

# Graphes de voisinage pour l'indexation et l'interrogation d'images par le contenu

Hakim Hacid \*, Abdelkader Djamel Zighed\*

\*Université Lyon 2, Laboratoire ERIC  
Bat. L, 5 Av. Pierre Mendès-France  
69676 Bron cedex - France  
hhacid@eric.univ-lyon2.fr, zighed@univ-lyon2.fr,  
<http://eric.univ-lyon2.fr/>

**Résumé.** La découverte d'informations cachées dans les bases de données multimédias est une tâche difficile à cause de leur structure complexe et à la subjectivité liée à leur interprétation. Face à cette situation, l'utilisation d'un index est primordiale. Un index multimédia permet de regrouper les données selon des critères de similarité. Nous proposons dans cet article d'apporter une amélioration à une approche déjà existante d'interrogation d'images par le contenu. Nous proposons une méthode efficace pour mettre à jour, localement, les graphes de voisinage qui constituent notre structure d'index multimédia. Cette méthode est basée sur une manière intelligente de localisation de points dans un espace multidimensionnel. Des résultats prometteurs sont obtenus après des expérimentations sur diverses bases de données.

## 1 Introduction

La recherche d'information dans les bases de données image est toujours un défi. Pour l'être humain, l'accès à la sémantique d'une image est naturel et non explicite. Par conséquent, la sémantique provient de l'image sans processus cognitif explicite. Dans la vision par ordinateur, il existe plusieurs niveaux d'interprétation. Le plus bas est celui des pixels et le plus haut est celui des scènes ; entre eux beaucoup de niveaux d'abstraction existent. Le défi est alors de remplir la gouffre entre le bas niveau et le haut niveau.

Il existe au moins deux issues intermédiaires auxquelles nous nous intéressons. La première est la représentation de l'image sous forme de vecteurs qui est appelée indexation. Elle consiste à extraire quelques caractéristiques (composantes d'un vecteur) à partir de la représentation de bas niveau (Pixel). Par exemple, l'histogramme des couleurs, les différents moments, les paramètres de forme, etc. La seconde issue est l'ensemble des étiquettes associées à une image. Ces étiquettes sont fournies par l'humain au moyen de mots, d'adjectifs, ou au moyen de tout autre attribut symbolique. Les étiquettes sont compréhensibles et mieux manipulées. La sémantique peut être considérée comme le résultat du traitement des attributs symboliques qui sont liés à l'image.

Donner à l'ordinateur la capacité d'imiter l'être humain dans l'analyse de scènes nécessite d'explicitier le processus par lequel il peut se déplacer de la représentation bas niveau à la

représentation haut niveau de l'image. Les outils de traitement d'images offrent beaucoup de possibilités pour transformer une image en un vecteur de caractéristiques. Par exemple, le protocole MPEG-7 associe un ensemble d'attributs quantitatifs à chaque image. Le calcul de ces derniers est entièrement intégré et automatisé dans beaucoup de plates-formes logicielles. En retour, les étiquettes sont principalement données par l'utilisateur.

Dans la recherche d'images, l'utilisateur s'intéresse à trouver les images qui ont une sémantique assez proche de celle à laquelle il s'attend sans se soucier du vecteur de caractéristiques. Pour accéder aux images appropriées, l'utilisateur peut exprimer sa requête en utilisant soit des mots-clés (sous forme de langage naturel semblable à la recherche d'information sur Internet), ou par des images exemples. Aujourd'hui, l'annotation textuelle est un processus manuel et cher, c'est pourquoi l'approche par vecteurs est la plus souvent employée.

La pertinence du processus d'interrogation d'images dépend du vecteur de caractéristiques. Néanmoins, si nous supposons que les caractéristiques sont appropriées dans l'espace de représentation, censé être  $R^p$ , les images qui sont voisines devraient avoir des significations très semblables. Ceci est basé sur le principe général du "qui se ressemble s'assemble".

Nous proposons un cadre utilisant les caractéristiques de bas niveau, automatiquement extraites à partir des images, comme un moyen de navigation et d'interrogation de bases de données images. De cette façon, nous pouvons accéder à l'aspect sémantique d'une image en se basant sur les images voisines dans l'espace de représentation. Par conséquent, l'utilisateur peut rechercher des images semblables. Si les images à proximité d'une image requête sont étiquetées, alors l'image requête peut hériter des étiquettes disponibles dans le voisinage. Ce processus s'appelle l'apprentissage à base d'exemples. L'algorithme le plus connu dans cette catégorie est l'algorithme des *k-plus proches voisins*. Cependant, l'algorithme des *k plus proches voisins* a beaucoup d'inconvénients soulignés dans notre papier précédent (Scuturici et al., 2004). Nous préférons employer les graphes de proximité (Toussaint, 1980) qui ont des propriétés très intéressantes dans ce contexte pour atteindre notre objectif.

Les avantages de l'approche par graphes de proximité sont montrés dans (Scuturici et al., 2004). Cependant, cette approche souffre de quelques limitations liées principalement à la complexité de l'algorithme. En effet, pour chaque nouvelle requête, nous devons reconstruire le graphe en entier. Sachant que l'algorithme de construction des graphes de proximité a une complexité  $O(n^3)$  où  $n$  est le nombre d'individus, le problème devient ainsi insurmontable. Dans cet article, nous proposons un nouvel algorithme capable de calculer et de mettre à jour, localement, le voisinage exact d'un nouveau point. Ceci fournit un nouvel avantage à notre approche.

La section suivante présente une brève description de quelques travaux relatifs aux bases de données multimédias. Nous présentons le concept des graphes de proximité dans la section 3. La section 4 présente notre approche ainsi que notre méthode de mise à jour locale des graphes de proximité. Nous décrivons nos expérimentations et l'évaluation de la méthode dans la section 5. Une conclusion et des perspectives sont présentés dans la section 6.

## 2 Etat de l'art

L'interrogation des bases de données multimédias inclue trois fonctions principales, à savoir : la recherche, la navigation et le parcours ((Bolle et al., 1996)). Cette interrogation peut être effectuée sur deux niveaux (Rui et Huang (1999)) : le bas niveau (niveau structurel) et le

haut niveau (niveau sémantique). Le bas niveau utilise les caractéristiques, qui peuvent être extraites, à partir des données multimédias comme la couleur, la texture, la forme, etc. Le haut niveau quant à lui, consiste généralement en une liste de mots-clés qui est associée à la donnée multimédia qui sert à décrire son contenu sémantique. L'utilisation d'annotations textuelles présente deux inconvénients principaux : le premier est le fait que cette tâche est lente et très coûteuse ; la seconde est liée à la subjectivité de l'annotation des données multimédias. En effet, par exemple, deux personnes différentes peuvent annoter la même image de deux manières différentes.

A cause de ces inconvénients, l'interrogation est généralement faite en utilisant les caractéristiques de bas niveau. Dans toutes les approches d'interrogation de données multimédias (l'approche bas niveau ou l'approche haut niveau), chaque donnée est localisée par ses coordonnées dans un espace multidimensionnel  $R^p$ . Un vecteur de caractéristiques (caractéristiques de bas niveau ou annotations textuelles) est associé à chaque donnée. Rui et Huang (1999) estiment que l'interrogation par le contenu ne peut être effectuée de manière efficace uniquement en combinant les deux niveaux (bas niveau et haut niveau). Cependant, ceci peut soulever le problème de la subjectivité des annotations ce qui est un problème important qui peut détériorer considérablement les performances d'un système de recherche d'informations par le contenu.

Afin de capturer des aspects sémantiques à partir des caractéristiques de bas niveau, l'utilisation d'un index multimédia est nécessaire. Un index permet de regrouper des individus ayant des caractéristiques assez proches.

Plusieurs systèmes de recherche d'informations par le contenu se basent sur le principe des *k plus proche voisins* (Fix et Huges, 1951) en utilisant une mesure de similarité (Velkamp et Tanase, 2000). L'idée est de trier les individus de la base de données, en fonction de leur distance, par rapport à l'individu requête, et ensuite répondre à la requête en retournant un nombre  $k$  fixe d'individus les plus proches. Par exemple, le système QBIC, dans son implémentation pour le musée de l'Hermitage<sup>1</sup> (Faloutsos et al., 1994) renvoie les 12 plus proches images voisines de l'image requête. Les inconvénients d'une telle approche sont discutés dans Scuturici et al. (2004).

Le modèle de structuration des bases de données multimédias est (ou peut être vu) comme un graphe basé sur des relations de similitude entre les individus, par exemple *K-NN* (Mitchell, 1997) ou le graphe des voisins relatifs (Scuturici et al., 2004). L'objectif est d'explorer une base de données d'images par les similarités entre les images. Explorer les similarités peut être considéré comme la recherche des voisins des images requêtes.

Le modèle de structuration est très important car les performances d'un système de recherche d'informations par le contenu dépend fortement sur la structure de représentation (structure d'indexation) qui gère les données.

Plusieurs systèmes de recherche d'informations multimédias ont été proposés. Les systèmes de recherche d'images sont plus répandus que ceux pour la vidéo. Nous pouvons citer par exemple QBIC (Flickner et al., 1995; Niblack et al., 1993), CANDID (Kelly et al., 1995), CHABOT (Ogle et Stonebraker, 1995), VIRAGE (Ogle et Stonebraker, 1995), PhotoBook (Pentland et al., 1994), BlobWorld (Carson et al., 1999), VisualSeek (Chang et al., 1996; Smith et Chang, 1997) et RETIN (Fournier et al., 2001) pour les images, et CVEPS (Chang et al., 1996), JAKOB (La-Cascia et Ardizzone, 1996), VISION (Li et al., 1996), et SWIN (Zhang et al., 1995) pour la vidéo.

---

<sup>1</sup><http://www.hermitagemuseum.com/>

A partir de maintenant, nous considérerons le contexte de l'interrogation des grandes bases de données images par le contenu pour illustrer les propositions de cet article.

### 3 Graphes de voisinage

Les graphes de voisinage sont utilisés dans divers systèmes. Leur popularité est due au fait que le voisinage est déterminé par des fonctions cohérentes qui reflètent, d'un certain point de vue, le mécanisme de l'intuition humaine. Cependant, plusieurs problèmes relatifs au graphes de voisinage sont toujours d'actualité et exigent des travaux détaillés afin de les résoudre. Ces problèmes sont principalement liés à leur coût de construction élevé et à leurs difficultés de mise à jour. Pour cette raison, les optimisations sont nécessaires pour leur construction et leur mise à jour.

Afin d'éviter quelques problèmes liés à l'utilisation des  $K$ - $NN$  (problème de symétrie, subjectivité liée à la détermination du paramètre  $k$ ), l'utilisation d'un autre modèle de structuration basé sur les graphes de voisinage a été proposé dans ((Scuturici et al., 2004)). Cette proposition a beaucoup d'avantages, c'est pourquoi nous adoptons la même approche (l'utilisation des graphes de voisinage) pour l'interrogation d'images par le contenu. Nous allons dans ce qui suit présenter les graphes de voisinage.

Les graphes de voisinage ou graphes de proximité sont des structures géométriques qui utilisent le concept de voisinage pour déterminer les sommets les plus proches d'un sommet donné. Pour cela, ils se basent sur les mesures de "distances" (Toussaint (1991)). Nous allons utiliser les notations suivantes dans cet article :

Soit  $\Omega$  un ensemble de points dans un espace multidimensionnel  $R^d$ . Un graphe  $\mathcal{G}(\Omega, \varphi)$  est composé de l'ensemble de points  $\Omega$  et de l'ensemble d'arêtes  $\varphi$ . A chaque graphe nous pouvons associer une relation binaire  $R$  sur  $\Omega$ , dans laquelle un couple de points  $(\alpha, \beta) \in \Omega^2$  sont en relation binaire si et seulement si  $(\alpha, \beta) \in \varphi$ . En d'autres termes,  $(\alpha, \beta)$  sont en relation binaire si et seulement s'ils sont directement reliés dans le graphe  $\mathcal{G}$ . A partir de là, le voisinage  $V(\alpha)$  d'un point  $\alpha$  dans le graphe  $\mathcal{G}$ , peut être considéré comme un sous-graphe qui contient le point  $\alpha$  ainsi que tous les points qui sont directement relié à ce point.

Plusieurs possibilités ont été proposées pour la construction des graphes de voisinage. Nous pouvons citer la triangulation de Delaunay (Preparata et Shamos, 1985), le graphe des voisins relatifs (Toussaint, 1980), le graphe de Gabriel (Gabriel et Sokal, 1969) et l'arbre de recouvrement minimum (Preparata et Shamos, 1985). Dans cet article, nous considérons seulement l'un d'entre eux, le graphe des voisins relatifs ( $RNG$ ). La motivation principale pour ce choix est sa simplicité et sa large utilisation. Nous décrivons ci-après deux exemples de graphes de voisinage : le graphe des voisins relatifs ( $RNG$ ) et le graphe de Gabriel ( $GG$ ).

#### 3.1 Graphe de voisins relatifs

Dans un graphe de voisins relatifs  $\mathcal{G}_{rng}(\Omega, \varphi)$ , deux points  $(\alpha, \beta) \in \Omega^2$  sont des voisins s'ils vérifient la propriété de voisinage définie ci-après.

Soit  $\mathcal{H}(\alpha, \beta)$  une hyper-sphère de rayon  $\delta(\alpha, \beta)$  et de centre  $\alpha$ , et soit  $\mathcal{H}(\beta, \alpha)$  une hyper-sphère de rayon  $\delta(\beta, \alpha)$  et de centre  $\beta$ .  $\delta(\alpha, \beta)$  et  $\delta(\beta, \alpha)$  sont des mesures de similarité entre les deux points  $\alpha$  et  $\beta$ .  $\delta(\alpha, \beta) = \delta(\beta, \alpha)$ . Alors,  $\alpha$  et  $\beta$  sont des voisins si et seulement si la

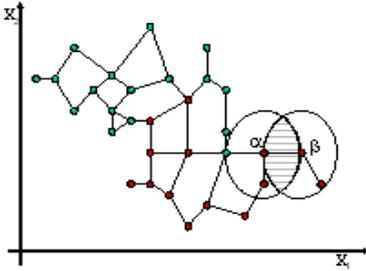


FIG. 1 – *Graphe de voisins relatifs*

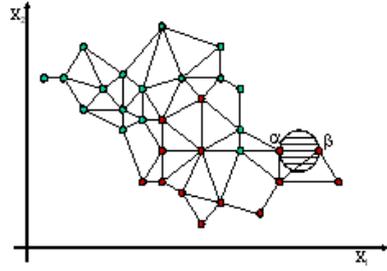


FIG. 2 – *Graphe de Gabriel*

lunule  $\mathcal{A}(\alpha, \beta)$  formée par l'intersection des deux hyper-sphères  $\mathcal{H}(\alpha, \beta)$  et  $\mathcal{H}(\beta, \alpha)$  est vide (Toussaint (1980)). Formellement :

$$\mathcal{A}(\alpha, \beta) = \mathcal{H}(\alpha, \beta) \cap \mathcal{H}(\beta, \alpha) \text{ Alors } (\alpha, \beta) \in \varphi \text{ Ssi } \mathcal{A}(\alpha, \beta) \cap \Omega = \phi$$

La figure 1 illustre le graphe des voisins relatifs.

### 3.2 Graphe de Gabriel

Ce graphe est proposé par Gabriel et Sokal (1969) dans un contexte de mesure de variations géographiques. Soit  $\mathcal{H}(\alpha, \beta)$  l'hyper-sphère de diamètre  $\delta(\alpha, \beta)$  (cf. figure 2). Alors,  $\alpha$  est le voisin de  $\beta$  si et seulement si l'hyper-sphère  $\mathcal{H}(\alpha, \beta)$  est vide. Formellement

$$(\alpha, \beta) \in \varphi \text{ Ssi } \mathcal{H}(\alpha, \beta) \cap \Omega = \phi$$

### 3.3 Algorithmes de construction des graphes de voisinage

Nous pouvons considérer deux situations quand nous traitons le problème d'optimisation des graphes de voisinage. La première situation est quand nous avons à disposition un graphe déjà construit. Dans cette situation, si nous utilisons une méthode d'approximation, nous risquons d'obtenir un autre graphe avec un voisinage de moindre qualité que l'existant, nous pouvons obtenir plus ou moins de voisins pour quelques individus. Dans ce cas, nous devons trouver une solution pour mettre à jour efficacement le graphe sans le reconstruire entièrement. La deuxième situation est celle où le graphe n'est pas encore construit. Dans cette situation nous pouvons appliquer une méthode d'approximation pour avoir un graphe qui est aussi similaire que possible à celui que nous pouvons obtenir en utilisant l'algorithme standard. Nous sommes intéressés dans cet article par le premier cas.

Plusieurs algorithmes pour la construction des graphes de voisinage ont été proposés. Les algorithmes que nous citons ci-après concernent la construction du graphe des voisins relatifs.

L'une des approches commune aux différents algorithmes est l'utilisation des techniques de raffinement  $\hat{z}$ . Dans ce type d'approches, le graphe est construit par étapes. Chaque graphe est construit à partir du graphe précédent, contenant toutes les connexions, en éliminant un certain nombre d'arrêtes qui ne vérifient pas la propriété de voisinage du graphe à construire.

L'élagage (élimination des arrêtes) se fait généralement en tenant compte de la fonction de construction du graphe ou à travers des propriétés géométriques.

Le principe de construction des graphes de voisinage consiste à chercher pour chaque point si les autres points de l'espace sont dans son voisinage. Le coût de cette opération est de complexité  $O(n^3)$  ( $n$  étant le nombre de points dans l'espace). Toussaint (Toussaint, 1991) a proposé un algorithme de complexité  $O(n^2)$ . Il déduit le *RNG* à partir d'une triangulation de Delaunay (Preparata et Shamos, 1985). En utilisant les voisins géographiques (Octant neighbors) Katajainen (1988) a proposé également un algorithme de complexité  $O(n^2)$ . Smith (1989) a proposé un algorithme de complexité  $O(n^{23/12})$  qui est moins importante que  $O(n^3)$ .

En ce qui nous concerne, l'approche que nous proposons est une amélioration de celle déjà proposée dans Scuturici et al. (2004). En effet, avec l'ancienne méthode, le graphe n'est pas vraiment mis à jour. Les voisins d'un individu requête sont considérés comme étant les voisins de son plus proche voisin. Cette approche n'est pas correcte car dans un espace multidimensionnel et avec les contraintes géométriques à respecter, les voisins d'un individu ne peuvent être ceux de son voisin le plus proche. Ainsi, en utilisant cette méthode le graphe sera inévitablement détérioré. Nous proposons dans ce qui suit une méthode de mise à jour locale efficace qui est stable et insensible aux effets de la dimension des données).

## 4 Recherche d'informations par le contenu : approche par graphes de voisinage

L'interrogation des bases de données images est généralement faite par la soumission d'une requête au système, cette requête est généralement sous forme d'image, le système pré-traite (segmente, égalise, etc.) la requête et produit un vecteur de descripteurs qui représente un point dans un espace multidimensionnel. Ce point est inséré dans la structure de représentation (structure d'indexation) et ses voisins sont alors retournés comme une réponse à la requête.

Dans notre cas, une approche naïve en utilisant les graphes de voisinage serait la reconstruction du graphe de voisinage qui contient les données déjà existantes dans la base de données tout en ajoutant l'individu requête. Cette approche n'est, malheureusement, pas appropriée car elle est très coûteuse particulièrement quand le nombre d'individus dans la base de données est important. Une autre approche est de mettre à jour localement le graphe de voisinage, c'est-à-dire, trouver une manière de telle sorte que seuls les individus potentiellement voisins soient affectés par la possible modification ou interrogation.

La tâche de mise à jour locale des graphes de voisinage passe par la localisation du point inséré aussi bien que les points qui peuvent être affectés par la mise à jour. Pour cela, nous procédons en deux étapes principales : nous recherchons d'abord une surface optimale de l'espace de représentation pouvant contenir un nombre maximum de points potentiellement voisins au point requête. La deuxième étape est réalisée dans le but de filtrer les individus trouvés préalablement afin de récupérer les vrais voisins en considérant une propriété de voisinage. Cette dernière étape cause la mise à jour effective des relations de voisinage entre les points concernés.

L'étape principale dans cette méthode est la détermination de la surface de recherche. Ceci peut être considéré comme un problème de détermination d'une hyper-sphère ayant pour centre

le point requête  $\alpha$  maximisant les chances de contenir les voisins du point requête tout en réduisant au minimum le nombre de points qu'elle contient.

Nous tirons profit de la structure générale des graphes de voisinage afin d'établir le rayon de l'hyper-sphère. Nous nous concentrons particulièrement sur le concept du voisin le plus proche et le concept du voisin le plus éloigné. Ainsi, deux observations en relation avec ces deux concepts nous semblent intéressantes :

- Les voisins du voisin le plus proche du point requête sont des candidats potentiels au voisinage du point requête.

A partir de là et par généralisation, nous pouvons déduire que :

- Tous les voisins directs d'un point sont également des candidats au voisinage d'un point requête pour lequel il est voisin.

Cernant la première étape, le rayon de l'hyper-sphère ayant les propriétés citées ci-dessus est celui comprenant tous les voisins du plus proche voisin de la requête. Ainsi, en considérant que l'hyper-sphère est centrée dans  $\alpha$ , son rayon est égale à la somme des distances entre le point requête  $\alpha$  et de son plus proche voisin et celle entre le voisin le plus proche et son voisin le plus éloigné.

Le contenu de l'hyper-sphère est traité pour vérifier s'il existe quelques voisins (ou tous les voisins). La deuxième étape constitue une étape de renforcement et vise à éliminer le risque de perdre des voisins ou d'en inclure des faux. Cette étape procède de telle sorte à tirer profit de la deuxième observation. Ainsi, nous prenons tous les vrais voisins du point requête, récupérés précédemment (ceux retournés dans la première étape), ainsi que que leurs voisins et mettons à jour les relations de voisinage entre ces points.

Considérons alors  $\alpha$  le point requête et  $\beta$  son plus proche voisin avec une distance  $\delta_1$ . Considérons aussi  $\lambda$  le voisin le plus loin de  $\beta$  avec une distance  $\delta_2$ . Le rayon  $SR$  de l'hyper-sphère peut être exprimé avec la formule suivante :

$$SR = \delta_1 + \delta_2 + \epsilon$$

$\epsilon$  est un paramètre de relaxation, il peut être fixé selon l'état des données (leur dispersion par exemple) ou par la connaissance du domaine. Nous avons fixé expérimentalement ce paramètre à 1.

La complexité de cette méthode est très basse et se rejoint parfaitement notre objectif de départ (localisation des voisins d'un point dans un temps très court). Elle est exprimée par :

$$O(2n + n'^2)$$

avec

- $n$  :le nombre d'individus dans la base de données.
- $n'$  :le nombre d'individus dans l'hyper-sphère ( $\ll n$ ).

Cette complexité inclue les deux étapes décrites précédemment, à savoir, la recherche du rayon de l'hyper-sphère et la mise à jour du voisinage dans l'hyper-sphère. Le deuxième terme correspond au temps nécessaire pour la mise à jour effective des relations de voisinage entre les vrais voisins. Le temps nécessaire pour cette opérations est très faible. Cette complexité constitue la complexité maximale et peut être optimisée de plusieurs manières. La plus simple est d'employer un algorithme plus robuste pour la recherche du plus proche voisin à la place du parcours séquentiel. L'exemple ci-après illustre et récapitule le principe de la méthode.

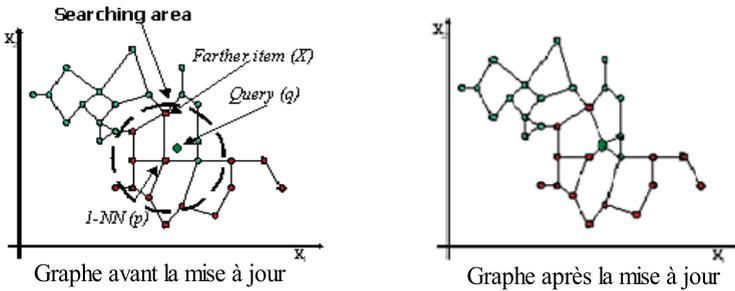


FIG. 3 – Illustration du principe de la méthode proposée

## 5 Expérimentations et résultats

L'intérêt des graphes de voisinage pour la recherche d'informations par le contenu est montrée et discutée dans Scuturici et al. (2004). Des comparaisons ont été effectuées avec différentes configurations du  $K$ - $NN$  et des résultats concluants ont montré l'utilité des graphes de voisinage dans ce domaine.

En ce qui nous concerne, nous sommes intéressés dans la présente section par deux différents types d'évaluation : évaluation de la validité des résultats obtenus par l'utilisation de la méthode proposée et l'évaluation des temps de réponse. Nous utilisons pour cela une base de données d'images (Nene et al., 1996). Cette base de données contient 7200 images représentant 100 objets différents pris avec sous diverses vues. Pour réaliser nos expérimentations, nous pré-traitons la base de données en appliquant quelques algorithmes de traitement d'image pour extraire certaines caractéristiques. Nous employons particulièrement la couleur, quelques caractéristiques de texture et de forme. Chaque image est représentée sous forme d'un vecteur de caractéristiques. Un graphe de voisinage est alors construit en utilisant les vecteurs. L'interrogation est également faite de la même manière en employant un vecteur de caractéristiques. Ainsi, chaque image est représentée comme un point dans un espace multidimensionnel.

Notre intérêt en effectuant ces expériences est de montrer que notre méthode donne les mêmes résultats que ceux obtenus en construisant un graphe de voisinage avec un algorithme standard (qualité des résultats).

Pour vérifier la validité des résultats obtenus, nous avons besoin d'un graphe de référence. Pour cela, nous prenons la base de données entière (vecteurs de descripteurs) et nous construisons un graphe de voisins relatifs. Après avoir établi le graphe de référence, nous pouvons commencer les tests de validité. Nous prenons arbitrairement un individu de la base de données et nous construisons un nouveau graphe de voisins relatifs avec les individus restant dans la base de données (les  $n - 1$  individus). Après cela, nous insérons l'individu préalablement retiré dans le graphe en employant la méthode proposée. Les voisins du point inséré sont alors comparés aux voisins de ce même individu dans le graphe de référence. Cette opération est répétée plusieurs fois en prenant différents individus à chaque itération.

Le Tableau 1 illustre un voisinage obtenu en insérant des individus aléatoirement pris de la collection d'image. Chaque individu dans l'ensemble de données est identifié par un identifiant

unique. La première colonne de chaque tableau constitue l'individu requête. Nous pouvons clairement voir que les résultats obtenus après l'insertion de l'individu restant (Graphe 2), sont exactement les mêmes que ceux obtenus en employant l'algorithme standard (Graphe 1). Ainsi, nous ne perdons pas de voisins. Une évaluation globale (utilisant les mesures de rappel et de précision) est donnée dans (Hacid et Zighed (2005)).

Graphe standard (Graphe 1)		Insertions locales (Graphe 2)	
Image requête	Images trouvées	Image requête	Images trouvées
IMG85	IMG74 IMG87 IMG118 IMG131 IMG3665	IMG85	IMG74 IMG87 IMG118 IMG131 IMG3665
IMG4804	IMG484 IMG623 IMG7170 IMG4763 IMG4803 IMG4805 IMG6532	IMG4804	IMG484 IMG623 IMG7170 IMG4763 IMG4803 IMG4805 IMG6532
IMG6807	IMG4345 IMG6781 IMG6791 IMG6795 IMG6825 IMG6830	IMG6807	IMG4345 IMG6781 IMG6791 IMG6795 IMG6825 IMG6830
IMG7196	IMG74 IMG87 IMG7171 IMG7195 IMG7197	IMG7196	IMG74 IMG87 IMG7171 IMG7195 IMG7197

**TAB. 1** – Comparaison du voisinage obtenu en utilisant un graphe de référence (Graphe 1) et en utilisant des insertions locales (Graphe 2)

La validité des résultats obtenus étant acquise, à présent nous sommes intéressés par les temps de réponse en utilisant la méthode proposée, c'est-à-dire, les temps que la méthode prend pour insérer un individu requête dans une structure existante. Nous employons le même protocole que celui décrit précédemment, mais au lieu de récupérer les individus voisins, nous prenons en considération uniquement les temps de réponse de chaque requête. Nous utilisons pour ces expériences une machine avec un processeur INTEL Pentium 4 (2.80 Gz) et 512 Mo de mémoire. Les temps de réponse de 10 insertions, sont montrés dans le graphique dans la figure 4.

Les temps de réponse (exprimés en millisecondes) sont intéressants en considérant le volume de données utilisé, ils sont variable d'un individu à un autre, ceci est due particulièrement

Graphes de voisinage pour l'indexation et l'interrogation d'images par le contenu

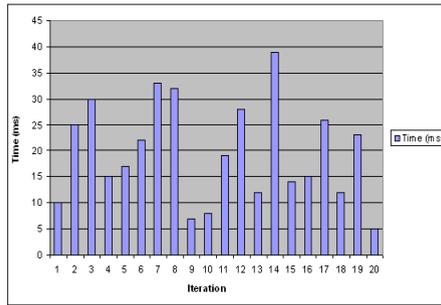


FIG. 4 – Illustration des temps d'exécution en utilisant 10 individus

au fait que différents nombre d'individus candidats pour le voisinage sont utilisés à chaque itération. D'une manière générale, l'insertion d'un nouvel article dans le graphe en utilisant notre méthode nécessite seulement 50 millisecondes en moyenne, et nécessite en moyenne 40 minutes en utilisant l'algorithme standard dans les mêmes conditions.

## 6 Conclusion et travaux futurs

La recherche d'informations par le contenu dans les bases de données multimédias est une tâche complexe en raison de, principalement, la nature des données multimédias et la subjectivité liée à leur interprétation. L'utilisation d'une structure d'index appropriée est primordiale.

Dans cet article nous nous sommes intéressés aux graphes de voisinage qui constituent notre structure d'indexation, nous avons proposé une méthode pour mettre à jour localement les graphes de voisinage dans un but d'apporter une amélioration à une approche existante. Notre méthode est basée sur la localisation des individus potentiels, qui peuvent être affectés par la mise à jour. Les expérimentations effectuées sur différents ensembles de données ont montré l'efficacité et l'utilité de la méthode proposée.

En tant que travaux futurs, nous projetons de fixer le problème du paramètre de relaxation en proposons une fonction de détermination automatique en tenant compte de quelques statistiques comme la dispersion. Par ailleurs, nous envisageons d'étendre cet algorithme afin de mettre en place une version incrémentale de construction des graphes de voisinage. L'intégration de cette approche et son application dans plus de fonctions liées aux données multimédias comme l'annotation automatique et la classification de données multimédias constituent une suite logique de ce travail.

## Références

- Bolle, R. M., B.-L. Yeo, et M. M. Yeung (1996). Video query : Beyond keywords. Research Report RC 20586, IBM.
- Carson, C., M. Thomas, S. Belongie, J. M. Hellerstein, et M. Jitendra (1999). Blobworld : A system for region based image indexing and retrieval. *Visual information and informa-*

- tion systems, proceedings of the third international conference VISUAL'99, Amsterdam, the Netherlands.*
- Chang, S.-F., J. R. Smith, et J. Meng (1996). Efficient techniques for feature-based image/video access and manipulation. In *Data Processing Clinic*.
- Faloutsos, C., R. Barber, M. Flickner, J. Hafner, W. Niblack, D. Petkovic, et W. Equitz (1994). Efficient and effective querying by image content. *J. Intell. Inf. Syst.* 3(3/4), 231–262.
- Fix, E. et J. L. Huges (1951). Discriminatory analysis - non parametric discrimination : consistency properties. Technical Report 21-49-004, USAF School of aviation medicine, Randolph Field, Texas.
- Flickner, M., H. S. Sawhney, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, et P. Yanker (1995). Query by image and video content : The qbic system. *IEEE Computer* 28(9), 23–32.
- Fournier, J., M. Cord, et S. Philipp-Foliguet (2001). Retin : A content-based image indexing and retrieval system. *Pattern Anal. Appl.* 4(2-3), 153–173.
- Gabriel, K. R. et R. R. Sokal (1969). A new statistical approach to geographic variation analysis. *Systematic zoology* 18, 259–278.
- Hacid, H. et A. D. Zighed (2005). An effective method for locally neighborhood graphs updating. In *DEXA*, pp. 930–939.
- Katajainen, J. (1988). The region approach for computing relative neighborhood graphs in the lp metric. *Computing* 40, 147–161.
- Kelly, P. M., T. M. Cannon, et D. R. Hush (1995). Query by image example : The comparison algorithm for navigating digital image databases (candid) approach. In *Storage and Retrieval for Image and Video Databases (SPIE)*, pp. 238–248.
- La-Cascia, M. et E. Ardizzone (1996). Jakob : Just a content based query system for video databases. *ICASSP, Atlanta*.
- Li, W., S. Gauch, J. M. Gauch, et K. M. Pua (1996). Vision : A digital video library. In *Digital Libraries*, pp. 19–27.
- Mitchell, T. M. (1997). Machine learning meets natural language. In *EPIA*, pp. 391.
- Nene, S. A., S. K. Nayar, et H. Murase (1996). Columbia object image library (coil-100). Technical report, Technical Report CUCS-006-96.
- Niblack, W., R. Barber, W. Equitz, M. Flickner, E. H. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, et G. Taubin (1993). The qbic project : Querying images by content, using color, texture, and shape. In *Storage and Retrieval for Image and Video Databases (SPIE)*, pp. 173–187.
- Ogle, V. E. et M. Stonebraker (1995). Chabot : Retrieval from a relational database of images. *IEEE Computer* 28(9), 40–48.
- Pentland, A., R. W. Picard, et S. Sclaroff (1994). Photobook : Tools for content-based manipulation of image databases. In *Storage and Retrieval for Image and Video Databases (SPIE)*, pp. 34–47.
- Preparata, F. et M. I. Shamos (1985). *Computational Geometry-Introduction*. New-York : Springer-Verlag.

- Rui, Y. et T. S. Huang (1999). Image retrieval : current techniques, promising directions, and open issues. *Journal of Visual Communication and Image Representation* 10(1), 39–62.
- Scuturici, M., J. Clech, V. M. Scuturici, et D. Zighed (2004). Topological representation model for image databases query. *Journal of Experimental and Theoretical Artificial Intelligence (JETAI)*, 145–160.
- Smith, J. R. et S. F. Chang (1997). Querying by color regions using the visualseek content-based visual query system. *M. T. Maybury, editor, Intelligent Multimedia Information Retrieval*. AAAI Press.
- Smith, W. D. (1989). Studies in computational geometry motivated by mesh generation. *PhD thesis, Princeton University*.
- Toussaint, G. T. (1980). The relative neighborhood graphs in a finite planar set. *Pattern recognition* 12, 261–268.
- Toussaint, G. T. (1991). Some insolved problems on proximity graphs. *D. W Dearholt and F. Harrary, editors, proceeding of the first workshop on proximity graphs. Memoranda in computer and cognitive science MCCS-91-224. Computing research laboratory. New Mexico state university Las Cruces*.
- Veltkamp, R. C. et M. Tanase (2000). Content-based image retrieval systems : A survey. Technical Report UU-CS-2000-34, Department of Computing Science, Utrecht University.
- Zhang, H. J., C. Y. Low, S. W. Smoliar, et J. H. Wu (1995). Video parsing retrieval and browsing : An integrated and content-based solution. *Proc. Of ACM Multimedia'95., San Francisco*.

## Summary

Retrieving hidden information in multimedia data is a difficult task according to their complex structure and the subjectivity related to their interpretation. In this situation the use of an index is primordial. A multimedia index makes it possible to group data according to similarity criteria. We propose in this paper to bring an improvement to an existing content based image retrieval approach. We propose an effective method for locally updating neighborhood graphs which constitute our multimedia index. This method is based on an intelligent way for locating points in a multidimensional space. Promising results are obtained after experimentations on various databases.