الجمهورية الجزائرية الديمقراطية الشعبية

République Algérienne Démocratique et Populaire

Ministère de l'enseignement supérieur et de la recherche scientifique

Université 8 Mai 1945 - Guelma -

Faculté des Mathématiques d'Informatique et des Sciences de la matière

Département d'Informatique

Mémoire de Fin d'études Master

Filière : Informatique

Option : Informatique Académique

Thème :

Intégration d'une application d'indexation dans un environnement cloud computing

Encadré Par :

Dr.KOUAHLA Zineddine

Présenté par :

ALLALA Nourelhouda

MOUAS Zeyneb

Juin 2017

Résumé

Dans des nombreuses activités humaines, les images numériques constituent une source d'informations très expressive qui jouent un rôle très important. Par conséquence, avoir des volumes croissants des bases d'images est un résultat naturel. Cependant ce volume d'images n'a aucun intérêt s'il on ne pouvait pas le retrouver facilement. Ce rapport a pour objectif principale de présenter une solution efficace et performante pour la recherche des données complexe (application sur les images). Ce travail consiste à proposer un système d'indexation parallèle qui permit la recherche dans une grande collection de données disant des centaines de dimension et des centaines de milliers d'images, utiliser les connaissances tirées de cette étude et construire un prototype d'un système de recherche d'images par le contenu (CBIR). Nous avons parlés dans ce rapport sur la version d'un index séquentiel qui déjà fait l'année passer et nous avons proposés une autre version parallèle complémentaire. En plus, nous avons lancé plusieurs simulations pour valider les résultats de ce moteur avec l'utilisation des données réel (descripteur MPEG-7) de la base CoPhIR des images Flicker stocké sur internet.

REMERCIEMENT

Au terme de ce mémoire nous tenons à exprimer nos

remerciements et notre profonde gratitude avant tout au

bon

DIEU qui nous donne le courage et la force pour mener

bien ce modeste travail.

Nous tenons à remercier au premier lieu

À notre enseignant encadreurMr Le Docteur

KOUAHLA ZINEDDINE

pour son soutien et son louable effort.

Dédicace

A Mes Très Chers Parents

Aucune dédicace ne saurait exprimer mon respect, mon amour

éternel et ma considération pour les sacrifices que vous avez consenti pour mon

instruction et mon bien être.

Je vous remercie pour tout le soutien et l'amour que vous me portez

depuis mon enfance et j'espère que votre bénédiction m'accompagne toujours.

Puisse Dieu, le Très Haut, vous accorder santé, bonheur et logue vie et faire en sorte que jamais je ne vous déçoive.Avec toute ma tendresse.

Mes chers frères : Moured et sa femme mouna et ses deux fils Fahed et baraa, Youssef,Mohamed, Abderahmane

Je prie pour qu'Allah vous protège.

Mes chères cousines : premièrement à mes sœurs « Amina,Hanane »

« Nawel,Bouchra,Nihed,Samia,Mouna,Merieme,Amina,Nassima,Wahida»

A tous les membres de ma grande famille...

A ma chère binôme Nourelhouda : Meilleurs vœux de succès dans ta vie

A mes chères ami(e)s : « Kahina,Loubna,Hana,Imene,Sara,Rym,Touta,Bichou»

A mes chers collègues : « ma promotion de l'année 2016/2017 »

A tous mes enseignants que j'ai l'honneur de

rencontrer tout au long de mes études au département Informatique à Guelma.

Zeyneb

Dédicace

Je dédie ce mémoire a :

Ma mère ma source de tendresse pour toujours.

Mon père pour son bien violence.

Les mots ne sont jamais forts pour leur exprimer ma gratitude.

Mes sœurs HANENE, ILHEM et surtout ma sœur AHLEM que je la souhaite la réussite à l'examen de BAC.

Mon grand frère ABDERREZAK et sa femme ASIA que je le souhaite tout le bonheur du monde

Je prie pour qu'Allah vous protège.

A tous les membres de ma grande famille...

Ma très belle binôme : ZEYNEB MOUAS et sa famille.

Mes amies SARA KAMOUCHE, SARA BOUSSEDIRA, RYMOUCHA et MARWA.

Ma promotion informatique 2016/2017 Surtout TOUTA et BICHOUTA.

A tous mes enseignants que j'ai l'honneur de rencontrer tout au long de mes études au département Informatique à Guelma.

Nourelhouda

Sommaire

Table des figures

Table des tableaux

Introduction générale

Grace aux avancées récents de la technologie ces dernières années, en particulier dans le domaine de données multimédia, l'information numérique est devenu le cœur de tous les secteurs d'activités, dans le monde industriel, médicale, scientifique, juridique, géographique, etc.

Ces progrès se sont accompagnés d'une baisse des couts des équipements informatiques qui a facilité la diffusion et l'échange de ce type de données vers le grand public.

Cette masse de donnée n'aurait aucun intérêt si l'on ne pouvait pas facilement les retrouver.

Cela suscité un besoin en développement de nouvelles techniques de recherche d'information dans les grandes collections, et en particulier dans le domaine des images.

L'indexation et la recherche d'images par le contenu est une piste prometteuse. Elle offre la possibilité aux utilisateurs d'accéder, d'interroger et d'exploiter directement ces bases d'images en utilisant leur contenu ; ceci explique l'activité de recherche consacrée à ce domaine.

L'idée de faciliter l'accès à des données n'est pas neuve, des techniques de recherche d'images ayant été développées à cet effet depuis la fin des années 70.

Parmi ces approches on trouve la technique de recherche d'images à base de texte connue sous le nom « Text-based Image Retrieval » ou TBIR qui est l'approche la plus ancienne utilisée jusqu'à nos jours. Il s'agit d'annoter manuellement chaque image par un ensemble de mots-clés décrivant leur contenu, puis d'utiliser un système de gestion de base de données pour gérer ces images. A travers des descripteurs textuels, les images peuvent être organisées hiérarchiquement selon les thèmes ou les sémantiques afin de faciliter la navigation et la recherche dans la base.

Ces dernières années avec l'explosion de la quantité d'informations, on parle de millions voir des trillions images sur le web.

Les algorithmes parallèles et distribuer sont venu pour facilité les algorithmes de recherche dans le domaine informatique.

Les images sont des données souvent utilisé dans les moteur de recherche, leurs complexité est très grandes ce qui pose un problème de complexité des algorithmes de construction de l'index et les algorithmes de recherche associées.

Le CBIR a fait son apparition au début des années 90. Il a un rôle d'aide à la recherche automatique et à la décision. Cette approche, à laquelle nous nous intéressons dans ce mémoire, consiste à représenter chaque image par un ensemble de caractéristiques visuelles de bas niveau telle que la couleur, la texture et la forme. Ces caractéristiques visuelles, calculées de manière automatique, sont ensuite exploitées par le système pour comparerait retrouver des images. Ces dans ce contexte restreint de la recherche d'images par le contenu que se situera notre travail.

Notre manuscrit comportera les parties suivantes :

- La première partie, expose un état de l'art qui composé en deux chapitres : « Recherche d'information et indexation avec la recherche d'images par le contenu »et « les algorithmes parallèles et distribués ». Dans la première partie concerne l'approche informatique étudiée, appliquée et quelques systèmes de recherche d'images déjà existants à l'heure actuelle. Puis, il décrit le principe de la recherche d'images par le contenu et les différents types de requêtes. Ensuite, il énonce les différentes méthodes d'extraction des descripteurs utilisés dans la recherche CBIR. Ces descripteurs permettent d'extraire des informations pertinentes, en vue d'une exploitation efficace des bases d'images. La deuxième présente quelques notions sur les algorithmes parallèle, à savoir les architectures parallèles, ensuite nous allons à présent introduire les principaux critères d'évaluation pour mesurer les performances des programmes parallèles.
- La deuxième partie est composée de deux sections : Conception et Implémentation. Nous présentons dans la première section le système de recherche d'image par le contenu version parallèle (CBIR v2.0), et dans la deuxième section nous donnons un cas particulier de l'implémentation.

Enfin nous avons faire une conclusion qui donne une idée générale sur notre travail et

sur quoi fonctionner.

Chapitre N° 01

Recherche d'information
et
indexation dans une base de données image

1. Introduction

Dans le premier chapitre nous présentons un bref état de l'art concernant notre travail, nous commençons d'abord par le domaine de la recherche d'information et les systèmes de recherche par le contenu. Ensuite nous allons parler sur ensuite, nous allons présentés les différentes descripteurs d'images (couleur, forme, texture, points des intérêts). A la fin, nous parlons sur l'indexation et leurs structures.

2. La Recherche d'information

2.1. Définition

Le domaine de la recherche d'information est un ensemble d'outils et techniques qui permettent de trouver les documents (texte, image, vidéo, site web…….) contenant l'information pertinente à un besoin. L'objectif de ce domaine est pour fournir à un utilisateur un accès facile à l'information qui l'intéresse, cette information située dans une masse de documents, un Système de Recherche d'Information doit représenter, stocker et organiser l'information, puis fournir à l'utilisateur l'information exprimée par sa requête [1].

2.2. Eléments clés en RI

Les principaux acteurs de la recherche d'information sont [1]:

Une requête qui exprimé par un utilisateur (un besoin en information). Une masse de documents.

Un système de recherche d'information (SRI) qui est l'interface entre l'utilisateur et la masse de documents.

- La requête

La requête est une représentation possible du besoin (Le besoin d'information est une expression mentale d'un utilisateur), Elle peut être exprimée par un mot clé ou une liste de mots clés incluant des opérateurs logiques ou d'autres types d'opérateurs ou bien par image [1].

- Le document

Un document c'est toute unité qui peut constituer une réponse à une requête d'utilisateur, Ce document peut se présenter sous plusieurs formats soit par un texte ou par une image ou bien une page web ou soit par vidéo ... .etc [1].

2.3. Le système de recherche d'information

2.3.1. Définition

Un Système de Recherche d'Informations (SRI) est un système informatique qui permet de retourner à partir d'un ensemble de documents, ceux dont le contenu correspond le mieux à un besoin en informations d'un utilisateur, exprimé à l'aide d'une requête, Un SRI inclut un ensemble de procédures et d'opérations qui
permettent la gestion, le stockage, l'interrogation, la recherche, la sélection et la représentation de cette masse d'informations [2].

2.3.2. Les processus de système de recherche d'information Le

schéma représente les différentes étapes du processus de RI

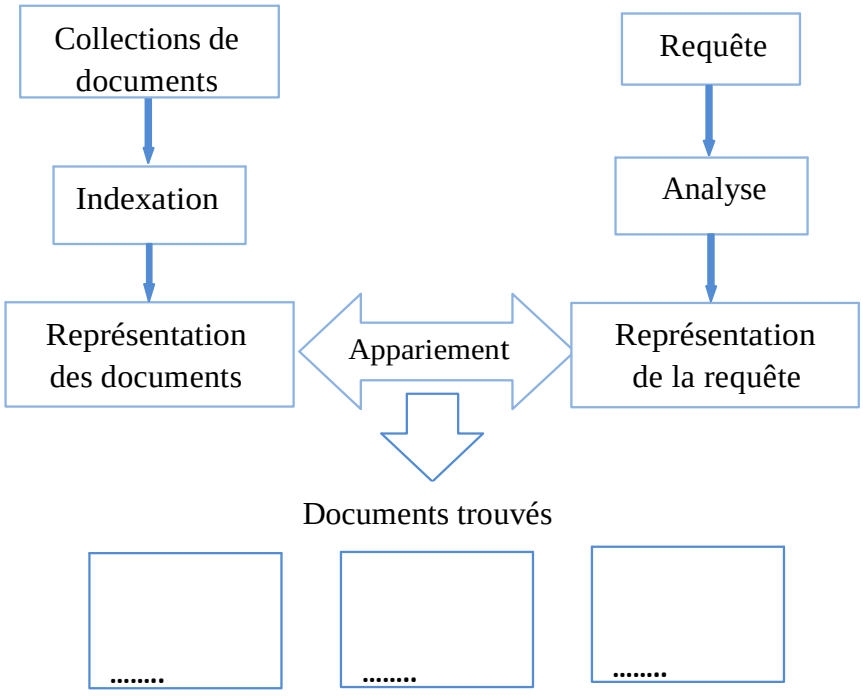

Figure 1.1 Processus de la recherche d'information [2]. les notions de documents et de requêtes qui sont des conteneurs d'informations.

les opérations d'analyse, d'indexation et d'appariement qui permettent

globalement de traiter la requête dans le but de sélectionner des documents à présenter à l'utilisateur.

3. Les systèmes de recherche d'image par le contenu

3.1. Définition

La recherche d'image par le contenu (CBIR) est une technique permettant de rechercher des images à partir de ses caractéristiques visuelles [3] (c'est à dire à partir des données de l'image elles même). Les CBIR permettent de retrouver au sein d'une base d'images les images les plus similaires visuellement à une image requête donnée en fonction de leurs caractéristiques visuelles (la couleur, la texture, la forme……..).

Le principe général de la recherche d'image par le contenu se déroule en deux phases. Lors d'une première le système décrit le contenu des images. Lors de la seconde phase, l'utilisateur interroge la base à l'aide de la requête. Le système recherche les images de la base qui corresponde à la demande de l'utilisateur. Elle est constituée de deux directions selon leurs niveaux de représentation du contenu des images

- Le niveau symbolique : la recherche est effectuée selon une similarité visuelle sur des traits de bas niveau (couleurs, textures, forme) [4].
- Le niveau sémantique : l'indexation et la recherche sont fondées sur une interprétation sémantique du contenu de l'image [4].

3.2. Architecture de système de recherche d'image par le contenu(CBIR)

Le système de recherche d'image par le contenu s'exécute en deux étapes l'étape d'indexation et l'étape de recherche, La première concerne le mode de représentation informatique des images et le second concerne l'utilisation de cette représentation dans le but de la recherche.

- Indexation : une étape de la caractérisation au les attribues sont automatiquement extraites à partir de l'image et stockées dans un vecteur numérique appelé descripteur visuel. Grâce aux techniques de la base de données, on peut stocker ces caractéristiques et les récupérer rapidement et efficacement [5].

- Recherche : le système prend une ou des requêtes à l'utilisateur et lui donne le résultat correspond à une liste d'images ordonnées en fonction de la similarité entre leur descripteur visuel et celui de l'image requête en utilisant une mesure de distance [5].

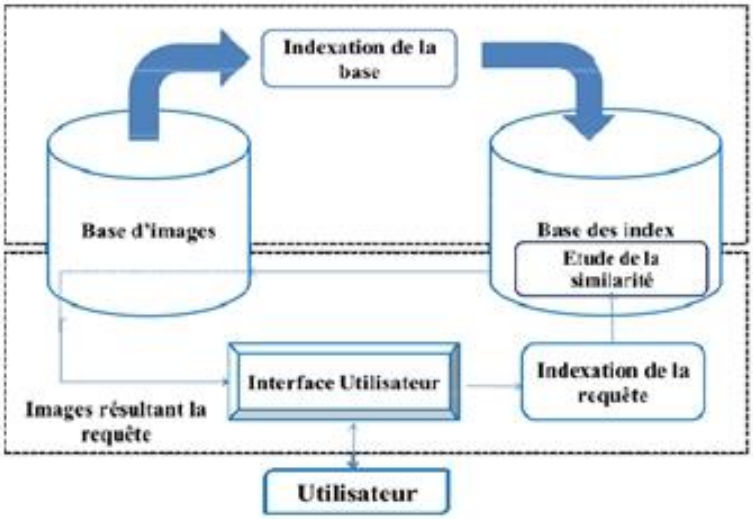

Figure 1.2 Architecture de système de recherche d'image par le contenu.

3.3. Quelques systèmes de recherche d'image par le contenu

Ces dernières années, de nombreux systèmes d'indexation et de recherche d'images par le contenu, ont vu le jour. La plupart de ces systèmes, permettent de naviguer au sein de la base d'images.

Voici le tableau suivant qui représente quelque systèmes de recherche d'images avec leurs explications [6] :

Moteur de recherche	L'année	Explication
Le système QBIC (Query By Image Content)	1993	c'est le premier système de recherche conçu par IBM, Ce système permet de créer une requête avec les formes et les couleurs des objets. Les résultats sont affichés en ordre décroissant de pertinence.

Le système CHABOT	1995	propose de stocker les histogrammes de couleur et le texte associé dans une base de données relationnelle. Il permet de réaliser des recherches .Ces recherches sont exprimées sous la forme de requêtes dans la base relationnelle. Il ne fusionne donc pas réellement les informations textuelles et visuelles.
Le système PHOTOBOOK	1996	est un système d'indexation d'images développer par MIT (Media Laboratory).Ce système se base sur la couleur, la texture et la forme pour définir les signatures d'une image.
Le système MARS (MultimediaAnalysis and Retrieval System)	1997	système qui implique plusieurs domaines de recherche : traitement d'image, gestion de base de données et recherche d'information. En plus, est un des premiers systèmes de recherche d'images par le contenu à utiliser le bouclage de pertinence.
Le système IMAGEROVER	1998	a pour objectif de combiner les informations textuelles d'une page web avec le contenu visuel des images. Dans ce système, chaque image est indexée par un vecteur global concaténant les vecteurs visuels (réduits par ACP) et textuels (réduits dans l'espace latent).

Le système NETRA	1999	est un système de recherche d'image utilise une segmentation pour calculer les attributs de couleur, texture et forme.
Le système PICHUNTER	2000	est un système qui utilise les probabilités bayésiennes pour prédire quel est l'objectif des utilisateurs en fonction de leurs actions. Le système utilise des annotations «cachées» pour améliorer les résultats. Le principe est que le texte associé aux images est utilisé pour améliorer les résultats renvoyés, mais il n'est pas proposé à l'utilisateur de faire des recherches textuelles.
Le système SIMPLICITY (Semantics-sensitive IntegratedMatching for Picture Libraries)	2001	Développer par JAMES WANG, la recherche se fait par une requête exemple, le résultat est affiché par ordre décroissant de similarité. Ce système n'utilise pas d'information textuelle. Par contre, il possède un module appelé A-LIP (voir ci-après) qui permet de faire de l'auto-annotation d'images.
Le système IKONA	2002	est un système de recherche et navigation interactive dans de grandes bases de données multimédia développées par l'équipe IMEDIA de l'INRIA est basé sur une architecture client/serveur. Ce système propose de rechercher des images par similarité visuelle (couleur, texture, forme).

Le système A-LIP (AutomaticLinguisticIndexing of Pictures)	2003	est un module du système SIMPLICITY. Il propose d'annoter les images à l'aide de modèles de Markov cachés. A partir des descripteurs des images d'apprentissage correspondant à un certain concept, le système construit des modèles des images sur plusieurs niveaux de résolution.
Le système RETIN	2004	Conçu par JEROME FOURNIER. C'est un système destinée uniquement à démontrer une nouvelle approche de l'indexation et de la recherche par similarité incluant un retour de pertinence.

Tableau 1.1 Quelque moteur de recherche d'images.

D'autre part, nous avons parlés sur les autres moteurs de recherche actuels dans le tableau suivant [7] :

Moteur de recherche	L'année	Explication
Google Images	2001	est un service proposé depuis 2001 par le moteur de recherche Google pour permettre de trouver sur le web des images en rapport avec un sujet donné.
www.picsearch.com	2006	Picsearch effectue sa recherche parmi 2 milliards d'images en quelques secondes. Efficace et rapide, il constitue une alternative à Google Images.

www.xcavator.net	2007	Xcavator.net est un portail de recherche de photos basé sur la technologie de recherche visuelle.Xcavator.net fournit une recherche interactive naturelle et intuitive pour la photographie stock fournissant aux acheteurs une expérience de navigation basée sur le contenu visuel et les mots clés.
Live Search Images	2008	Le moteur Live Search 2008 permet depuis hier une fonctionnalité intéressante sur sa recherche d'images avec la possibilité de trouver des images proches ou similaires grâce au choix "Show Similar Images" qui apparait lorsqu'on passe la souris sur une image proposée dans les résultats. Un assortiment d'images proches (au niveau de la forme, de la couleur, etc.) est alors proposé. A noter que seul le site américain de Live propose actuellement cette fonction.
Picitup Incogna image search	2011	Est un service de recherche visuelle et un fournisseur de gestion de données visuelles. Le serveur permet aux utilisateurs de rechercher par similarité et des fonctionnalités exactes et effectue également une extraction automatique des couleurs et une création automatique de catalogues visuels. est un moteur de recherche d'images qui organise ses images par contenu; Lorsque vous cliquez sur une image, elle trouvera des images similaires.

http://spffy.com	--	Le fait de proposer un outil qui calcule le tarif de l'image en fonction de sa taille en fait un moteur plutôt réservé à un usage professionnel. Il explore le contenu de Yahoo!, de Flickr et les agences Getty Images et Corbis. Mais un filtre permet de ne remonter que celles qui sont libres de droits.
www.oskope.com	--	Oskope inaugure le shopping visuel d'eBay et d'Amazon, mais aussi de You Tube et de Flickr. Outre la polyvalence, on apprécie le choix dans le mode de présentation des vignettes.
Everypixel	2017	est un nouveau moteur de recherche d'images qui vient de voir le jour, il va vous permettre en quelques clics de trouver la ressource dont vous avez besoin pour votre création. Ce moteur permet de faire des recherches parmi les plus célèbres banques d'images payantes, afin de trouver la plus belle image, il est moteur très puissant et excellent.

Tableau 1.2 Les moteurs de recherche d'images actuels.

3.4. Type de requête

Le traitement d'une requête dans un système de recherche de la manière dont on lui présente l'information. Un modèle de recherche spécifie le mode de représentation de la requête. Il existe 3 façons de faire une requête dans un système d'indexation et recherche des images :

- Requête par mots clés :

Les images sont recherchées suivant un ou plusieurs critères, par exemple trouver les images contenant 80% de rouge. Donc, le système se base sur l'annotation manuelle et textuelle d'images [4].

- Requête par l'exemple :

Dans ce cas le système a besoin de comparer un exemple de même type (image) avec la base pour produire les documents similaires. Cette méthode est simple naturelle et ne nécessite pas de connaissance approfondies pour manipuler le système. Elle est donc bien adaptée à un utilisateur non spécialiste [5].

4. Recherche d'information dans les images

La performance du système de recherche d'images dépend notamment de l'indexation des images qui doit permettre de retrouver la sémantique associée à l'image, du modèle de représentation qui doit être efficace et de la mesure de similarité qui doit permettre de retrouver les documents pertinents [8].

4.1. Descripteur des images

Le but de l'indexation est de fournir une représentation image permettant des recherches efficaces. Il se concentre sur l'information qui permet de traduire efficacement une similarité proche des besoins exprimés par un utilisateur. Une des clés de l'indexation efficace est l'extraction des caractéristiques primaires en accord avec le type et le but des recherches visées par le système. L'analyse faite se focalise généralement autour des attributs de bas niveau tel que la couleur, la texture et la forme. L'extraction de ces attributs constitue le premier pas de toutes les procédures d'analyse d'images [3].

4.1.1. Descripteur de Couleur

La couleur est l'information visuelle la plus utilisée dans les systèmes de recherche par le contenu. Ces valeurs tridimensionnelles font que son potentiel discriminatoire soit supérieur à la valeur en niveaux de gris des images. Avant de sélectionner le descripteur de couleur approprié, la couleur doit être déterminée d'abord [3].

- Histogramme :

Un histogramme est un graphique statistique permettant de représenter la distribution des intensités des pixels d'une image, c'est-à-dire le nombre de pixels pour chaque intensité lumineuse comme l'exemple suivant :

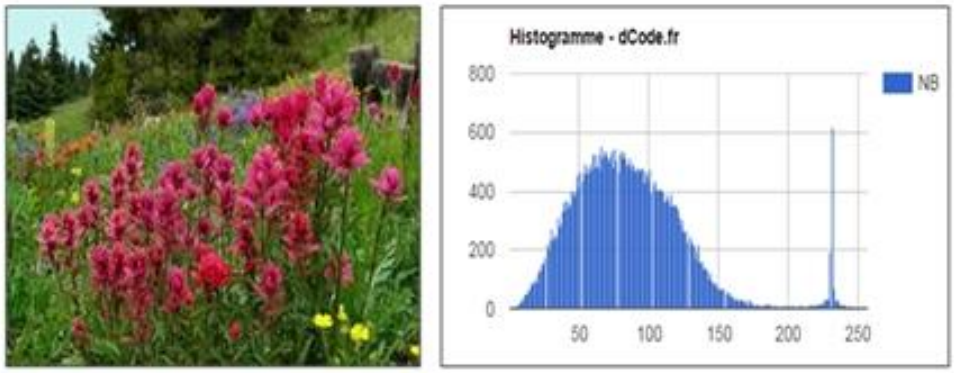

Figure 1.3Exemple d'histogramme.

4.1.2. Descripteur de Texture

La texture est une information de plus en plus utilisée en indexation et la recherche d'images par le contenu, car elle permet de pallier certains problèmes posés par l'indexation par la couleur [3], Mais d'une manière générale la texture se traduit par un arrangement spatial des pixels que l'intensité ou les couleurs seules ne suffisent pas à décrire [4].

Figure 1.4 Exemple de descripteur de texture [9].

4.1.3. Descripteur de forme

La forme est utilisée pour caractériser les objets dans les images. On distingue deux catégories de descripteurs de formes : les descripteurs basés régions et les descripteurs basés frontières. Les premiers sont utilisés pour caractériser l'intégralité de la forme

d'une région, Les seconds portent sur la caractérisation des contours de la forme [4].

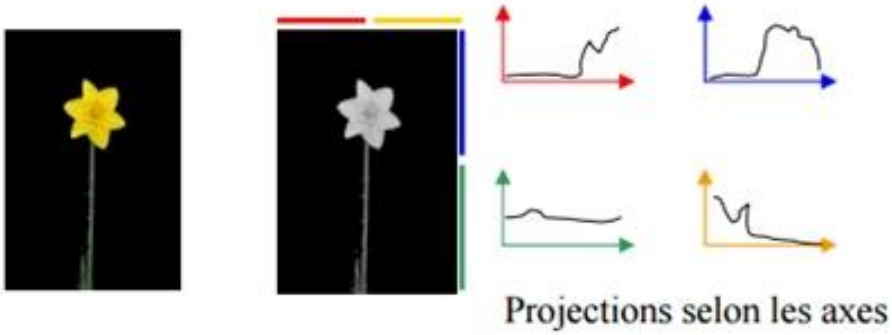

Figure 1.5 Exemple de descripteur de forme [9].

4.1.4. Descripteurs des points d'intérêts

Les points d'intérêt sont des points qui contiennent beaucoup d'information relativement à l'image. Ce sont des points aux voisinages desquels l'image varie significativement dans plusieurs directions [3].

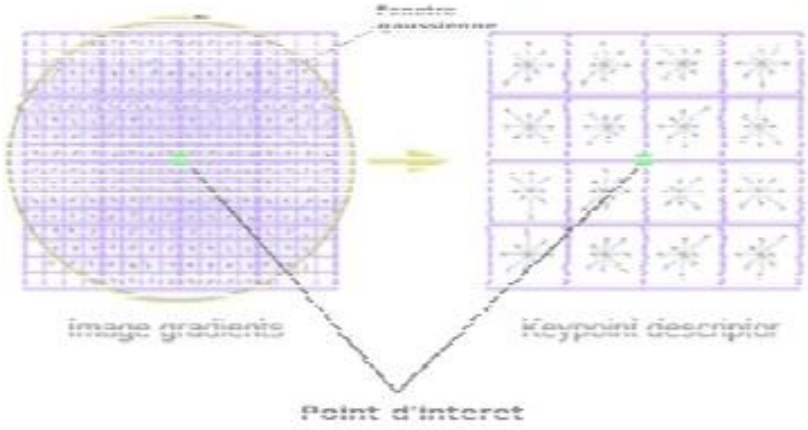

Figure 1.6 Exemple de descripteur des points d'intérêt [9].

5. L'indexation

5.1. Définition

L'indexation est une étape très importante dans le processus de RI, le processus d'indexation permet d'analyser chaque document de la collection (ou chaque requête) et d'extraire pour chacun d'entre eux un ensemble d'informations. Et pour faciliter la recherche, les informations extraites sont stockées dans un descripteur dénommé fichier d'index [1].

5.2. Les modes d'indexation

indexation manuelle : chaque document est analysé par un spécialiste du domaine correspondant ou par un documentaliste [1].

indexation automatique : chaque document est analysé à l'aide d'un processus entièrement automatisé [1].

indexation semi-automatique : le choix final reste au spécialiste du domaine correspondant ou documentaliste, qui intervient souvent pour établir des relations sémantiques entre mots-clés et choisir les termes significatifs [1].

5.3. La structure d'indexation

L'indexation est une étape très importante dans le processus de RI, le processus d'indexation est composé de plusieurs techniques, Cet techniques d'indexation on peut les classer en deux grandes approches :

- Le partitionnement de l'espace qui utilise des cellules d'espace pour indexer les données.
- Le partitionnement des données qui utilise des cellules d'objets similaires (fonction d'approximation) pour indexer les données.

5.3.1. Le partitionnement des données

Le partitionnement de données est une des méthodes d'analyse des données. Elle vise à diviser un ensemble de données en différents « paquets » homogènes, Nous allons voir comme principaux représentants de cette approche plusieurs techniques : l'arbreR, l'arbre-SR et l'arbre-M [10].

- L'arbre-R :

L'arbre-R a été proposé par Antonin Guttman, La structure d'un arbre-R est basée sur la décomposition de l'espace autour de rectangles englobant minimaux (REM). Dans un espace multidimensionnel, disons ◌ ◌ par souci de simplification, un rectangle englobant minimal est défini par un couple de vecteurs tel que les composantes du premier vecteur sont deux à deux inférieures ou égales à celles du second vecteur. Ce couple de coordonnées définit le plus petit volume qui englobe un ensemble de points et/ou de formes géométriques donné [10].

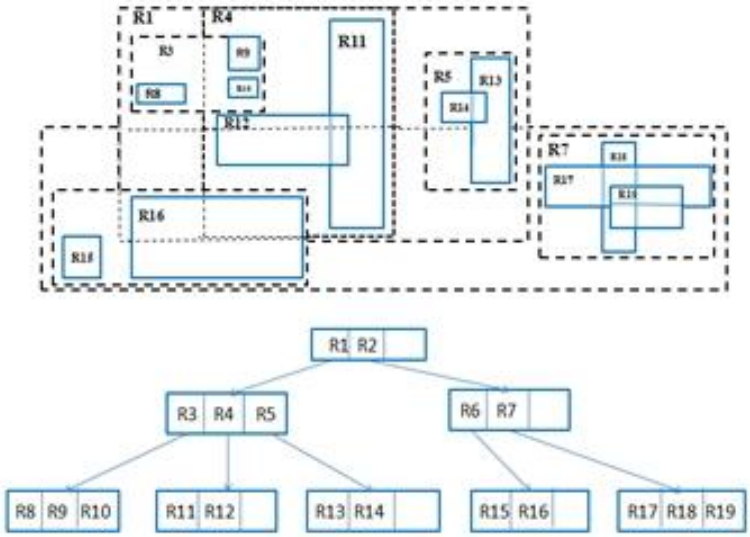

Figure 1.7 L'arbre R [10].

L'arbre-R indexe aussi bien des points que des rectangles, les premiers pouvant être vus comme des cas dégénérés de rectangles. Les données dans un arbre-R sont organisées en pages, qui peuvent avoir un nombre variable d'entrées :

Les nœuds feuilles stockent les données. Le nombre de rectangles n'est pas fixé, car il dépend de la taille des données elles-mêmes et de la taille des pages qui les stockent sur le disque. Les nœuds internes stockent un nombre variables d'« entrées ». Chaque entrée stocke deux éléments : un rectangle englobant minimal et un sous-arbre.

5.3.2. Partitionnement de l'espace

Suite au principal problème associe aux techniques d'indexation basée sur le partitionnement des données, c'est-à-dire les recouvrements entre formes englobantes, il existe une autre approche ou les intersections de régions sont nulles. Cette approche est basée sur le partitionnement de l'espace.

Plusieurs techniques s'appuient sur ce principe : arbre-kD, arbre-LSD, fichier- grille, etc [10].

Arbre-GH :

L'arbre-GH (Generalised Hyper-plane) utilise le principe du partitionnement dans les espaces métriques à l'aide d'hyper-plans. Rappelons que ce plan est défini par deux points p_1 et p_2. La figure illustre le concept.

Soit (O, d) un espace métrique. Soit E un sous-ensemble d'éléments à indexer.

Alors, on définit les nœuds N_{GH} d'un arbre-GH de la manière habituelle :

- Tout d'abord, un nœud NGH consiste en deux éléments et deux fils :

$$(p_1, p_2, G, D) \in E \times E \times \mathcal{N}_{GH} \times \mathcal{N}_{GH}.$$

Ou:

- P_1 , P_2 sont deux éléments non confondus, d (p_1 , p2) > 0, dénommes « pivots », ils définissent ainsi un hyper-plan;
- G et D sont les sous-arbres associes aux éléments se plagant respectivement dans les parties « gauche », G(O,d,p_1 ,p_2), et « droite », D(O, d,p_1,p_2) de l'hyper-plan H(O,d,p_1 ,p_2).
- Un (sous) arbre peut être vide, ce que l'on dénote par T.

Un arbre-GH est donc un type récursif [1 0] .

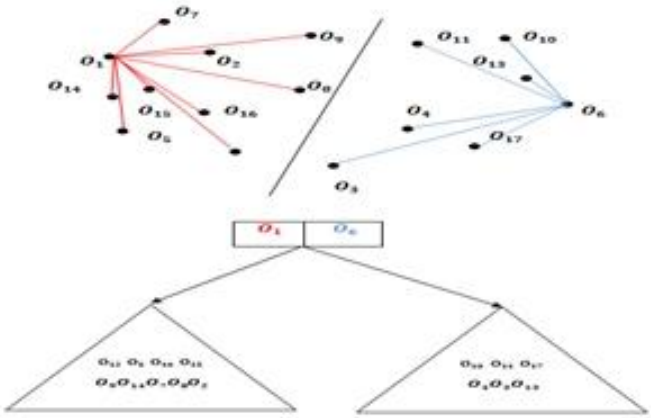

Figure1.8 L'arbre GH [10].

6. Conclusion

Dans ce chapitre nous avons présentés quelque concept du domaine de la recherche d'information, et nous avons présenté le principe et les fonctionnalités d'un SRI, ensuite nous avons parlé sur quelque technique d'indexation.

Chapitre N° 02

Les algorithmes parallèles
et
Les algorithmes distribués

1. Introduction

Le chapitre deux présente quelques notions sur les algorithmes parallèle, à savoir les architectures parallèles, ensuite nous allons à présent introduire les principaux critères d'évaluation pour mesurer les performances des programmes parallèles (c.à.d. concept de base de l'algorithme parallèle) et les modèles de programmation parallèle. Aussi, nous y présenterons les algorithmes distribués et leurs caractéristiques.

2. Algorithmique parallèles

Les algorithmes séquentiels sont exécutés sur une seule machine et un seul processeur.

Le parallélisme est utilisé depuis longtemps en informatique pour résoudre des problèmes scientifiques de grande taille (simulation, météorologie, biologie, jeux vidéo) le plus rapidement possible [11]. Le parallélisme permet d'utiliser plusieurs processeurs qui fonctionnent simultanément pour augmente les performances (puissance de calcule et capacité de stockage et le temps d'exécution) d'un problème donné.

2.1. Les architectures parallèles

Une machine parallèle est un ensemble de processeurs capables de travailler en ensemble à travers un réseau d'interconnexion pour résoudre un problème de grosse taille.

Les architectures parallèles caractérisées principalement par différents modèles d'interconnexions entre les processeurs et la mémoire. Flynn propose quatre types principaux de machines parallèles basée sur la notion de flot de données et de flot d'instructions [12]:

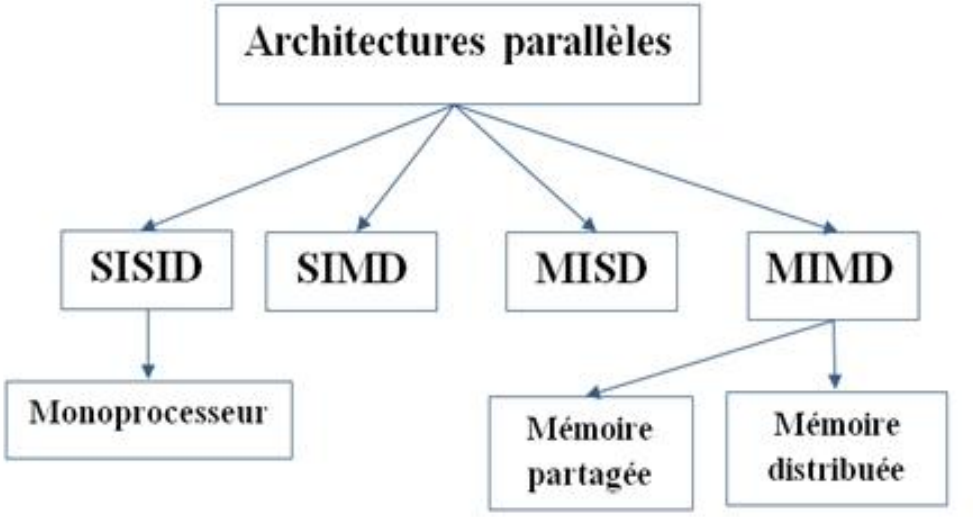

Figure 2.1 classification de Flynn.

Les machines SISD (Single Instruction Single Data) : Dans ces machines une seule instruction est exécutée et une seule donnée est traitée à tout instant (séquentiellement). Ce modèle correspond à une machine monoprocesseur.

Les machines SIMD (Single Instruction Multiple Data) : Dans ces machines, tous les processeurs sont identiques et sont contrôlés par une unique unité de contrôle centralisée. A chaque étape, tous les processeurs exécutent la même instruction de façon synchrone mais sur des données différentes.

Les machines MISD (Multiple Instructions Single Data) : Ces machines peuvent exécuter plusieurs instructions sur la même donnée. Les machines MIMD (Multiple Instructions Multiple Data) : Dans ce modèle, chaque processeur est autonome, dispose de sa propre unité de contrôle et exécute son propre flot d'instructions sur son propre flot de données. Ces machines sont les plus courantes aujourd'hui. Parmi ces types de machines il y a ceux à :

- mémoire commune: plusieurs processeurs se partagent une même mémoire.
- mémoire distribuée : chaque processeur possède sa propre mémoire et peut communiquer avec les autres processeurs.

- hybrides : mémoire commune / mémoire partagée.

2.2. Concept de base de l'algorithmique parallèle

Généralement l'algorithme séquentiel est évalué en termes de temps d'exécution exprimé en fonction de la taille de ses entrées. Par contre le temps d'exécution d'un algorithme parallèle dépend de la taille de ses entrées et l'architecture parallèle sur laquelle il est exécuté (nombre de processeurs) [12].

2.2.1. Le temps d'exécution

Le temps d'exécution d'un programme séquentiel est la déférence entre la fin et le début de son exécution. Le temps d'exécution d'un programme parallèle (p nombre de processeurs) est le temps écoulé entre le moment où le calcul parallèle débute et le moment où le dernier processeur termine son exécution [12].

2.2.2. Accélération

Le gain de performance obtenu en parallélisant d'une application donnée par rapport à son implantation séquentielle. L'accélération permet de mesurer ce gain.
L'accélération est notée et est donc formulée de la façon suivante [12]:

$$\text{accélération} = \frac{\text{temps séquentiel}}{\text{temps parallèle}}$$

2.2.3. L'efficacité

L'efficacité E est définie comme le rapport entre l'accélération et le nombre de processeurs, la formule mathématique est donnée comme suit [12]:

$$E = \frac{\text{accélération}}{P}$$

2.2.4. Le Coût

Le coût d'un système parallèle c'est le produit du temps d'exécution parallèle par le nombre de processeurs utilisés. Par contre le cout de problème d'un seul processeur correspond au temps d'exécution du meilleur algorithme séquentiel connu [12].

3. Algorithmique distribué

3.1. Système distribué

Un système distribué est un système qui s'exécute sur un ensemble des machines interconnecté via un réseau de communication son mémoire partagé comme une seule machine. Donc un système distribué est similaire à une machine parallèle à la différence qu'il y a plusieurs machines de calculs autonomes distantes.

3.1.1. La différence entre un système distribué et un système parallèle

Une machine parallèle est composée de plusieurs processeurs qui coopèrent à la solution d'un même problème. Mais le système distribué est de plusieurs processeurs distants impliqués dans la résolution d'un ou plusieurs problèmes.

Les deux définitions sont proches et la frontière entre machine parallèle et système distribué est un peu floue le tableau suivant présente une petite comparaison entre une machine parallèle et un système distribué [13] :

Parallélisme	Systèmes distribués
Objectif de traiter des problèmes plus gros plus vite.	Résulte souvent de la nécessité de répartir spécialement des services ou des composants software sur des ordinateurs distants.
Mise en commun organisées de ressources de calcul.	Résulte aussi du fait que certaines applications définissent naturellement des entités en concurrence.
On va souvent essayer d'exploiter des régularités de l'architecture.	Reflète le modèle client-serveur, producteur-consommateur, la mise en réseau de plus en plus de ressources.

On a un couplage fort entre processeurs, une granularité fine du découpage du problème (au niveau des variables du programme : les processeurs travaillent sur des variables différents du programme).	Il n'y a pas d'enjeu de performance : couplage faible, granularité grosse (découpage au niveau de l'application : distribution de morceaux d'application à chaque processeur).

Tableau 2.1 Comparaison entre les systèmes parallèles et les systèmes distribués [13].

3.2. Les caractéristiques de système distribué

Un système distribué possède des caractéristiques fondamentales [14] :

Absence d'état global : pas de référentiel temporel commun (plusieurs horloges) et pas de référentiel spatial commun (plusieurs mémoires).

Existence d'un réseau (hors système d'exploitation).

L'architecture matérielle.

L'architecture système (exécution parallèle).

3.3. Les propriétés d'un système distribué

Un système distribué a plusieurs propriétés [14]:

La disponibilité : le système est toujours accessible. La

fiabilité : le système continu le service.

La tolérance : le système fonctionne correctement même en présence de fautes. La

sécurité : le système combiner plusieurs techniques de sécurité.

La performance : le système est parallèle et concurrent.

La transparence : le système ne peut être qualifié de réparti.

3.4. Communication dans les systèmes distribués

3.4.1. Réseau de communication

Un système distribué est un ensemble des machines interconnecté via un réseau de communication (échange des massages) son mémoire partagé. Le réseau de

communication a une topologie décrit comment les machines peuvent communiquer entre eux. Le réseau est présenté par deux processus sur des machines distantes l'un est l'émetteurs et autre est récepteur (synchronisé) [14].

3.4.2. Le modèle client-serveur

Le modèle client-serveur est un système distribué (plusieurs machines) composé de deux types des machines les machines clients qui se connectent la machine serveur généralement très puissante.

Le client est la machine qui envoie des demandes des services à un serveur par une requête et reçois la réponse.

Le serveur est la machine qui offert un service a les clients. Il accepte et traite les requêtes des clients ensuite envoie le résultat au client qui demande le service.

La requête est un message transmis par un client à un serveur.

La réponse est un message transmis par un serveur à un client après l'exécution de la requête du client.

Le système client-serveur fonctionne comme le schéma suivant :

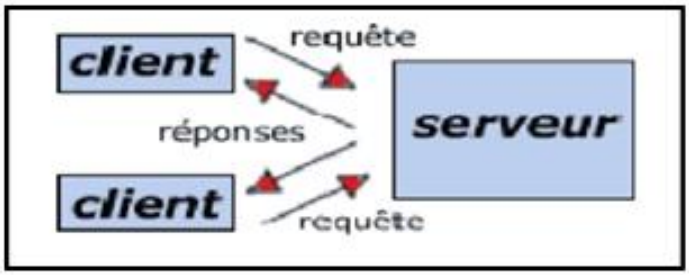

Figure 2.2 Fonctionnemt de modele client-serveur.

4. Conclusion

Dans ce chapitre, nous avons présenté l'algorithmique parallèle et les algorithmes distribués. Nous avons vu que plusieurs types d'architectures parallèles existent pour l'exécution des programmes parallèles. Mais chaque type d'architecture a ses particularités (mémoire partagée ou distribuée).En plus, nous avons vu la présentation de quelques critères de performance permettant de juger de la qualité de l'algorithme parallèle. Enfin, ce chapitre s'est terminé par la présentation d'une petite comparaison entre le parallélisme et les systèmes distribués, aussi la communication

dans les systèmes distribués.

Chapitre N° 03

La conception de système

1. Introduction

Dans ce chapitre nous avons présenté la conception de notre système d'indexation et de recherche d'image par le contenu (CBIR 2.0) qui on a déjà la version séquentielle (CBIR 1.0) qu'il été réalisé par deux étudiants de niveau master deux en 2016 [15]. Cette version séquentielle abordé le problème d'indexation dans une grande collection des données avec une grande dimensionnalité, Mais si la dimensionnalité des images augmentant, la complexité de construction et de recherche augmente rapidement.

Pour résoudre ce problème nous avons proposé une nouvelle version parallèle (CBIR v 2.0).

2. Conception séquentielle (version 2016 CBIR 1.0)

En 2016, les deux étudiants de deuxième année master proposé une conception d'un index arbre binaire avec des conteneurs au niveau des feuilles (clusters), qui vont permettre d'indexer des objets images. Ils ont proposé deux algorithmes, le premier un algorithme de construction et le deuxième un algorithme de recherche KNN plus proche voisins. Ils ont aussi proposé des algorithmes qui s'adaptent avec tous type d'information sur les images (textuelles et visuelles).

Voilà le schéma suivant qui représenter leur version séquentielle (CBIR 1.0) :

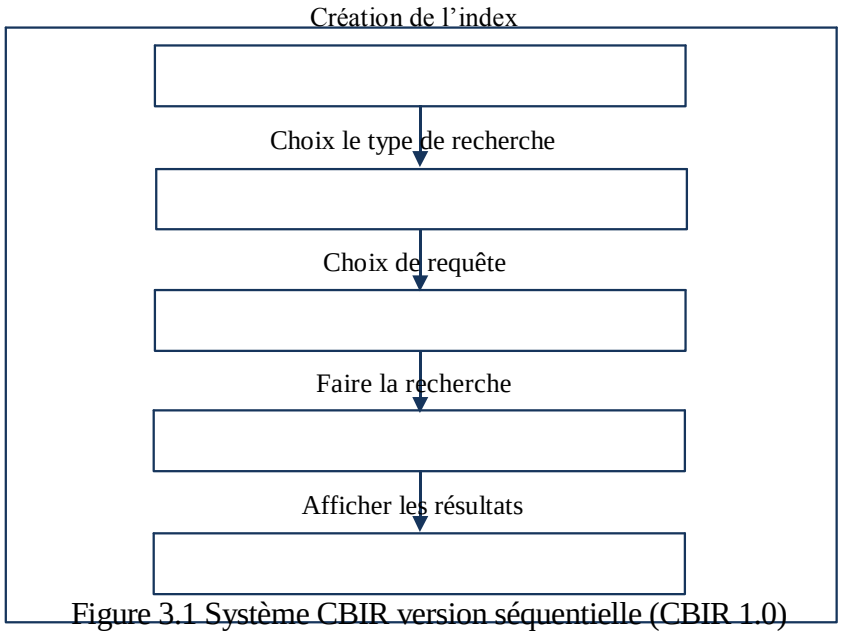

Figure 3.1 Système CBIR version séquentielle (CBIR 1.0)

3. Notre proposition

Vu la dimensionnalité des images actuelle (plus de 100 dimension) et la quantité énorme de l'information qui existes ; nous proposons notre recherche adopter par ce système.

3.1. Architecture générale du système

L'objectif de notre projet est la réalisation parallèle d'une application de recherche d'image par le contenu en parallèle, pour cela nous avons choisi le modèle client-serveur.

Le modèle client-serveur comme nous avons défini dans le chapitre 2 est un système distribué composée de deux types des machines une machine de type serveur les autres sont de type client.

- Notre système est composé de :
    1. Machine serveur : est une machine qui crée l'index à partir de la base qui stocké dans la machine elle-même.
    2. Machines clients : un groupe des machines connectés au machine serveur pour aide dans les calcules.
    3. Index : la machine serveur crée l'index avec les images stocké dans la machine : regrouper les images dans des collections des éléments similaires (des clusters).
    4. Cluster : le serveur distribué l'ensemble des clusters à les machines clients.
    5. Requête : La requête est une représentation possible du besoin, elle détermine le type de recherche et elle représenté par trois type sois une image ou image XML ou par mot clé.
    6. Le tri : une méthode de tri pour ordonner les résultats.

Les schémas suivants expliquent l'architecture de notre système:

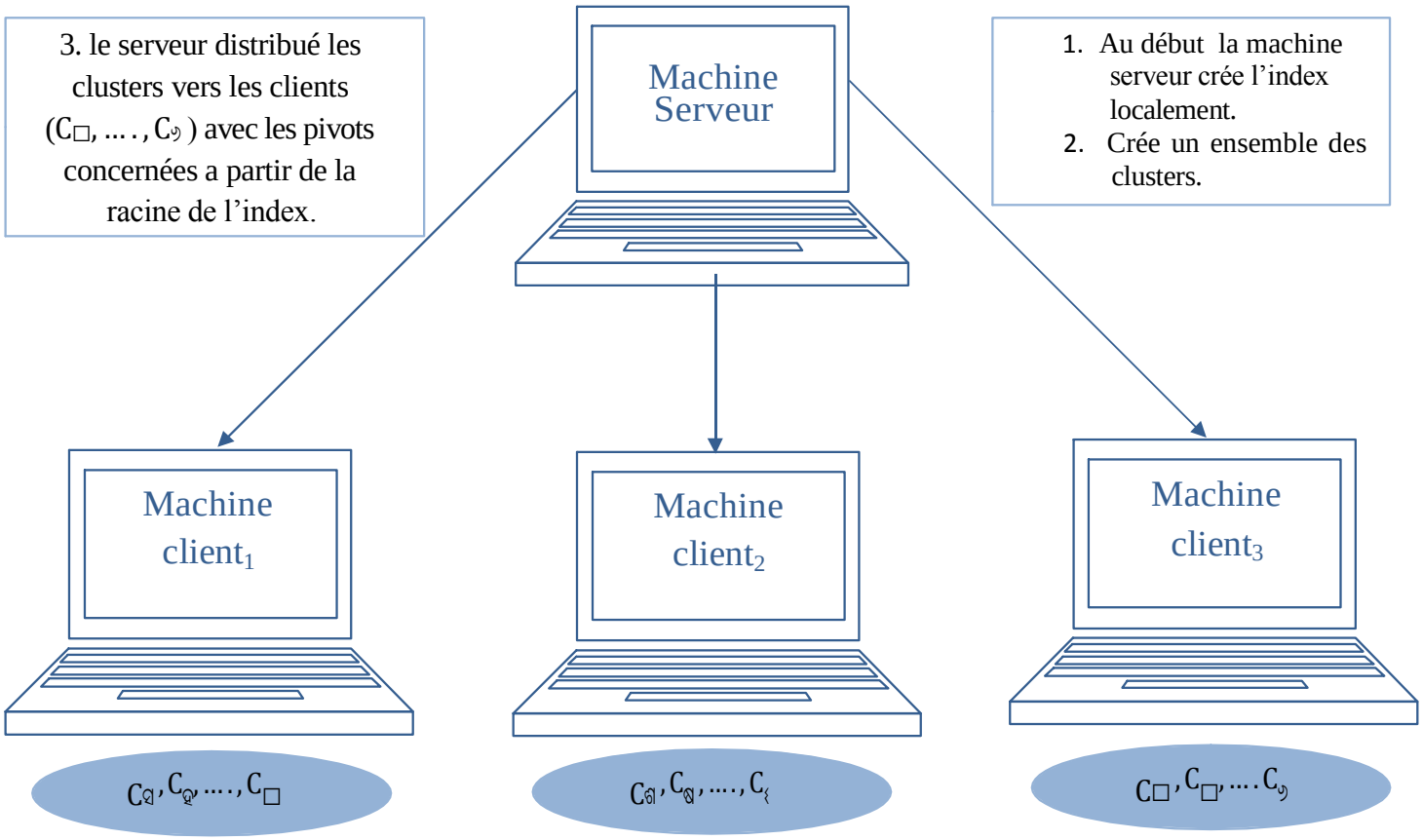

Figure 3.2 Création de l'index sur plusieurs machines.

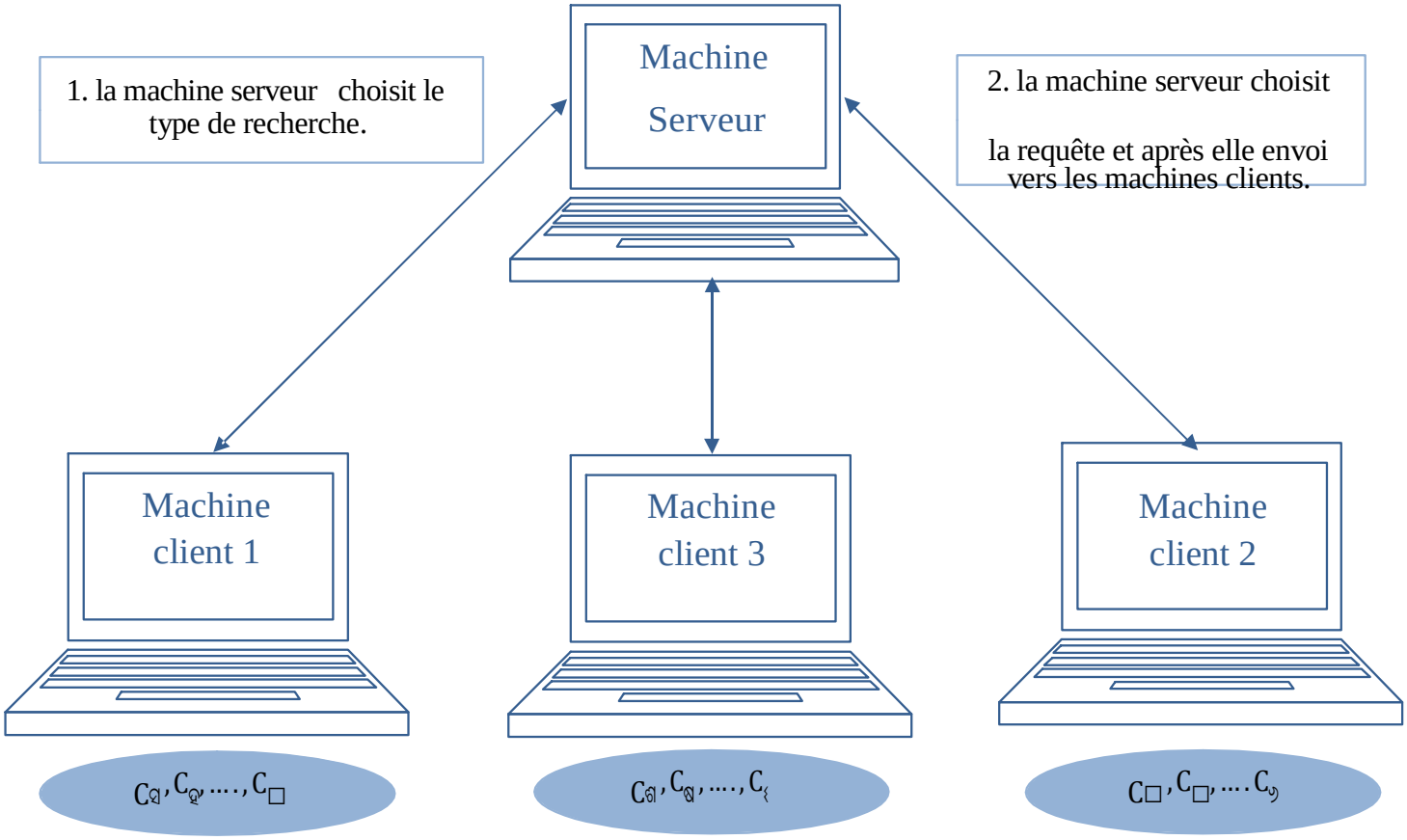

Figure 3.3 Notre algorithme de recherche sur plusieurs machines.

3.1.1. Création

3.1.1.1. Création d'index

Comme l'indexation c'est l'étape la plus importante pour la recherche d'information (un accès facile à l'information); nous avons parlés sur les étapes de préparation des données avant de les insérer dans la structure d'indexation.

Le schéma suivant explique les étapes:

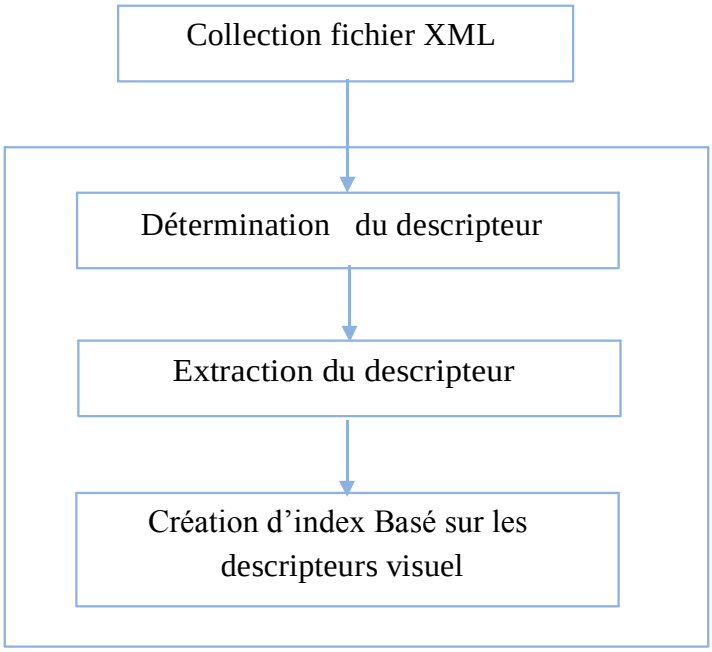

Figure 3.4 Création d'index.

- Collection fichier XML : la préparation de la collection des images (fichier XML).

- Détermination du descripteur : choix les types de descripteur.

Extraction du descripteur : Le système extrait les informations visuelles via les images (fichier XML).

- Création d'index Basé sur les descripteurs visuel: le système crée l'index à partir les descripteurs visuel.

3.1.1.2. La détermination des descripteurs d'images

Nous avons utilisé la base CoPhIR des images Fliker. La récupération d'image de photo basée sur le contenu (CoPhIR) est la plus grande base de données disponible des images numériques avec des descripteurs visuels correspondants (plus de 106 millions d'images traitées).CoPhIR est maintenant disponible pour essayer de comparer et obtenir [16]:

l'efficacité de l'indexation et de la recherche basée sur la similarité.

l'expressivité de la combinaison des descripteurs avec respect de la perception subjective de la similarité visuelle.

Pour Chaque photo dans la collection CoPhIR Contient une description XML de son caractéristiques :

Une structure XML avec les informations utilisateur Flickr dans l'entrée Flickr correspondante: titre, emplacement, GPS, tags, commentaires, etc.

Le lien vers l'entrée correspondante dans le site Web Flickr et la miniature de l'image photo:

Figure 3.5 Structure d'un fichier XML partie photo.

Une structure XML avec 5 fonctions d'image MPEG-7 standard extraites les caractéristiques de l'image :

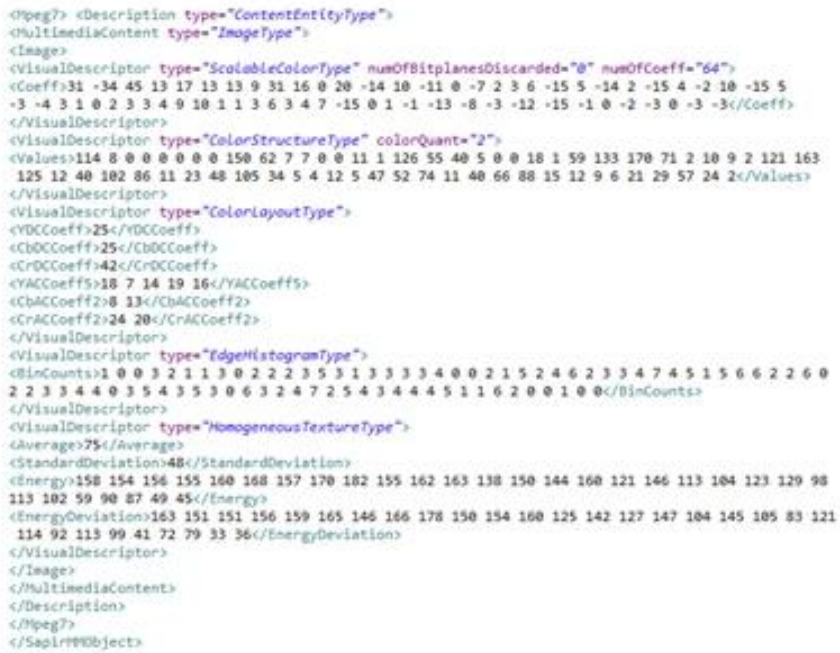

Figure 3.6 Structure d'un fichier XML partie MPEG-7.

Chaque descripteur d'image définit par une dimension qui représenté par le tableau suivant :

Descripteur	Dimension
ScalableColor	64
ColorStructure	64
ColorLayout	12
EdgeHistogram	80
HomogeneousTexture	62
Mot clé	_

Tableau 3.1 Dimensionnalités des descripteurs MPEG7 utilisés.

3.1.1.3. La structure de l'indexation

Nous avons utilisé la structure d'indexation arbre binaire GH. Le principe de l'arbre GH est de partitionne les espaces métriques à l'aide d'hyper-plans en deux parties. Rappelons que ce plan est défini par deux points $p_1$et p_2 (deux pivots). A chaque itération nous calculons deux distances d_1, d_2 par apport à deux pivots $p_1$et p_2.
On utilisant la distance euclidienne pour regrouper la base des images et extraire les images les plus similaires à l'image requête. Cette distance est définit par la formule suivante :

$$\square)\square_{\square}, \square_{\square}) = \square\square(\square_{\square}(\square - (\acute{\square}_{\square} (\acute{\square}))^{\square}$$

Comme déjà dit précédemment, Nous avons utilisé l'arbre binaire GH pour faire le regroupement des données, L'arbre GH partitionne les espaces métriques à l'aide d'hyper-plansen deux parties, avec chaque plan est défini par deux point $p_1$et p_2.
L'arbre GH est composé de deux types des nœuds : les nœuds N_{GH} et les nœuds feuilles.
Tout d'abord, on définit les nœuds N_{CH}, un nœud N_{CH} consiste en deux éléments et deux fils :

$$(p_1, p_2, G, D) \in E \times E \times \mathcal{N}_{GH} \times \mathcal{N}_{GH}.$$

Les deux pivots P_1, P_2 sont deux éléments non confondus (d (p_1 , p_2) > 0), définissent ainsi un hyper-plan.

Les deux fils (G et D) sont les sous arbres associes aux éléments se plagant respectivement dans les parties « gauche », G (O, d, p_1, p_2), et « droite », D (O,d,p_1 , p_2) de hyper-plan H (O,d,p_1 , p_2).

Un sous arbre peut être vide.

Donc l'arbre-GH est un type récursif.

La taille de chaque cluster (nœud feuille) varie entre un objet et C_{max} objets.

Après, nous représentons comment regrouper les données à partir de l'arbre GH dans le schéma suivant :

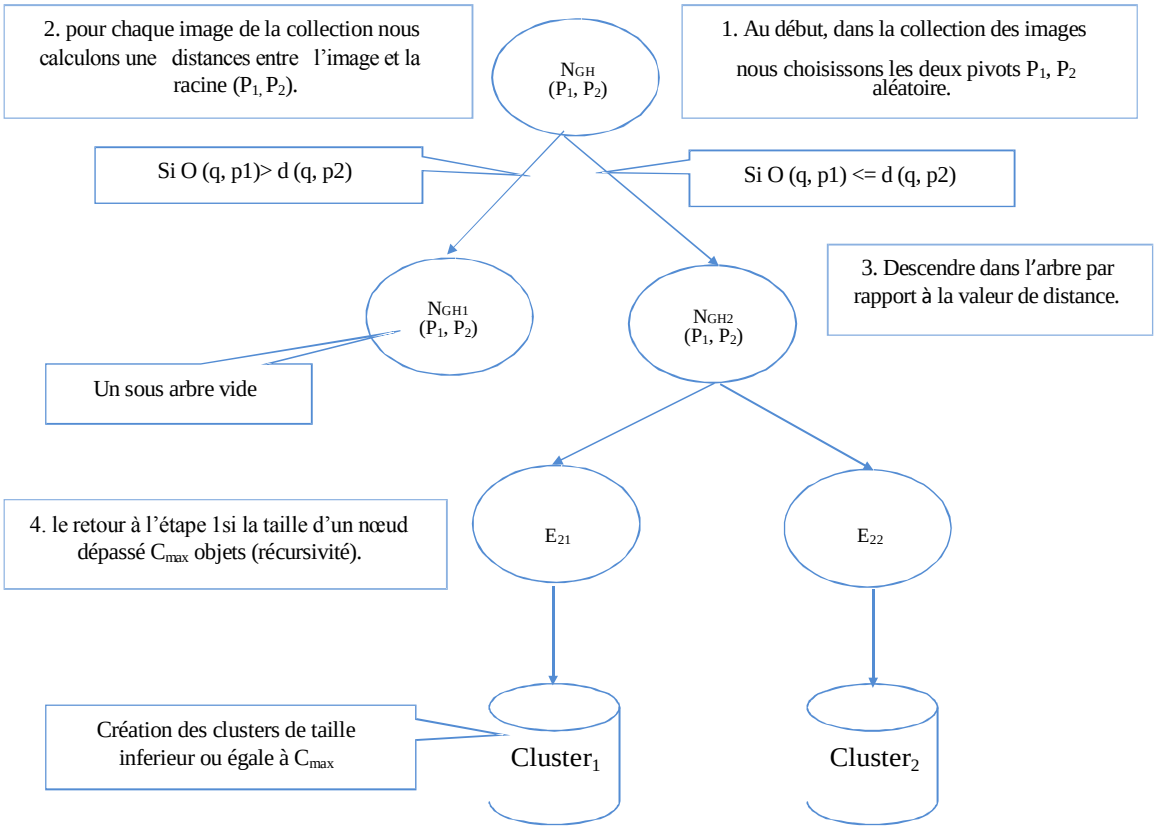

Figure 3.7 Structure d'indexation.

3.1.2. Recherche

Nous avons présenté la méthode d'indexation dans la partie précédente, maintenant

dans cette partie nous arrivons à la partie recherche.

3.1.2.1. Type de recherche

Comme nous avons dit précédemment la requête de la recherche est existée en trois façons :

Par mot clé.

Par exemple XML.

Par image.

La recherche est déterminée sur la base de choix de type de la requête qui passer par les étapes suivantes :

Choix de la requête.

Extraire les descripteurs de la requête.

Calcule de distance entre les descripteurs de la requête et les descripteurs des images de l'index.

Affiche les résultats de la recherche dans une liste d'image ordonnée. Le schéma suivant explique les trois types de recherches :

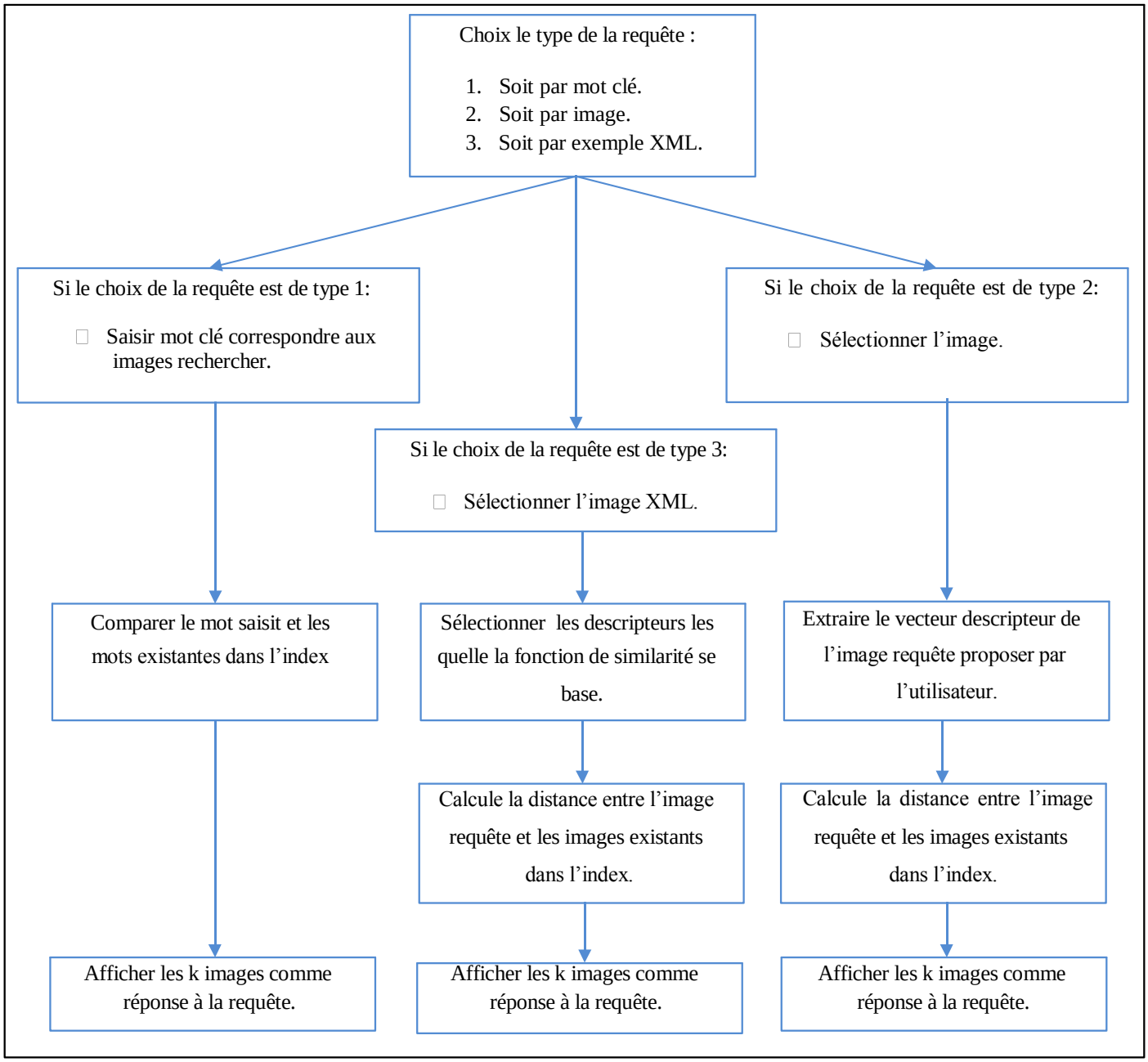

Figure 3.8 Les types de recherche.

3.1.2.2. Méthode de recherche

Après la construction de notre index, maintenant nous arrivons a la partie de recherche qu'est faite au niveau des machines clients.

Nous rappelons que notre objectif principalement fixé sur le temps de construction de l'index et aussi le temps de recherche.

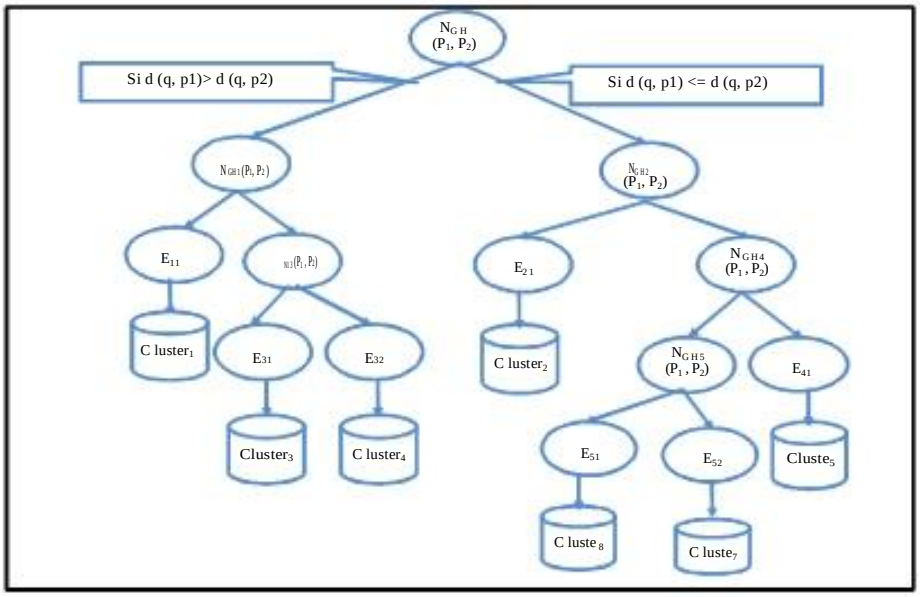

Figure 3.9 Exemple d'index.

La machine serveur distribue les clusters de l'index vers les machines clients, chaque machine client reçoit un ensemble de clusters avec les pivots concernés à partir de la racine d'index jusqu'aux clusters. Ensuite, quand la machine client reçue la requête, elle commence la comparaison de cette requête avec les pivots concernés jusqu'à le cluster le plus similaire.

Nous proposons qu'il y a deux machines clients, l'index précédent est distribué en deux sous indexes :

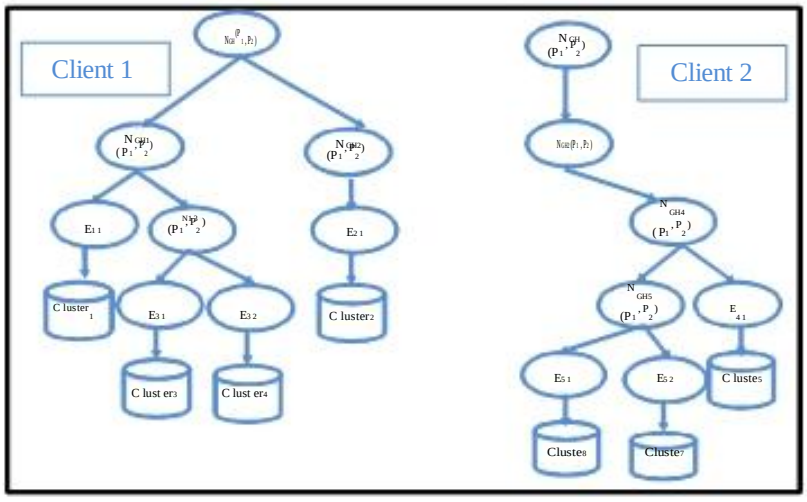

Figure 3.10 Exemple de recherche sur deus machine client.

La figure 3.10 représente la recherche sur deux machines, chaque recherche passe par les étapes suivantes :

1. Au début, on commence par le calcul de distance entre la requête et les deux pivots p_1, p_2.
2. On descendra par rapport à la valeur de distance :

   Si d (q, p1)> d (q, p2) vers le nœud droit.

   Si d (q, p1) <= d (q, p2) vers le nœud gauche.
3. Calcul la distance entre la requête et chaque image qui résident dans le cluster le plus similaire à la requête.
4. Après le calcule des distances on lance une méthode de tri pour trié les Objets « trie par bulle».
5. Chaque machine client transférer les résultats vers la machine serveur.

La machine server reçue les résultats de chaque machine client, après elle lance la fonction de tri pour ordonner les résultats et afficher les K plus proche.

3.1.2.3. Méthode de tri

Pour cette partie nous avons utilisé l'algorithme de tri le tri a bulle, l'algorithme parcourt le tableau et compare les éléments adjacents. Lorsque les éléments ne sont pas dans l'ordre, ils sont échangés.

Après le premier parcours, le plus grand élément étant à sa position définitive, il n'a plus à être traité. Le reste du tableau est en revanche encore en désordre. Il faut donc le parcourir à nouveau, en s'arrêtant à l'avant-dernier élément. Après ce deuxième parcours, les deux plus grands éléments sont à leur position définitive. Il faut donc répéter les parcours du tableau, jusqu'à ce que les deux plus petits éléments soient placés à leur position définitive [17].

```
Tri  à bulle(t : tableau)
 Début
        pour i allant de taille de t à1
        faire
              pour  j  allant  de  1  à  taille  de  t
              faire
                    Si t[j+1] < t[j] alors
                                      Echanger (t[j+1], t[j] ) ;
```

fin;

fin;

fin;

Fin.

4. Conclusion

Dans ce chapitre nous avons proposé une conception d'un index arbre binaire GH basé sur le partitionnement d'espace, après la création de l'index nous avons proposé une méthode de recherche parallèle pour augmenter la vitesse de recherche et réduire le temps de réponse.

Chapitre N° 04

Réalisation et Implémentation

1. Introduction

Nous arrivons à la phase la plus importante, c'est celle de l'implémentation et tests. Le choix des outils de développement influence énormément sur le coût en temps de programmation, ainsi que sur la flexibilité du système à réaliser.
Nous allons commencer par la description de l'environnement de travail puis à dégager et élaborer les composants de notre système.

2. Environnement de travail

L'environnement de travail est constitué par deux parties nommées environnement matériel et environnement logiciel.

2.1. Environnement matériel

Le développement de l'environnement matériel est caractérisé par :

Système d'exploitation : Windows 7 Intégrale
CPU : Pentium (R), 2.30 GHz
Mémoire : 3 Go

2.2. Environnement logiciel

L'environnement logiciel consiste les composants suivants :

- Java.
- Outil de développement Eclipse.

3. Implémentation

3.1. Choix de langage de programmation : Java

Pour implémenter notre système, le langage de programmation java est le mieux adapté. En effet, Java s'annonce comme une des évolutions majeures de la programmation. Pour la première fois, un langage efficace, performant, standard et facile à apprendre (et, de plus, gratuit) est disponible. Il est un langage de programmation orienté objet multi-plate-forme qui permettrait, selon le principal proposé par Sun Microsystems en 1995, son concepteur, d'écrire des applications capables de fonctionner dans tous les environnements. Java donne aussi la possibilité

de développer des programmes pour téléphones portables et assistants personnels. L'objectif était de taille, puisqu'il a réussit à gagner une grande popularité auprès des programmeurs grâce à ses avantages. Ce langage offre une portabilité maximale grâce à une indépendance totale par rapport au système [18].

3.2. L'éditeur

Eclipse est un IDE, IntegratedDevelopmentEnvironment (EDI environnement de développement intégré en français), c'est-à-dire un logiciel qui simplifie la programmation en proposant un certain nombre de raccourcis et d'aide à la programmation. Il est développé par IBM, est gratuit et disponible pour la plupart des systèmes d'exploitation.
Au fur et à mesure que vous programmez, eclipse compile automatiquement le code que vous écrivez [19].

Figure 4.1 L'éditeur éclipse.

4. Les données utilisées

Dans notre système nous avons utilisé la base CoPhIR (enseemble des fichiers XML qui contient les descripteurs MPEG7) des images Flickr.

Nous avons exploité dans notre système encore cent milles images pour le tester sa rapidité et son efficacité.

En plus, nous avons testé sur les différentes quantité des images (100,1000,10000) dans les deux parties qui concerne la création d'index et la recherche.

5. Développement de l'application

Dans cette partie, nous allons présenter les différentes phases de la réalisation de notre projet en mentionnant des imprimes écrans de notre application.

Voila l'exécution de notre système CBIR v2.0 du début de la création de l'index jusqu'à ce que l'accès aux résultats de la recherche.

Figure 4.2 Fonctionnalité de notre système CBIR v2.0.

Cette interface est composée par les différentes fonctionnalités de notre système comme suivant :

- Indexation : cette fonction est consistée à la création de notre index, elle est composée de deux parties :
  - Création de l'index : permet le partitionnement des données dans des clusters.
  - Distribuer l'index : permet la transformation des clusters vers les autres machines qui sont connectés avec lui.
- Recherche : permet la détermination des besoins de la recherche.
- Aide : contient les paramètres qui déterminent à quoi la création d'index basé.
- Les clients connectés : espace pour afficher les machines clients disponibles.
- Les informations de l'index : espace pour afficher les informations de création d'index par exemple (nombres des clusters générés, les descripteurs

….. etc.).

- Autres informations : espace pour afficher le nombre de clusters de chaque client, le temps de création d'index et le temps de réponse d'une recherche.

Figure 4.3 Paramètre d'index.

Figure 4.4 Détermination des besoins de la recherche.

Figure 4.5 Choix de la requête (fichier XML).

Figure 4.6 Les résultats de la recherche.

6. Les tests

Dans cette partie nous avons faire des simulations sur notre système d'indexation et recherche, et nous avons focalisé sur le temps de création d'index et de réponse sur plusieurs machines.

6.1. Base d'images varie

Au début, nous allons prendre une base d'images de taille varie entre [400 jusqu'à 10000 images] avec une taille du cluster fixé [100] et une combinaison des descripteurs.

Le tableau suivant représente les résultats de ce test avec nombres des machines différents:

Machines	La taille de base (images)	Temps de création d'index (ms)	Temps de réponse (ms)
Une seule machine	400	533	360
	1000	2231	184
	10000	284578	158
Deux machines	400	854	225
	1000	2532	167
	10000	482451	142
Trois machines	400	1070	231
	1000	4254	190
	10000	261862	180

Tableau 4.1 Index avec une taille de base varie.

6.1.1. Explication et interprétation des résultats

A partir des résultats qui obtenus dans le tableau nous remarquons que:

- Par apport la taille de la base: tant que la base des images augmente, le temps de création et de réponse augmente aussi.
- Par apport le nombre des machines : tant que le nombre des machines augmente, le temps de création augmente par ce que, cette augmentation est provoqué par le réseau d'interconnexion. (distribution) et le temps de réponse diminuer.

6.2. Taille de cluster varie

A la suite, nous avons testé sur la taille de cluster qui a été varié [30, 60, 100] avec le fixage de la taille de la base [400] et les descripteurs Ce qui est représenté dans le tableau suivant :

Machines	La taille de cluster	Temps de création d'index (ms)	Temps de réponse (ms)
Une seule machine	30	1157	219
	60	823	78
	100	658	164
Deux machines	30	979	181
	60	967	158
	100	776	162
Trois machines	30	1012	736
	60	894	409
	100	852	255

Tableau 4.2 Index avec une taille de cluster varié.

6.2.1. Explication et interprétation des résultats

Nous remarquons que le temps de création diminuer si la taille de cluster augmente, mais il est varie lorsque le nombre des machines modifies, cette variance est provoqué par le réseau d'interconnexion comme nous ditons précédemment.

6.3. Descripteur varie

A la fin, nous travaillons pour ce test sur les descripteurs d'images, voir le tableau suivant :

Machines	Descripteurs des images	Temps de création (ms)
Une seule machine	Couleur dominante	383
	Couleur dominante + Structure de couleur	626
	Couleur dominante + Structure de couleur + Histogramme de couleur	527
	Couleur dominante + Structure de couleur + Histogramme de couleur + Texture homogène	631
Deux machines	Couleur dominante	610
	Couleur dominante + Structure de couleur	550
	Couleur dominante + Structure de couleur + Histogramme de couleur	644

	Couleur dominante + Structure de couleur + Histogramme de couleur + Texture homogène	625
Trois machines	Couleur dominante	611
	Couleur dominante + Structure de couleur	681
	Couleur dominante + Structure de couleur + Histogramme de couleur	825
	Couleur dominante + Structure de couleur + Histogramme de couleur + Texture homogène	1031

Tableau 4.3 Index avec les différents descripteurs.

6.3.1. Explication et interprétation des résultats

Nous remarquons que le temps de création augmente si nous sélectionnons a chaque fois un autre descripteur.

7. Qualité d'index

Descripteurs	Nombre d'images pertinentes dans 20 images	Précision
Couleur dominante	5/20	25 %
Couleur dominante + Structure de couleur	6/20	30 %
Couleur dominante + Structure de couleur + Histogramme de couleur	9/20	45%
Combinaison des descripteurs	10/20	50 %

Tableau 4.4 Qualité d'index.

7.1. Explication et interprétation des résultats

Nous remarquons qu'augmentant la taille de vecteur de l'image la précision des résultats augmente du 25% avec la couleur dominante seul jusqu'à 50% avec un vecteur de grande dimension (combinaison des descripteurs).

8. Conclusion

Dans ce chapitre nous avons effectué des expérimentations sur notre système CBIR et Nous avons détaillé les étapes du travail élaboré afin d'obtenir les résultats souhaités.

Conclusion et perspectives

Dans ce mémoire, nous avons abordé le problème de la recherche d'images. Plus précisément, nous nous sommes focalisés à la recherche d'images basée sur le contenu. Notre choix a été motivé par la quantité phénoménale d'images disponible aujourd'hui, qui ne cesse de croitre.

Ces dernières années avec l'explosion de la quantité d'informations, on parle de millions voir des trillions images sur le web.

Les images sont des données souvent utilisé dans les moteurs de recherche, leurs complexité est très grandes ce qui pose un problème de complexité des algorithmes de construction de l'index et les algorithmes de recherche associées.

Les algorithmes parallèles et distribuer sont venu pour facilité les algorithmes de recherche dans le domaine informatique.

Le but de ce travail n'a pas été de proposer une nouvelle technique, mais d'améliorer une technique d'indexation existé déjà, il s'agit des arbre-Gh et de trouver une solution basé sur une méthode efficace et performante pour la recherche des données complexe (sur les images). Pour atteindre cet objectif, nous avons proposé une version des arbres binaire paginé c'est à dire une création des conteneurs au niveau des feuille qui permis d'accéléré la recherche et aussi la construction d'index.
Avec cette version parallèle nous avons effectué quelque teste et nous avons réalisé que le système ne dépasse pas plus que 10.000 images ce que tous les chercheur du domaine d'indexation.
Bien que nous ayons accompli un certain nombre de contributions mentionnées précédemment d'autres améliorations sont encore possibles.

Comme suite de ce travail, on envisage de développer nos perspectives suivant :

1. concernons la création d'index :
   utilisation de l'arbre binaire équilibré (détermination des pivots).
   Création d'index distribué centralisé, la première machine crée l'index à base de nombre des machines connecté avec lui et après elle transfère à chaque machine une collection des données, a partir de ces

collection chaque machine de deuxième type crée un index local à base de taille des nœuds feuilles.

2. concernons les descripteurs d'index :

   Proposer une méthode plus efficace pour construire un index basé sur les descripteurs textuels.

3. Concernons le parallèle :

   Améliorer la technique de communication entre les machines pour garder l'efficacité de la création et assurer la vitesse du temps de réponse.

Bibliographie

- [1] Radia Abdi, intégration d'une application d'indexation dans un environnement Cloud, mémoire de master, 2012.
- [2] https://bu.univ-ouargla.dz/master/pdf/master-ABBASSI-MEFTAH.pdf, consulter le 11/03/2017.
- [3] CHERAA Saadeddine & BOUCETTA Samir, Etude comparative de quelques mesures de similarité, et leur application à la recherche d'images, mémoire de master, 2016.
- [4] MASTER ACADEMIQUE, recherche d'image par le cotenu, mémoire de majister.
- [5] Ben Cheikh Noura et Ben Bezziane Rima, LA RECHERCHE D'IMAGES PAR LA SEMANTIQUE, mémoire de master, 2011.
- [6] http://www-ia.lip6.fr/~tollaris/ARTICLES/THESE/node6.html#SECTION0212001000000000000, consulter le 25/04/2017.
- [7] http://www.01net.com/actualites/et-aussi-399644.html, consulter le 26/04/2017.
- [8] http://www-ia.lip6.fr/~tollaris/ARTICLES//Articles/DEA2003/node2.html, consulter le 26/04/2017.
- [9] Recherche d'information dans les images, pdf.
- [10] KOUAHLA Zineddine, Indexation dans les espaces métriques Index arborescent et parallélisassions, thèse de doctorat, 2013.
- [11] Daouda Traoré, Algorithmes parallèles auto-adaptatifs et applications, thèse de doctorat, 2008.
- [12] cours, algorithmique parallèle, master deux, 2016/2017.
- [13] Bastien Chopard. Parallélisme. Cour
- [14] cour systèmes distribués. Cour de 3eme Ingénieurs en informatique. https://fr.slideshare.net/adelessafi/cours-systeme-distribu
- [15] Bouhalit Naim, mémoire de master, 2016.
- [16] https://www.researchgate.net/publication/221338246_CoPhIR_Image_

Collection_under_the_Microscope?enrichId=rgreq-90f556c9-3b5a-4007-88bd-89ee30d80b26&enrichSource= Y292ZXJQYWdlOzIyMTMzODI0NjtBUzo5NzI4Mjk2MzgwNDE2M0AxNDAwMjA1NDc0MTI4&el=1_x_2 , consulter le 15/05/2017.

- [17] https://fr.wikipedia.org/wiki/Tri_%C3%A0_bulles, consulter le 05/06/2017.
- [18] Sana SELLAMI, Conception et Réalisation d'un outil de génération automatique de Mappage pour la transformation de documents XML, Diplôme d'Ingénieur, 2006.
- [19] dept-info.labri.fr/ENSEIGNEMENT/programmation2/intro-eclipse, consulter le 07/06/2017.