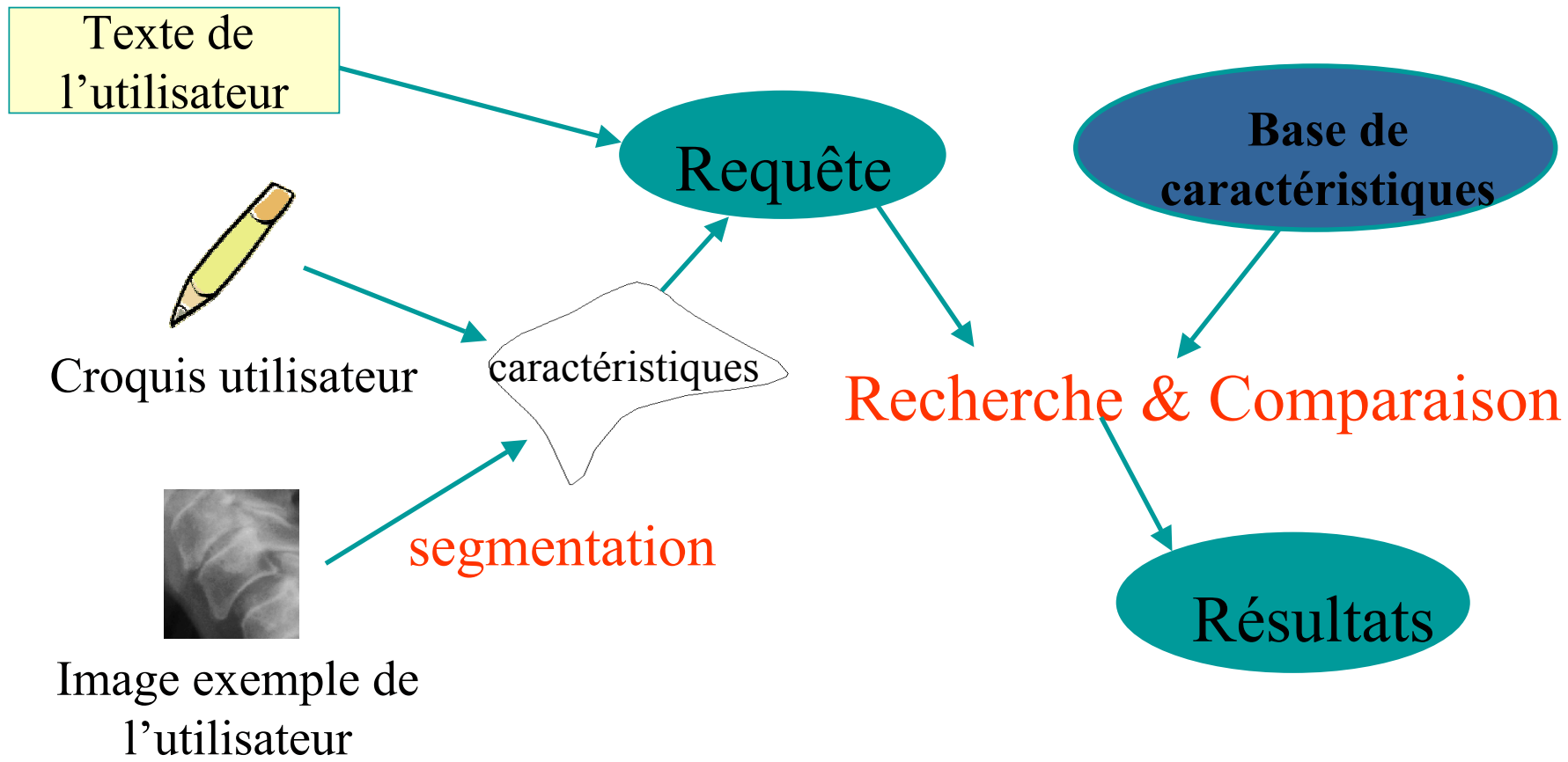
# Contenu sémantique des images médicales

- **Contenu sémantique** : nom de l'organe, nom de l'anomalie
  - **Aussi important que le contenu physique**
  - Requête de type : "*Images de tumeur dans le lobe droit du cerveau*"
  - Classification des images par rapport à leur contenu sémantique
- Définition du contenu sémantique
  - Par un spécialiste du domaine [CAF00, CHI+98]
  - Par la machine : reconnaissance des formes

# Des images aux bases de caractéristiques



# Interrogation et retour des résultats



# Exemple d'interface utilisateur

The screenshot shows a software interface for medical image analysis. It features a central window displaying a sagittal MRI scan of a human head. Two small square overlays are present on the scan: a red one with a white heart icon and a white one. To the left of the scan is a control panel with a tab labeled 'Organ'. Below the tab, there is a dropdown menu for 'Anatomic Organ' set to 'Brain', three small thumbnail icons, a 'Stage' slider, and a 'Show Medical Regions' checkbox. To the right of the scan are 'Execute' and 'Cancel' buttons. Below the scan area, there are three more settings: 'Matching Level' with a numeric input set to 50, 'Anomaly Similarity Level' with a dropdown set to 'Same Anomalies', and 'Spatial Similarity Level' with a dropdown set to 'Same Positionning'. At the bottom, there is a section with 'Ordered By' set to 'Pertinence', a 'View' dropdown set to 'Tile Horizontally', and an 'Abstract' checkbox that is checked.

**Organ**

Anatomic Organ: Brain

Stage: [Slider]

Show Medical Regions: ☐

Execute

Cancel

Matching Level: 50

Anomaly Similarity Level: Same Anomalies

Spatial Similarity Level: Same Positionning

Ordered By: Pertinence

View: Tile Horizontally

Abstract: ☒

Interface d'interrogation dans le système MIMS  
[CAF00]

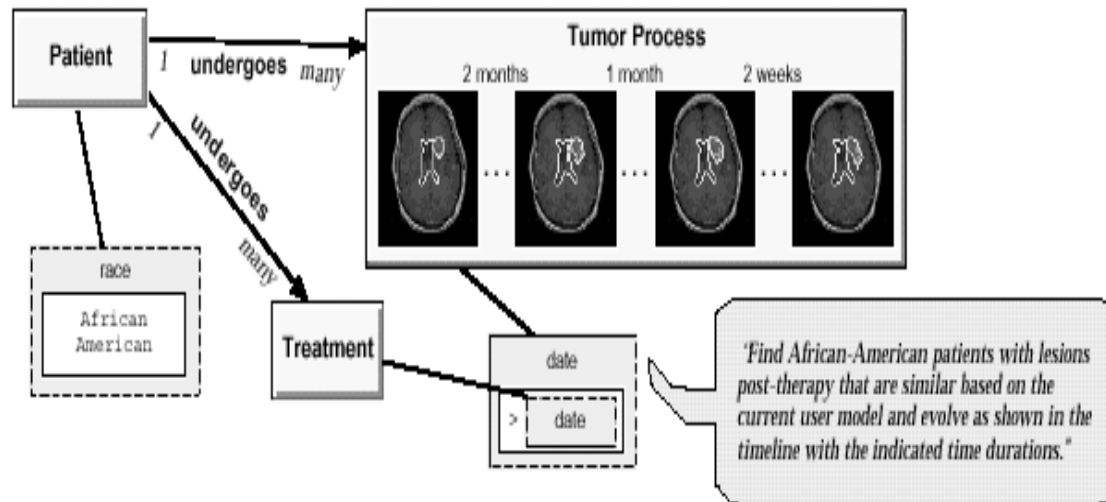
# Aspect évolutif des images médicales

- Suivi de l'évolution des éléments qui changent dans une séquence d'images médicales

Evolution d'une anomalie chez un patient



- Recherche par séquence d'images [CHI+98]



## **6. Conclusions**

- **Résumé de ce qui a été vu**
- **Ce qui n'a pas été vu ...**
- **Recherches actuelles du domaine**
- **Références**

# Résumé

- **Généralité sur l'intégration du multimédia dans les bases de données**
- **Bases de données d'images**
  - ◆ **Caractéristiques des images**
  - ◆ **Interrogation et recherche par similarité**
  - ◆ **Stockage et représentation**
  - ◆ **Exemple de domaine d'application : médical**

# Ce qui n'a pas été vu ...

- Bases de données textuelles / vidéo / audio
- Présentation multimédia
- Bases de données multimédia réparties
  - ◆ Qualité de service
  - ◆ Systèmes de stockage (RAID etc.) et placement des données
- Les standards : MPEG-7 ...  
(*Multimédia Content Description Interface*)  
cf. SIGMOD RECORD 31(2):34-39, June 2002

# Recherches actuelles

- **Entrepôt de données multimédias**
- **Similarité de séquence d'images**
- **Expression de requêtes multimédia**
- **Optimisation de requêtes multimédia**
- **Fouille d'images**
- **Combinaison texte/contenu**
- **Langage d'interrogation**
- **« Résolution » du fossé sémantique ...**

# **Exemple de systèmes en ligne**

- **IKONA (INRIA – équipe Imedia)**
- **BlobWorld (Berkeley)**
- **QBIC (IBM – DB2)**

# Où chercher des références?

- <http://www.bu.dauphine.fr/dauphine/pages/RevueElectroniques/default.htm>

Site de Science Direct d'Elsevier ou de Kluwer Online

- <http://citeseer.nj.nec.com/cs> (avec quelques accès aux articles en ligne)

- <http://liinwww.ira.uka.de/bibliography/index.html>

- <http://www.informatik.uni-trier.de/~ley/db/index.html>

*Pour avoir les  
références  
exactes d'un  
article*

- <http://HPSearch.uni-trier.de/>

*pour trouver les pages web  
concernant un chercheur*

- <http://www.cs.wisc.edu/dbworld/browse.html>

*Pour avoir les adresses des  
sites de conférences et  
workshops récents en BD*

# Exemples d'articles récents

*First International Workshop on Computer Vision meets Databases (CVDB 2004) In cooperation with ACM SIGMOD* June 13, 2004,  
Maison de la Chimie, Paris, France  
*<http://cvdb04.irisa.fr/> (slides online)*

- *Multi-Vector Feature Space Based on Pseudo-Euclidean Space and Oblique Basis for Similarity Searches of Images*  
Yasuo Yamane, Tadashi Hoshiai, Hiroshi Tsuda, Kaoru Katayama, Manabu Ohta, Hiroshi Ishikawa
- *The PIBE Personalizable Image Browsing Engine*  
Ilaria Bartolini, Paolo Ciaccia, Marco Patella
- *A Multimedia Data Base Browsing System*  
Massimiliano Albanese, Carmine Cesarano, Antonio Picariello

# Exemples d'articles récents

## *MDM/KDD2005 - Sixth International Workshop on Multimedia Data Mining "Mining Integrated Media and Complex Data "*

*<http://www-sop.inria.fr/axis/mdm-kdd05/MDM05.htm>*

- *Effective Image and Video Mining: An Overview of Model-Based Approaches* - Missaoui & Palenichka
- *Seeing and Reading Red: Hue and Color-word Correlation in Images and Attendant Text on the WWW* - Newsam
- *A Framework for a Video Analysis Tool for Suspicious Event Detection* - Lavee, Khan, Thuraisingham
- *Analyzing user's behavior on a video database*  
- Mongy, Bouali, Djeraba

# Exemples d'articles récents

**Multimedia Tools and Applications Kluwer, December 2004,  
Volume 24, Issue 3 - *Special Issue: Selected Papers from MIS 2003*  
(Guest Editors: V.S. Subrahmanian and Antonio Picariello)**

- *Ranked Relations: Query Languages and Query Processing Methods for Multimedia*  
Sibel Adali, Corey Bufl, Maria-Luisa Sapino
- *Merging Results for Distributed Content Based Image Retrieval*  
Stefano Berretti, Alberto Del Bimbo, Pietro Pala
- *An Algebra for PowerPoint Sources*  
Marat Fayzullin, V.S. Subrahmanian

# Exemples d'articles récents

**IEEE Transactions on Knowledge and Data Engineering (TKDE),  
March 2005 (Vol. 17, No. 3)**

*Indexing High-Dimensional Data for Efficient In-Memory Similarity  
Search*

Bin Cui, Beng Chin Ooi, Jianwen Su, Kian-Lee Tan

**IEEE Transactions on Knowledge and Data Engineering (TKDE),  
November 2004 (Vol. 16, No. 11)**

■ *Information Retrieval in Document Image Databases*

Yue Lu, Chew Lim Tan

# Exemples d'articles récents

## Workshop on Multimedia Information Systems 2004, College Park, MD

<http://lass.cs.umass.edu/~shenoy/conf/mis04/>

- *Integrating the Results of Multimedia Sub-Queries Using Qualitative Preferences*

Ilaria Bartolini, Paolo Ciaccia, Vincent Oria, M. Tamer Ozsü

- *Improving Image Similarity Search Effectiveness in a Multimedia Content Management System*

Fabrizio Falchi, Claudio Gennaro, Fausto Rabitti, Giuseppe Amato, Pasquale Savino, Peter Stankev

- *A Generic Model for Syntactical and Semantical Description of Images: a Core Component of a framework for image database engineering*

L. Besson, J. Savelli, E. Leclercq, M.N. Terrasse

# Exemples d'articles récents

**International Conference on Information Systems and Databases  
(ISDB 2002) September 25-27, 2002 - Tokyo, Japan**

**<http://www.iasted.org/conferences/2002/japan/isdb.htm>**

- *Extending SQL to Support Image Content-based Retrieval*  
M.R.B. Araujo, A. Traina, and C. Traina, Jr. (Brazil)
- *Cosmos: Convenient Image Retrieval System of Flowers for Mobile Computing Situations*  
M. Noda, H. Sonobe, S. Takagi, et al. (Japan)
- *Clustered R-trees: Improving R-trees for Nearest Neighbor Search*  
Y. Feng and A. Makinouchi (Japan)

# Exemples d'articles récents

**MIS 2002 - International Workshop on Multimedia Information Systems - 10/30/02 - 11/1/02 - Tempe, Arizona**

**<http://www.geometrek.com/mis2002/accepted.htm>**

- *A Multi-Level Index Structure for Video Databases*  
Lei Chen, . Oria and M. Tamer Ozsü
- *Multi-Resolution Similarity Search in Image Databases*  
D. Keim, M. Heczko, A. Hinneburg, M. Wawryniuk
- *A Fuzzy Algebra for Image Data Bases*  
A. Picariello and M.L. Sapino

# Exemples d'articles récents

**8th International Conference on Database Systems for Advanced Applications (DASFAA 2003) - 26 - 28 March, 2003, Kyoto, Japan**

**<http://db-www.aist-nara.ac.jp/dasfaa2003/program.html>**

- *Indexing High-Dimensional Data for Content-Based Retrieval in Large Databases* - Manuel J. Fonseca, Joaquim A. Jorge
- *iSearch: Mining Retrieval History for Content-Based Image Retrieval*  
Hongyu Wang, Beng Chin Ooi, Anthony K. H. Tung

# Exemples d'articles récents

## MDM/KDD2002 Third International Workshop on Multimedia Data Mining

[http://www-staff.it.uts.edu.au/~simeon/mdm\\_kdd2002/](http://www-staff.it.uts.edu.au/~simeon/mdm_kdd2002/)

- *Multimedia data mining using P-trees*  
W. Perrizo, W. Jockheck, A.l Perera, D. Ren, W. Wu, Yi Zhang
- *User concept pattern discovery using relevance feedback and multiple instance learning for content-based image retrieval*  
Xin Huang, Shu-Ching Chen, Mei-Ling Shyu and Chengcui Zhang

# Exemples d'articles récents

## IEEE Transactions on Knowledge and Data Engineering

- *Association and Content-Based Retrieval* - Chabane Djeraba  
Volume 15 - 2003
- *ImageMap: An Image Indexing Method Based on Spatial Similarity* -  
Euripides G. M. Petrakis, Christos Faloutsos, King-Ip Lin  
Volume 14, Number 5, September/October 2002
- *An Object-Oriented Fuzzy Data Model for Similarity Detection in  
Image Databases*  
Arun K. Majumdar, Indrajit Bhattacharya, Amit K. Saha  
Volume 14, Number 5, September/October 2002

# Références

- [BCT+02] J. M. Bueno, F. Chino, A. J. M. Traina, C. Traina, Jr., P. M. Marques, *How to Add Content-based Image Retrieval Capability in a PACS*, IEEE Int. Conf. on Computer Based Medical Systems – 2002
- [BGR99] K.S. Beyer, J. Goldstein, R. Ramakrishnan, U. Shaft. *When Is "Nearest Neighbor" Meaningful?* ICDT 1999: 217-235
- [BKD99] C.E. Brodley, A. Kak, J. Dy, C. Shyu, A. Aisen et L. Broderick. *Content-based retrieval from medical image databases : a synergy of human interaction, machine learning and computer vision*. 16th National Conference on Artificial Intelligence – 1999
- [BKS+90] N. Beckmann, H.-P. Kriegel, R. Schneider, and B. Seeger, "The R\*-tree: An efficient and robust access method for points and rectangles," Proceedings of ACM SIGMOD Int'l. Conf. on Management of Data, pp. 322-331, 1990.
- [CAF00] R. Chbeir, Y. Amghar et A. Flory. *MIMS: A prototype for medical image retrieval*. RIAO - 2000
- [CF98] K.L. Cheung et A.W.C. Fu. *Enhanced Nearest Neighbour Search on the Rtree*. SIGMOD Record 27(3) - 1998
- [CHI+98] W.W. Chu, C.C. Hsu, I.T. Jeong, R.K. Taira. *Content-based image retrieval using metadata and relaxation techniques* . McGraw Hill - 1998 <http://www.kmed.cs.ucla.edu/>
- [CPZ97] P.Ciaccia, M. Patella, P Zezula. *An Efficient Access Method for Similarity Search in Metric Spaces*, VLDB Journal, 1997
- [EF97] G.M. P. Euripides, C. Faloutsos, *Similarity Searching in Medical Image Databases*. IEEE Transactions on Knowledge and Data Engineering, 9 (3) - 1997
- [Gut84] A.Guttman, *R-trees: A dynamic Index Structure for Spatial Searching*. ACM SIGMOD – 1984

# Références

- [KE80] E. Kawagushi et T. Endo. *On a method of binary picture representation and its application to data compression*. IEEE Trans. Pattern Anal. Mach. Intell. – 1980
- [KS97] N. Katayama et S. Satoh. *The SR-tree: An index Structure for High-Dimensional Nearest Neighbor Queries*. ACM SIGMOD - 1997
- [JMR99] G. Jomier, M. Manouvrier et M. Rukoz. *Stockage et Gestion d'Images par un Arbre Quaternaire Générique*. Conf. Bases de Données Avancées - 1999
- [JMR00] G. Jomier, M. Manouvrier et M. Rukoz. *Storage and Management of Similar Images*, Journal of Computer Society (JBCS), 3(6):13-25 - avril 2000
- [JM+04] G. Jomier, M. Manouvrier, V. Oria et M. Rukoz. *Indexation multi-niveau pour la recherche globale et partielle d'images par le contenu* Conf. Bases de Données Avancées – 2002
- [JM+05] G. Jomier, M. Manouvrier, V. Oria et M. Rukoz. *Multi-level index for global and partial content-based image retrieval*. IEEE Int. Workshop EMMA 2005
- [Lin97] T.W. Lin. *Compressed quadtree representation for storing similar images*. Image and Vision Computing 15- 1997
- [LOT94] H. Lu, B-C. Ooi et al. *Efficient Image Retrieval by Color Contents*. Int. Conf. on Applications of Database, Lecture Notes 891 - 1994
- [LTO+01] S. Lin, M. Tamer Özsu, V. Oria, et al. *An Extensible Hash for Multi-Precision Similarity Querying of Image Databases*. Int. Conf. on VLDB, 2001
- [Man00] M. Manouvrier. *Objets similaires de grande taille dans les Bases de Données*. Thèse de Doctorat en Informatique - Univ. Paris IX Dauphine - 2000  
[www.lamsade.dauphine.fr/~manouvri/sujet\\_these.html](http://www.lamsade.dauphine.fr/~manouvri/sujet_these.html)
- [MBNW99] J. Malki, N. Boujemaa et al. *Region Queries without Segmentation for Image Retrieval by Content*. Int. Conf. on Visual Info. Sys. - 1999  
[www-rocq.inria.fr/~awinter/publis.html](http://www-rocq.inria.fr/~awinter/publis.html)

# Références

- [MRJ02] . Manouvrier, M. Rukoz, and G. Jomier. *Quadtree representations for storage and manipulation of clusters of images*. Image and Vision Computing – 2002
- [MRJ05] . Manouvrier, M. Rukoz, and G. Jomier. *A Generalized Metric Distance between Hierarchically Partitioned Images*. MDM-KDD'2005
- [Nai02] Y.Naija, *Indexation des images - Application au domaine médical*. Mémoire de DEA, Université Paris Dauphine - 2002
- [OA04] V. Oria et M-A. Aufaure. *Recherche et Accès aux données Multimédia*. À paraître dans TSI en 2004.
- [OLD04] V. Oria, Y. Li and C. Dorai. *Multimedia Databases: Analysis, Modeling, Querying and Indexing*. À paraître dans Encyclopedia of Computer Science and Engineering, Handbook, 2nd Edition, CRC Press, 2004.
- [OTOL+97] V. Oria, M. Tamer Özsu, et al. *Modeling Images for Content-Based Queries : The DISIMA Approach*. Int. Conf. on Visual Info. Sys. - 1997  
[www.cs.ualberta.ca/~database/multimedia/publications.html](http://www.cs.ualberta.ca/~database/multimedia/publications.html)
- [RMJ02] M.Rukoz, M.Manouvrier et G. Jomier, *Distances de similarité d'images b basées sur les arbres quaternaires*, Conf. Bases de Données Avancées - 2002
- [SWS+00] A.W.M. Smeulders, M. Worring, et al., *Content-Based Image Retrieval at the End of the Early Years*, IEEE Trans. On Pattern analysis and Machine Intel., 22(12) – 2000
- [SRF87] T. Sellis, N. Roussopoulos, and C. Faloutsos, *The R+-tree: A dynamic index for multidimensional objects*, Proc. of the 13th Inter. Conf. on Very Large Databases, 1987
- [Sub98] V.S. Subrahmanian, *Principles of MultiMedia Database Systems*, Morgan Kaufmann – 1998
- [VME93] M. Vassilakopoulos, Y. Manolopoulos et al. *Overlapping Quadtrees for the Representation of Similar Images*. Image and Vision Computing 11(5) – 1993
- [WSB98] Roger Weber, Hans-Jörg Schek, Stephen Blott: *A Quantitative Analysis and Performance Study of Similarity-Search Methods in High-Dimensional Spaces*. VLDB 1998: 194-205

# Références

- *Multimedia and Imaging Database*  
S. Khoshafian et A. B. Baker, Morgan Kaufmann,  
1996 (*disponible à la BU et à la BR*)
- *Principles of Multimedia Database Systems*  
V.S. Subrahmanian, Morgan Kaufmann, 1999 (*disponible à la BR*)
- *Multimedia Database Management Systems*  
G. Lu, Artech House Publishers, 1999
- *Image Databases - Search and Retrieval of Digital Imagery*  
Edité par V. Castelli et L.D. Bergman, Wiley, 2002
- *Modern Information Retrieval*  
R. Baeza-Yates et B. Ribeiro-Neto, Addison Wesley, 1999
- *Bases de données multimédias,*  
M.Adiba, Chapitre 5 dans *Bases de données et Internet*,  
Hermes Sciences, Lavoisier, 2001
- *State-of-the-Art in Content-Based Image and Video Retrieval*  
Edité par R.C. Veltkamp, H. Burkhardt et H-P. Kriegel, Kluwer Academic Pub., 2001
- *Content-Based Image and Video Retrieval*  
O. Marques et B. Furhy, Kluwer Academic Pub., 2002
- *Multimedia Mining : A Highway to Intelligent Multimedia Documents*  
Edité par C. Djeraba , Kluwer Academic Pub., 2003

# *Démo online* de systèmes de recherche d'images par le contenu

- *QBIC*

[www.qbic.almaden.ibm.com](http://www.qbic.almaden.ibm.com)

- *VisualSEEK*

[www.ctr.columbia.edu/VisualSeek](http://www.ctr.columbia.edu/VisualSeek)

- *Virage*

[www.virage.com](http://www.virage.com)

- *IKONA*

<http://www-rocq.inria.fr/cgi-bin/imedia/ikona/exec>

- *Blobword*

<http://elib.cs.berkeley.edu/photos/blobworld/>

# IKONA

## Prototype de l'équipe IMEDIA de l'INRIA



<http://www-rocq.inria.fr/cgi-bin/imedia/ikona/exec>

Ikona Web Demo - Microsoft Internet Explorer

Fichier
Edition
Affichage
Favoris
Outils
?

Précédente



 Rechercher
 Favoris
 Média

Adresse
http://www-rocq.inria.fr/cgi-bin/imedia/ikona/exec
OK
Liens

**IKONA Project** [IMEDIA](#)

Copyright **INRIA** 2001

Project contact : [N. Boujemaa](#)

Demo contact : [J-P. Chièze](#)

Images du Sud (5601 images)

GoodShoot (4494 images)

Big database (3670 images)

3Dobjects (7200 images)

Ground-truth (1332 images)

\* arabidopsis (747 images)

arabidopsis : courtesy of [NASC](#)  
for [biotin project](#).

Switch to Faces DBs

☐ retrieve with feedback

Display 

2 rows

3 cols

Prev Next Shuffle

**Instructions :**

1. Choose databases.
2. Randomly initialize with Shuffle.
3. Click on an image to retrieve by content
4. Click on Feedback to retrieve with feedback

Microsoft PowerP...

Ikona Web Demo ...

FR

262

Ikona Web Demo - Microsoft Internet Explorer

Fichier Edition Affichage Favoris Outils ?

Précédente
 


 Rechercher
 Favoris
 Média

Adresse http://www-rocq.inria.fr/cgi-bin/imedia/ikona/exec OK Liens

[IKONA Project](#) [IMEDIA](#)

Copyright INRIA 2001

Project contact : [N. Boujemaa](#)

Demo contact : [J-P. Chièze](#)

Images du Sud (5601 images)

GoodShoot (4494 images)

Big database (3670 images)

3Dobjects (7200 images)

Ground-truth (1332 images)

\* arabidopsis (747 images)

arabidopsis : courtesy of [NASC](#)  
for [biotin project](#).

Switch to Faces DBs

☐ retrieve with feedback

Display 

2 rows

3 cols

Prev Next Shuffle

---

**Instructions :**

1. Choose databases.
2. Randomly initialize with Shuffle.
3. Click on an image to retrieve by content
4. Click on Feedback to retrieve with feedback

http://www-rocq.inria.fr/imedia/WebDemo/ikona/IDS/food.a\_000275.51.jpg

Microsoft PowerP...

Ikona Web Demo ...

FR

Ikona Web Demo - Microsoft Internet Explorer

FichierEditionAffichageFavorisOutils?

PrécédenteRechercherFavorisMédia

Adressehttp://www-rocq.inria.fr/cgi-bin/imedia/ikona/execOKLiens

**IKONA Project** [IMEDIA](#)

Copyright  **INRIA** 2001

Project contact : [N. Boujemaa](#)

Demo contact : [J-P. Chièze](#)

Images du Sud (5601 images)

GoodShoot (4494 images)

Big database (3670 images)

3Dobjects (7200 images)

Ground-truth (1332 images)

\* arabidopsis (747 images)

arabidopsis : courtesy of [NASC](#)  
for [biotin project](#).

Switch to Faces DBs

☒ retrieve with feedback

Display 2 rows 3 cols

Prev

Next

Shuffle

Feedback



Y?N



Y?N



Y?N



Y?N



Y?N



Y?N

**Instructions :**

1. Choose databases.
2. Randomly initialize with Shuffle.
3. Click on an image to retrieve by content
4. Click on Feedback to retrieve with feedback

Microsoft PowerP...

Ikona Web Demo ...

FR

264

**Ikona Web Demo - Microsoft Internet Explorer**

Fichier Edition Affichage Favoris Outils ?

Précédente Revenir Arrêter Rechercher Favoris Média Imprimer

Adresse <http://www-rocq.inria.fr/cgi-bin/imedia/ikona/exec> OK Liens

[IKONA](#) Project [IMEDIA](#)

Copyright  INRIA 2001

Project contact : [N. Boujemaa](#)

Demo contact : [J-P. Chièze](#)

Images du Sud (5601 images)  
 GoodShoot (4494 images)  
 Big database (3670 images)  
 3Dobjects (7200 images)  
 Ground-truth (1332 images)  
 \* arabidopsis (747 images)

arabidopsis : courtesy of [NASC](#)  
 for [biotin project](#)

Switch to Faces DBs

☒ retrieve with feedback

Display 2 rows 3 cols

Prev Next Shuffle Feedback

**Instructions :**

1. Choose databases.
2. Randomly initialize with Shuffle.
3. Click on an image to retrieve by content
4. Click on Feedback to retrieve with feedback



démarrer Microsoft PowerP... Ikona Web Demo ... FR

Ikona Web Demo - Microsoft Internet Explorer

Fichier Edition Affichage Favoris Outils ?

Précédente

Rechercher

Favoris

Média

Adresse http://www-rocq.inria.fr/cgi-bin/imedia/ikona/exec

IKONA Project IMEDIA

Copyright INRIA 2001

Project contact : N. Boujema

Demo contact : J-P. Chièze

Images du Sud (5601 images)

GoodShoot (4494 images)

Big database (3670 images)

3Dobjects (7200 images)

Ground-truth (1332 images)

\* arabidopsis (747 images)

arabidopsis : courtesy of NASC for biotim project.

Switch to Faces DBs

retrieve with feedback

Display 2 rows 3 cols

Prev

Next

Shuffle

Feedback












○ Y ○ ? ○ N

○ Y ○ ? ○ N

○ Y ○ ? ○ N

○ Y ○ ? ○ N

○ Y ○ ? ○ N

○ Y ○ ? ○ N

Instructions :

1. Choose databases.

2. Randomly initialize with Shuffle.

3. Click on an image to retrieve by content

4. Click on Feedback to retrieve with feedback

Microsoft PowerP...

Ikona Web Demo ...

FR

266

Ikona Web Demo - Microsoft Internet Explorer

Fichier Edition Affichage Favoris Outils ?

Précédente
 


 Rechercher
 Favoris
 Média
 

 OK
 Liens

Adresse http://www-rocq.inria.fr/cgi-bin/imedia/ikona/exec

**IKONA Project** [IMEDIA](#)

Copyright INRIA 2001

Project contact : [N. Boujema](#)

Demo contact : [J-P. Chièze](#)

Images du Sud (5601 images)  
 GoodShoot (4494 images)  
 Big database (3670 images)  
 3Dobjects (7200 images)  
 Ground-truth (1332 images)  
 \* arabidopsis (747 images)

arabidopsis : courtesy of [NASC](#)  
for [biotin project](#).

☐ retrieve with feedback

Display 2 rows 3 cols

Prev Next Shuffle

---

**Instructions :**

1. Choose databases.
2. Randomly initialize with Shuffle.
3. Click on an image to retrieve by content
4. Click on Feedback to retrieve with feedback

<http://www-rocq.inria.fr/imedia/WebDemo/Ikona/iIDS/flore.flo-84-1.li-.jpg>

Microsoft PowerP...

Ikona Web Demo ...

FR

Ikona Web Demo - Microsoft Internet Explorer

FichierEditionAffichageFavorisOutils?

Précédente



 Rechercher
 Favoris
 Média

Adresse http://www-rocq.inria.fr/cgi-bin/imedia/ikona/exec

OKLiens

IKONA Project IMEDIA

Copyright INRIA 2001

Project contact : [N. Boujemaa](#)

Demo contact : [J-P. Chièze](#)

Images du Sud (5601 images)

GoodShoot (4494 images)

Big database (3670 images)

3Dobjects (7200 images)

Ground-truth (1332 images)

\* arabidopsis (747 images)

arabidopsis : courtesy of [NASC](#) for [biotim project](#).

Switch to Faces DBs

☒ retrieve with feedback

Display

2 rows

3 cols

PrevNextShuffleFeedback

Instructions :

1. Choose databases.
2. Randomly initialize with Shuffle.
3. Click on an image to retrieve by content
4. Click on Feedback to retrieve with feedback

☒ Y
☐ ?
☐ N

☐ Y
☒ ?
☐ N

☐ Y
☒ ?
☐ N

☒ Y
☐ ?
☐ N

☐ Y
☒ ?
☐ N

☐ Y
☐ ?
☒ N

Microsoft PowerP...

Ikona Web Demo ...

FR

268

Ikona Web Demo - Microsoft Internet Explorer

Fichier
Edition
Affichage
Favoris
Outils
?

Précédente



 Rechercher
 Favoris
 Média

Adresse
http://www-rocq.inria.fr/cgi-bin/imedia/ikona/exec
OK
Liens

**IKONA Project**
**IMEDIA**

Copyright

INRIA 2001

Project contact :
[N. Boujemaa](#)

Demo contact :
[J-P. Chièze](#)

Images du Sud (5601 images)
GoodShoot (4494 images)
Big database (3670 images)
3Dobjects (7200 images)
Ground-truth (1332 images)
\*arabidopsis (747 images)

arabidopsis : courtesy of [NASC](#)  
for [biotin project](#).

Switch to Faces DBs

☒ retrieve with feedback

Display
2 rows
3 cols

Prev
Next
Shuffle
Feedback

○ Y ○ ? ○ N

○ Y ○ ? ○ N

○ Y ○ ? ○ N

○ Y ○ ? ○ N

○ Y ○ ? ○ N

○ Y ○ ? ○ N

Instructions :

1. Choose databases.
2. Randomly initialize with Shuffle.
3. Click on an image to retrieve by content
4. Click on Feedback to retrieve with feedback

Microsoft PowerP...
Ikona Web Demo ...
FR

269

© Maude Manouvrier - Lamsade

Ikona Web Demo - Microsoft Internet Explorer

Fichier Edition Affichage Favoris Outils ?

Précédente
 


 Rechercher
 Favoris
 Média
 

 OK
 Liens

Adresse http://www-rocq.inria.fr/cgi-bin/imedia/ikona/exec

[IKONA](#) Project [IMEDIA](#)

Copyright INRIA 2001

Project contact : [N. Boujemaa](#)

Demo contact : [J-P. Chièze](#)

Images du Sud (5601 images)

GoodShoot (4494 images)

Big database (3670 images)

3Dobjects (7200 images)

Ground-truth (1332 images)

\* arabidopsis (747 images)

arabidopsis : courtesy of [NASC](#)  
for [biotim project](#).

Switch to Faces DBs

☒ retrieve with feedback

Display 

2 rows

3 cols

Prev Next Shuffle Feedback

Retrieve with feedback

○ Y ○ ? ○ N

○ Y ○ ? ○ N

○ Y ○ ? ○ N

○ Y ○ ? ○ N

○ Y ○ ? ○ N

○ Y ○ ? ○ N

**Instructions :**

1. Choose databases.
2. Randomly initialize with Shuffle.
3. Click on an image to retrieve by content
4. Click on Feedback to retrieve with feedback

démarrer
 


 DEA127
 Microsoft PowerP...
 Ikona Web Demo ...
 FR
 17:08

# BlobWorld

**Blobworld - Microsoft Internet Explorer**

Fichier Edition Affichage Favoris Outils ?

Précédente Recherche Favoris Média

Adresse <http://elib.cs.berkeley.edu/photos/blobworld/> OK Liens

## Blobworld Image retrieval using regions

Blobworld is a system for content-based image retrieval. By automatically segmenting each image into regions which roughly correspond to objects or parts of objects, we allow users to query for photographs based on the objects they contain.

For example, here's a picture of a wolf along with its Blobworld representation:



From this page, you have several options:

- [Try a Blobworld query.](#)
- Take a quick tour of [sample queries.](#)
- Read a [technical paper](#) about Blobworld.

[List of images in the collection](#)

[Sample queries](#)

[Starting images](#)

[Source code](#)

<http://elib.cs.berkeley.edu/photos/blobworld/>

Sample Starting Images for Blobworld - Microsoft Internet Explorer

Fichier Edition Affichage Favoris Outils ?

Précédente

Rechercher

Favoris

Média

Adresse <http://elib.cs.berkeley.edu/photos/blobworld/start.html> OK Liens

Blobworld

Sample starting images

[Blobworld home](#)

[List of images in the collection](#)

[Sample queries](#)

Click on an image to start a Blobworld query using that image, or choose a different sample picture from one of these categories:

[Animals](#)

[People](#)

[Flowers](#)

[Ocean Scenes](#)

[Outdoor Scenes](#)

[Manmade Objects](#)



digital library

[Digital Library Project](#)  
University of California, Berkeley

Microsoft Power...

Blobworld Query ...

Sample Starting ...

FR

272



Keywords (optional):

## Step 2:

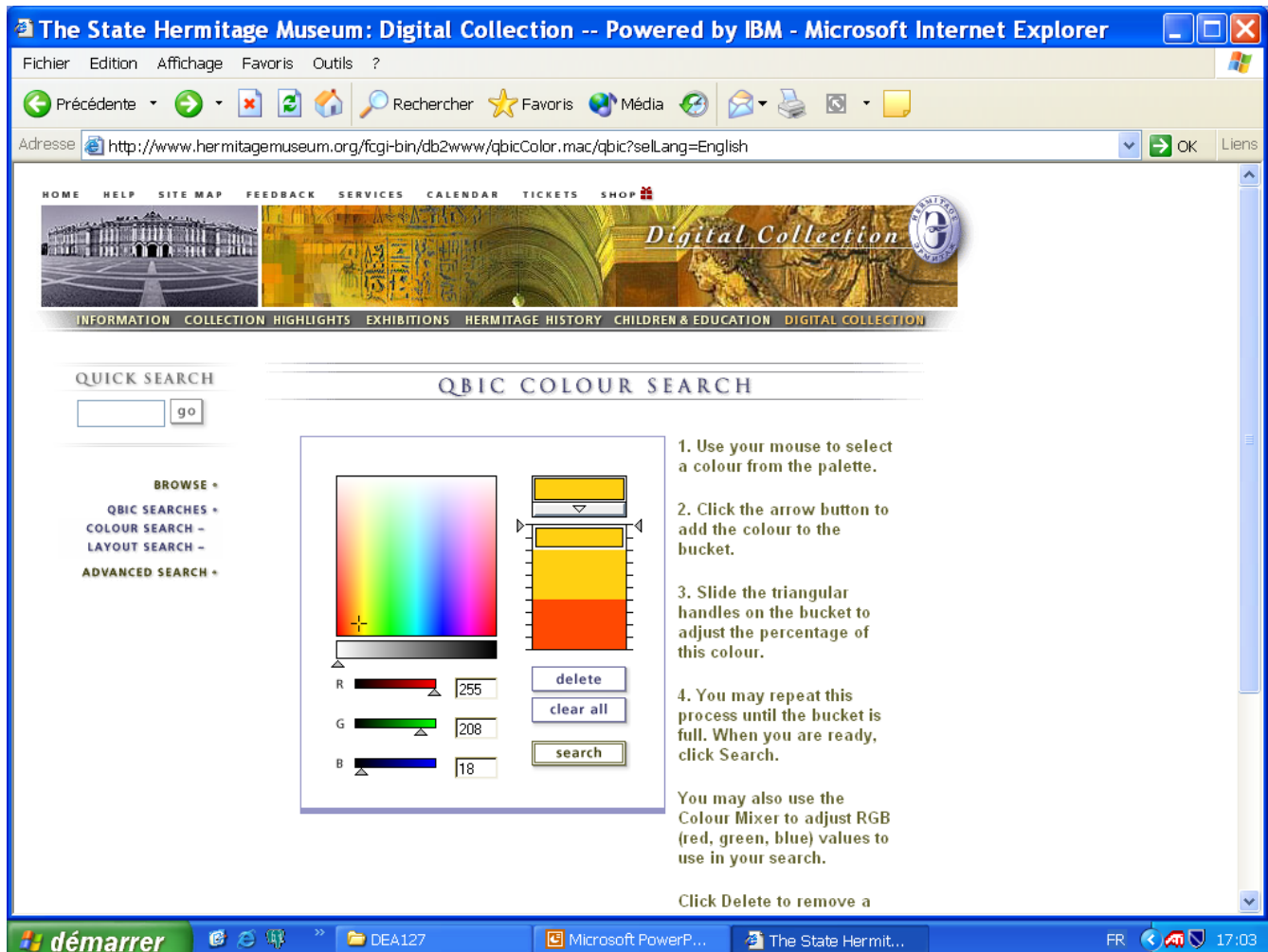
Adjust the weights below if you'd like, then click "Submit."

|                                                | Not Somewhat Very                |                                  |                                  |                                                                  | Not Somewhat Very     |                                  |                       |
|------------------------------------------------|----------------------------------|----------------------------------|----------------------------------|------------------------------------------------------------------|-----------------------|----------------------------------|-----------------------|
| How important is the selected region?          | <input type="radio"/>            | <input type="radio"/>            | <input checked="" type="radio"/> | How important is the background (everything outside the region)? | <input type="radio"/> | <input checked="" type="radio"/> | <input type="radio"/> |
| How important are the features of this region? |                                  |                                  |                                  |                                                                  |                       |                                  |                       |
| Color                                          | <input type="radio"/>            | <input type="radio"/>            | <input checked="" type="radio"/> |                                                                  |                       |                                  |                       |
| Texture                                        | <input type="radio"/>            | <input checked="" type="radio"/> | <input type="radio"/>            |                                                                  |                       |                                  |                       |
| Location                                       | <input checked="" type="radio"/> | <input type="radio"/>            | <input type="radio"/>            |                                                                  |                       |                                  |                       |
| Shape/Size                                     | <input checked="" type="radio"/> | <input type="radio"/>            | <input type="radio"/>            |                                                                  |                       |                                  |                       |

# BlobWorld

- Résultat de la requête sur la couleur.
- Résultat de la requête sur la texture.
- Résultat de la requête sur la couleur et la texture.
- Résultat de la requête sur la forme.
- Résultat sur la couleur/texture/forme/localisation.

# QBIC



<http://www.qbic.almaden.ibm.com/>



[INFORMATION](#) [COLLECTION HIGHLIGHTS](#) [EXHIBITIONS](#) [HERMITAGE HISTORY](#) [CHILDREN & EDUCATION](#) [DIGITAL COLLECTION](#)

## SEARCH RESULTS

[modify search](#)[new search](#)

## QBIC SEARCHES •

COLOUR SEARCH –

LAYOUT SEARCH -

**ADVANCED SEARCH** \*



1) Vase of Flowers

Huysum, Jan van  
Early 18th century



## 2) Avenue in a Park

Watteau, Antoine  
Circa 1715



### 3) Bahram Gur in the Red Pavilion

UNKNOWN  
1431



#### 4) The Triumph of Religion in Art

Overbeck, Friedrich  
Johann Between  
1829 and 1840



### 5) The Nativity of the Virgin

## REFERENCES



6) **St Peter**

UNKNOWN

The State Hermitage Museum: Digital Collection -- Powered by IBM - Microsoft Internet Explorer

Fichier Edition Affichage Favoris Outils ?

Précédente

Rechercher

Favoris

Média

Adresse
http://www.hermitagemuseum.org/fcgi-bin/db2www/qbic.mac/qbic?sellLang=English&selCateg=picture&check=false&comeFrom=simil&Quer
OK
Liens

HOME HELP SITE MAP FEEDBACK SERVICES CALENDAR TICKETS SHOP

Digital Collection

INFORMATION COLLECTION HIGHLIGHTS EXHIBITIONS HERMITAGE HISTORY CHILDREN & EDUCATION DIGITAL COLLECTION

QUICK SEARCH

go

BROWSE

QBIC SEARCHES

COLOUR SEARCH

LAYOUT SEARCH

ADVANCED SEARCH

1) Avenue in a Park  
Watteau,  
Antoine Circa  
1715

2) Vase of Flowers  
Huysum, Jan van  
Early 18th century

3) Portrait of S. I. Shchukin  
Krohn, Christian  
Cornelius (Xan)  
1915

4) Flower Study  
Mahdi, Muhammad  
Mid-18th century

5) Bahram Gur in the Red Pavilion  
UNKNOWN  
1431

6) The Wedding of Nicholas II and Grand Princess Alexandra Fyodorovna

Microsoft PowerP...

The State Hermit...

FR

© Maude Manouvrier - Lamsade

277



QUICK SEARCH

BROWSE

QBIC SEARCHES

COLOUR SEARCH

LAYOUT SEARCH

ADVANCED SEARCH

SEARCH RESULTS



1) [Portrait of David A. Delyanov \(1761-1833\)](#)

Dawe, George  
No later than 1825



2) [Portrait of Mikhail N. Ryleyev \(1771-1831\) \(1st\)](#)

Dawe, George 1824



3) [Portrait of Ivan L. Shakhovskoy \(1776-1860\)](#)

Dawe, George  
No later than 1825



4) [Portrait of Dmitry M. Yuzefovich \(1777-1821\)](#)

Dawe, George 1822



5) [Stained Glass Panel: Antichrist Making a Fiery Storm \(10a\)](#)



6) [Still Life with a Skull](#)

Bonnecroy