

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Saad Dahlab Blida



Faculté des sciences

Département d'informatique

Mémoire Présenté par :

Laïmouche Amine

Rechachou Hamza

En vue d'obtenir le diplôme de master

Domaine : Mathématique et informatique

Filière : Informatique

Spécialité : Informatique

Option : Ingénierie de logiciel

Sujet :

Annotation automatique d'images basée sur différentes méthodes de classification

Soutenu le : 20/09/2015

M. FERFARA SOFIAN	Président
Mme. M.ARKAM.	Examineur
Mme. N.ZERF.	Examineur
Mme. BENBLIDIA Nadjia	Promotrice
Mme. Bakalem Mahdia	Encadrante

Promotion
2014 / 2015

Remerciements

« Tous d'abord nous remercions dieu le tout puissant de nous avoir donné le savoir et l'opportunité de pouvoir poursuivre nos études »

Nous exprimons nos sincères remerciements et notre profonde gratitude à notre promotrice Mlle Najia Benblidia et l'encadrante Mlle Bakalem Mahdia pour la confiance qu'elle nous a témoignée en nous proposant ce sujet, sa disponibilité durant toute la période du projet, ses encouragements et sa patience, les discussions scientifiques qu'il a su générer, ses remarques et ses suggestions nous ont permis de réaliser ce travail.

Nous tenant aussi à remercier le personnel de l'USDB ainsi que toute les personnes travaillants aux niveaux de la faculté des sciences.

Pour l'aide précieuse qu'il nous a fourni et l'enseignement qu'il nous a dispensé durant nos années de formation.

Nous remercions infiniment le

Nous remercions s'adressent à tous les membres du jury pour l'honneur qu'ils nous font en acceptant de juger notre travail.

Nos remerciements vont aussi à tous ceux et celles qui ont participé de près ou de loin à l'élaboration de ce travail, à tous nos amis pour leur soutien moral tout au long de la préparation de ce mémoire.

Dédicace

*Dieu le tout puissant, qui m'avez la force de continuer sur ce chemin difficile et
le courage de poursuivre mes études.*

*Je dédie ce travail à mes chers parents qui m'ont toujours soutenu et encouragé
dans mes choix, je ne le dis pas peut être pas suffisamment mais merci pour tous
ce que vous avez fait pour moi depuis ma naissance je vous en serai éternellement
redevable.*

*A Mes cher frère et sœurs qui ont su me supporter durant toutes ces années, vous
êtes mes exemples.*

AMEN

Dédicace

Ce n'est qu'à grâce à l'aide d'ALLAH que j'ai finalisé cet humble travail que je dédie à toutes les personnes qui me sont chères et à qui je tiens énormément en reconnaissance de tout le soutien qu'elles m'ont apporté durant les moments difficiles.

À mes très chers parents, sans qui je n'aurais jamais vu la lumière du jour, ni devenu ce que je suis. Puissent-ils trouver en ce modeste travail toute la gratitude et la reconnaissance d'un fils dévoué, ma mère pour son dévouement et ses sacrifices et mon père pour son écoute et ses précieux conseils.

À mes frères adorés.

À mes sœurs.

À toute ma famille.

À mon amis Amine et sa famille.

À tous mes amis

À tous ceux qui m'aiment et qui me connaissent, à tous ceux pour qui je compte.

HAMZA

Résumé :

Le nombre d'images ne cesse d'augmenter. Face à cette croissance des données multimédia, il devient de plus en plus difficile à un utilisateur de retrouver l'information recherchée, d'où le besoin de développer des systèmes d'annotation qui permettent d'améliorer la recherche sur le Web, c'est la raison pour laquelle nous avons abordé dans ce mémoire l'annotation automatique d'images.

L'annotation automatique d'images consiste à assigner la sémantique à une image en associant des mots clés qui décrivent cette sémantique.

Dans le cadre de notre travail nous sommes intéressés de faire une étude comparative sur l'annotation automatique d'images basées sur différentes techniques de classifications.

Mots-clés: Image, Annotation automatique, techniques de classification.

المخلص:

إن عدد الصور أخذ في الازدياد, و نظرا لهذا النمو من بيانات الوسائط المتعددة, أصبح من الصعب على المستخدم العثور على المعلومات المطلوبة, ومن هنا تبرز الحاجة إلى تطوير أنظمة الشرح التي من شأنها تحسين البحث على شبكة الإنترنت, و هو سبب تناولنا لأنظمة الشرح التلقائية للصور في مذكرتنا.

الشرح التلقائي للصور هو تعيين دلالات على صورة من خلال ربط الكلمات الرئيسية التي تصف هذه الصورة. في إطار البحث الذي نقوم به, نهدف إلى القيام بدراسة تشمل المقارنة بين عدد من أنظمة الشرح التلقائي القائم على أساس مختلف طرق التصنيف.

كلمات رئيسية: الشرح التلقائي, صورة, طرق التصنيف.

Abstract :

The number of images is increasing. Given this growth of multimedia data, it becomes increasingly difficult for a user to find the information sought, as a consequence, the need to develop annotation systems that improve the web search that is why we have addressed in this brief automatic image annotation.

The automatic annotation of images is to assign semantics to an image by associating keywords that describe this semantic.

As part of our work we are interested to do a comparative study of the automatic annotation of images based on different technical classifications.

Keywords: Image, Automatic Annotation, classification techniques.

Table de matières

Introduction générale.....	1
Chapitre : I	Annotation des images
1. Introduction.....	4
2. Image numérique.....	5
2.1. Définition.....	5
2.2. Caractéristiques.....	5
3. Méthodes d'annotations.....	7
4. Les approches d'annotation automatique des images.....	9
4.1. L'annotation automatique à partir du texte associé à l'image.....	10
4.2. L'annotation automatique à partir du contenu visuel de l'image.....	10
5. Processus d'auto-annotation des images.....	10
5.1. Apprentissage.....	12
5.2. Traitement d'une nouvelle image.....	16
6. conclusion.....	16
Chapitre : II	Technique de classification
1. Introduction.....	18
2. Classification.....	18
3. Les critères de classification.....	19
4. Les méthodes de classification.....	20
4.1. Les méthodes monodimensionnelles	20
4.2. Les méthodes multidimensionnelles.....	21
4.2.1. Classification supervisée.....	21
4.2.2. Classification non supervisée.....	25
6. Conclusion.....	36

Liste des tableaux

Tableau 1.1: Avantages et inconvénient des différents types d'annotation	7
Tableau 4.1: Description des classes.....	47
Tableau 5.1: Résultats d'annotation pour huit images.....	68
Tableau 5.2: Résultat de temps d'exécution K-means.....	69

Liste des figures

Figure 1.1 : Présentation des phases du système d'auto-annotation	11
Figure 1.2 : Architecture de module de segmentation.....	13
Figure 1.3 : Les modules des calculs des caractéristiques des régions.....	14
Figure 1.4 : Schéma d'apprentissage : Construction de blobs.....	15
Figure 2.1 : Différents regroupements selon différentes distances.....	19
Figure 2.2 : Hyperplan séparateur.....	23
Figure.2.3 : La relation entre l'hyperplan optimal et l'hyperplan valide.....	24
Figure 2.4 : Illustration de regroupement en clusters.....	25
Figure 2.5: déroulement de K- means.....	28
Figure 2.6 : Mise à jour de la (BMU) et ses voisins.....	32
Figure 2.7 : Le rayon de voisinage.....	33
Figure 2.7 : Grille SOM rectangulaire (gauche) et grille SOM hexagonale (droite).....	34
Figure 2.8 : Classification hiérarchique ascendante.....	35
Figure 4.1 : Diagramme de cas d'utilisation global.....	44
Figure 4.2 : Diagramme de cas d'utilisation Apprentissage.....	45
Figure 4.3 : Diagramme de cas d'utilisation annotation de nouvelle images.....	45
Figure 4.4 : Diagramme des classes au niveau logique.....	46
Figure 4.5: La fenêtre principale du système.....	49
Figure 4.6: chargement de nouvelle image.....	50
Figure 4.7: résultat de niveau de gris.....	51
Figure 4.8: calcule de filtre médian.....	52
Figure 4.9: Résultat de filtre médian.....	52

Chapitre : III Etat de l'art

1. Introduction.....	38
2. Annotation avec les réseaux bayésiens.....	38
3. Annotation avec la méthode de k plus proche voisins.....	39
4. Annotation avec séparateur à vast marges SVM	39
5. Annotation avec k-means.....	40
6. Auto-annotation avec les cartes auto-organisatrice (SOM).....	40
7. Conclusion.....	42

Chapitre : IV Conception et mise en œuvre du système AIA

1. Introduction.....	44
2. Conception.....	44
2.1. Définition d'UML.....	44
2.2. Diagramme de cas d'utilisation.....	44
2.3. Diagramme des classes.....	45
3. Mise en œuvre d'un système d'auto-annotation d'images.....	48
3.1. Corpus Benchmark.....	48
3.2. Système d'auto-annotation d'images à base de classification.....	49
3.2.1. Phase d'apprentissage.....	50
3.2.2. Phase d'auto-annotation de nouvelle image.....	56
4. Conclusion.....	59

Chapitre : V Tests et validation des résultat

1. Introduction.....	61
2. Apprentissage.....	61
3. Annotation des images de test.....	61
4. résultats d'expérimentation.....	68
5. Interprétation des résultats.....	68
6. Temps d'exécution de l'algorithme de segmentation k-means.....	69
7. Conclusion.....	70
Conclusion général	70

Figure 4.10: résultat de la segmentation.....	54
Figure 4.11: résultat de la caractérisation.....	54
Figure 4.12: résultat de la catégorisation.....	55
Figure 4.13: résultat d'auto-annotation des blobs.....	56
Figure 4.14: résultat d'affectation des régions au blobs.....	57
Figure 4.15: Résultat d'annotation selon <i>k-ppv</i>	57
Figure 4.16: Résultat d'annotation selon <i>k_means</i>	58
Figure 4.17: Résultat d'annotation selon FCM.....	58
Figure 4.18: Résultat final d'annotation	58
Figure 5.1 : Histogramme de comparaisons entre les différentes classifications.	68

Liste des algorithmes :

Algorithme 2.1 : Algorithme de classification <i>k-ppv</i>	22
Algorithme 2.2 : Algorithme de classification <i>K means</i>	28
Algorithme 2.3 : Algorithme FCM.....	29
Algorithme 2.4 : Algorithme SOM.....	31
Algorithme 2.5 : Algorithme de classification hiérarchique ascendante.....	35
Algorithme 4.1: Algorithme de segmentation <i>k-means</i>	53

Introduction générale

Vu le développement des domaines tels que la médecine, les images satellitaires et la cartographie... les images numériques sur le Web n'arrêtent pas de se développer. Ainsi, la richesse de l'image en informations visuelles et sémantiques pose le problème du fossé sémantique dans la recherche d'images sur le Web. Pour cela La technique d'annotation automatique des images permet d'améliorer la recherche sur le Web. Cette technique consiste à assigner la sémantique à une image en associant des mots clés qui décrivent cette sémantique.

La classification est une étape primordiale dans l'apprentissage du processus d'auto-annotation d'image, elle permet de grouper les individus en exploitant leur similarité. Les individus peuvent être : les pixels, une sous image de taille fixe ou une région. Il existe plusieurs méthodes de classification, une bonne méthode de classification sera donc celle qui permettra d'arriver à classer les régions selon une similarité visuelle. Pour cela le résultat de classification influe d'une manière indirecte sur la qualité d'annotation. Le problème de classification est localisé sur la définition du nombre de cluster ainsi que le critère de classement.

Problématique :

Notre problème principal est d'améliorer la qualité d'annotation d'images, en répondant à la question suivante : quelle est la bonne méthode de classification utilisée afin de donner une bonne interprétation d'image à travers ces annotations. Nous envisageons de faire une étude comparative sur les annotations d'images basées sur différentes techniques de classification.

Objectif :

Pour cela, il faut :

- Faire une étude sur les méthodes de classifications utilisées dans le processus d'annotation d'images.
- Implémenter les différentes méthodes de classifications.

- Comparer les résultats obtenus

Organisation du mémoire

Ce mémoire est organisé en cinq chapitres :

- **Chapitre I :**

Ce chapitre présente les notions de base de l'annotation des images ainsi que les différentes étapes du processus d'annotation.

- **Chapitre II:**

Ce chapitre décrit la classification et leurs différentes méthodes utilisées dans le domaine de traitement des images.

- **Chapitre III:**

Le chapitre 3 présente un état de l'art sur les annotations automatique des images à base des méthodes de classification.

- **Chapitre IV:**

Ce chapitre concerne la conception et la mise en œuvre de notre application en précisant les différentes phases de système et les outils utilisé dans le développement de ce dernier.

- **Chapitre V:**

Ce dernier chapitre présente le test et la validation des annotations obtenus d'un échantillon d'image de test pour chaque méthode de classification. Ainsi que l'étude comparative entre les différents résultats obtenus afin de prédire l'influence des méthodes de classification dans le processus d'annotation automatique des images.

Chapitre : 1

Annotation des images

1. Introduction :

L'image est une représentation d'êtres, d'objets ou de scènes par les photographies, la démocratisation de l'informatique et l'évolution des réseaux de transmission, en particulier l'internet qui amène à partager et à manipuler un flot impressionnant d'images et de données multimédia a donné à toute personne la possibilité d'accéder à un nombre incalculable d'images ce qui n'était pas possible auparavant. C'est pour cette raison que l'image est devenue l'un des principaux supports de l'information.

Avec le développement d'internet et l'évolution technologique nous participons à un accroissement phénoménal du nombre de données multimédia.

En effet une fois ces images stockées, le problème est de pouvoir les retrouver le plus simplement et le plus rapidement possible et d'en faciliter l'accès. Tout ça a donné naissance à un nouveau domaine qui est la recherche et l'annotation d'images par le contenu. Ce domaine là a pour objectif le développement d'outils qui permettent de retrouver les images, en se basant sur leur contenu visuel (couleur, texture, etc.). La majorité des systèmes existants aujourd'hui utilisent les annotations qui accompagnent les images pour pouvoir indexer ces derniers.

L'annotation automatique des images permet d'associer automatiquement des mots-clés à des images à partir de leur contenu. Elle consiste à désambiguïser et interpréter les images d'une façon qui assure leur exploitation adéquate par la machine.

Par conséquent, L'objectif fondamental de l'auto-annotation des images est l'amélioration de la recherche des images sur le Web.

Dans ce chapitre, nous allons d'abord définir l'image, ses caractéristiques et l'annotation des images ainsi que ses différents types, puis nous allons présenter les différentes approches d'annotation. Enfin, nous allons décrire le processus d'annotation automatique d'images(AIA).

2. Image numérique :

2.1. Définition

Une image numérique est composée d'un ensemble de points, appelés « pixels », chaque pixel est désigné par ses coordonnées entières. Elle est créée, traitée, stockée sous forme binaire (suite de 0 et de 1). C'est cette forme numérique qui permet une exploitation ultérieure par des outils logiciels. [1]

2.2. Caractéristiques :

L'image est un ensemble structuré d'informations caractérisé par les paramètres suivants :

2.2.1. La Couleur

La couleur est l'une des plus reconnaissables éléments du contenu visuel d'une image, c'est une caractéristique très utilisée pour l'annotation et la recherche d'images. Il existe plusieurs descripteurs de couleur tel que: les histogrammes, les espaces de couleur et les moments de couleurs, etc. [2]

L'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.

Espace de couleur

Une image est composée de pixels, chaque pixel d'une image peut être représenté comme un point dans un espace de couleurs à 3 dimensions (généralement). Les plus utilisés sont: **RVB** (Rouge, Vert et Bleu), **TSL** (Teinte, Saturation, Luminosité)...etc [2]

Les moments statistiques

L'histogramme utilise la distribution complète de la couleur ce qui engendre une énorme quantité de données à stocker et par conséquent qui prend plus de temps et de mémoire. Pour résoudre cet inconvénient au lieu de calculer la distribution complète de l'histogramme on calcule seulement des dominantes caractéristiques de couleurs tels que l'espérance, la variance et d'autres moments. Il est possible de calculer l'espérance, la variance, les moments sur chaque composante couleur par la formule suivante [3]:

- Espérance = $\frac{1}{N} \sum_{j=1}^N P_j$
- variance = $(\frac{1}{N} \sum_{j=1}^N (P_j - E)^2)^{1/2}$
- moments d'ordre trois = $(\frac{1}{N} \sum_{j=1}^N (P_j - E)^3)^{1/3}$

2.2.2. La texture

La texture est étudiée depuis maintenant une vingtaine d'années. En effet, suite aux nombreuses études qui ont été réalisées, la texture est devenue l'une des caractéristiques qui est de plus en plus utilisée dans la recherche et l'annotation d'image. Il existe plusieurs approches pour analyser la texture, Haralick [4] propose une analyse des textures (une approche statistique) permet d'étudier les relations entre les pixels et leurs voisins. La méthode la plus citée est la méthode de la matrice de cooccurrence [5]. Dans ce qui suit, nous allons détailler cette méthode.

La méthode de matrice de cooccurrence

Une matrice de cooccurrence est une matrice de taille $N \times N$, où N est le nombre de niveaux de gris d'une image.

La méthode de matrice de cooccurrence est énormément utilisée dans le domaine du traitement d'images. Elle a été suggérée par Haralick et al. [4]. Elle permet de déterminer la fréquence d'apparitions d'un motif formé par deux pixels séparés par une distance déterminée d (d est la distance entre les 2 pixels) dans une direction μ (μ est l'angle de la droite reliant ces 2 pixels par rapport à l'horizontale). Plus exactement, dans une matrice de cooccurrence $P(a, b)_{d, \mu}$ représente le nombre de points de niveau de gris a ayant comme voisin un autre point de niveau de gris b . [4]

La matrice de cooccurrence contient une masse très importante d'informations est donc difficilement manipulable. De ce fait, quatorze indices qui correspondent à des caractères descriptifs des textures peuvent être calculés à partir de cette matrice. Nous ne présentons ici que quatre de ces indices [5]. Il est possible de calculer l'homogénéité locale, la variance, la moyenne et l'entropie par la formule suivante:

- $La\ moyenne = \frac{1}{N} \sum_i \sum_j p(i, j)$
- $La\ variance = \frac{1}{N} \sum_i \sum_j (p(i, j) - Moy)^2$
- $L'homogénéité\ locale = \frac{1}{N} \sum_i \sum_j \frac{1}{1+(i-j)^2} p(i, j)$
- $L'entropie = \frac{1}{N} \sum_i \sum_j p(i, j) \ln p(i, j)$

2.2.3. La forme

Au même titre que les caractéristiques de texture, les attributs de forme sont souvent complémentaires à la description de couleur. Les attributs de forme sont utilisés pour caractériser les objets dans les images. Il existe 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. Ces attributs sont robustes aux transformations géométriques comme la translation, la rotation et le changement d'échelle. Les seconds portent sur la caractérisation des contours de la forme. [6]

3. Méthodes d'annotation

Une source d'information, que ce soit sa nature, doit être bien décrite, sinon elle peut demeurer pratiquement inexploitable et impossible de la retrouver. L'annotation est une solution à ce problème.

L'annotation est un moyen d'exprimer la sémantique d'une image. Elle permet d'associer à l'image un mot ou un ensemble de mots, qui dénote son contenu sémantique.

Il existe trois types d'annotation d'images : manuelle, semi-automatique et automatique. La première, comme sa dénomination l'indique, est effectuée manuellement par un humain chargé d'attribuer à chaque image un ensemble de mots clé. L'annotation automatique, est une tâche effectuée par une machine et vise à alléger la charge de l'utilisateur. Le premier type d'annotation augmente la précision et baisse la productivité

tandis que le second type baisse la précision de la tâche et augmente sa productivité. Afin de faire un compromis entre ces deux tâches, leur combinaison est devenue nécessaire. C'est ce qui est connu sous le nom "l'annotation semi-automatique". [7]

3.1. Annotation manuelle

L'annotation manuelle d'images sur support électronique date des années 1970. Elle est réalisée par des personnes qui décrivent les images en les annotant avec des termes selon son point de vue.

Ce travail est notamment réalisé par des documentalistes, en général sur des collections spécifiques et de faibles tailles, telles que des photos de paysages pour les agences de presse, de voyage ou des musées. [8]

3.2. Annotation semi-automatique

L'annotation semi-automatique consiste à intégrer l'assistance humaine dans le processus d'annotation automatique, comme son nom l'indique, se devise en une première phase qui s'effectue manuellement et une deuxième qui s'effectue automatiquement. La phase manuelle, elle consiste à faire une annotation manuelle d'un échantillon d'informations seulement. Puis nous procédons dans la deuxième phase à une propagation de l'annotation.

L'annotation semi-automatique s'appuie généralement sur un moteur d'extraction d'informations qui est intégré dans un outil d'annotation. Il suggère des annotations à l'utilisateur qui peut ensuite les valider manuellement. [10]

3.3. Annotation automatique

L'annotation automatique d'images a été introduite au début des années 2000 dans le but de générer de façon automatique des mots-clés pour les images (c.-à-d. sans faire intervenir les utilisateurs). Elle a comme objectif de combler les lacunes de l'annotation manuelle et semi-automatique respectivement. [11]

Dans le tableau suivant nous représentons les avantages et les inconvénients de chaque type d'annotation :

Type	Avantage	Inconvénient
Annotation manuelle	-Bonne précision	-La difficulté de mise en œuvre, laborieuse, fastidieuse à réaliser pour de grands volumes de photos. -Sujette à l'interprétation des annotateurs (deux annotateurs différents peuvent décrire une même image avec des termes différents). -Coûteuse
Annotation semi-automatique	-Plus performante que l'annotation manuelle en termes d'efficacité (rapidité). -Bonne précision	-Elle est sujette à l'interprétation des individus. -Le coût est plus élevé que celui de l'annotation automatique.
Annotation automatique	-Plus rapide -Moins coûteuse	-Faible précision

Tab -1.1 : Avantages et inconvénient des différents types d'annotation.

4. Les approches d'annotation automatique des images

Dans la littérature, il existe deux approches d'annotation automatique : l'annotation automatique d'images basée sur le contenu et l'annotation automatique d'images à partir du texte associé à l'image (basée sur leur contexte) [Wu et al., 2005]. La première fait référence aux caractéristiques locales et globales des images (par exemple, texture, couleur, forme. . .), tandis que la deuxième fait référence aux caractéristiques internes (par exemple, localisation, date et heure. . .) et externes des images (par exemple, texte qui entoure l'image, profil des utilisateurs. . .).

4.1. L'annotation automatique à partir du texte associé à l'image

Dans cet approche, l'annotation n'est pas possible sauf si les images sont associées à du texte. C'est le cas des images du web, où l'annotation automatique peut s'effectuer à partir [8] :

- ✓ Des mots présents dans le titre de la page.
- ✓ Des mots pertinents de cette page.
- ✓ Des URL avec le nom d'image. Cependant, les images sont généralement nommées d'une manière générique (image01.jpg) qui ne comporte aucun sens.
- ✓ Des mots associés à attribut ALT de la balise IMG d'une image.
- ✓ Du texte proche de l'image.
- ✓ Ou de la fusion de toutes ces informations.

4.2. L'annotation automatique à partir du contenu visuel de l'image

Cette approche, souvent appelée l'auto-annotation par le contenu, permet de combiner et de trouver des liens entre les informations textuelles et visuelles,

Afin d'exploiter les informations contenues dans une image, une extraction des descripteurs visuels (couleur, texture et forme) est nécessaire. Ces descripteurs visuels [12], peuvent être :

- **Globale** : Les caractéristiques des images sont calculées sur la globalité de l'image (c.-à-d. l'image tout entière). [13]
- **Locale** : Les images sont divisées en blocs de taille fixe ou en régions, appelés aussi sous-unités d'images. Les caractéristiques visuelles sont ainsi calculées pour chaque sous-unité séparément. [13]
- **Hybride** : Des chercheurs ont montré que la combinaison des méthodes globales et locales peut améliorer l'annotation automatique d'images pour certaines tâches, telles que la reconnaissance des visages et la détection d'objets dans les images. Ces méthodes figurent dans la littérature sous le nom de « méthodes hybrides ». [13]

5. Processus d’auto-annotation des images

Nous allons présenter le processus d’auto-annotation des images du système proposé par [14] qui rentre dans le cadre de notre projet de recherche. Le processus d’auto-annotation se déroule en deux phases principales .Apprentissage et traitement d’une nouvelle image.

La première phase est basée sur un corpus d’images annotées. Où les images de corpus doivent être segmentées en régions afin de les grouper dans des classes de régions visuellement similaires appelées blobs.

Afin de corrélérer le textuel (annotation de corpus) avec le visuel (les blobs des régions), il faut faire une annotation des blobs précédemment construits.

La figure-1.2 schématise les phases du système d’annotation automatique [14].

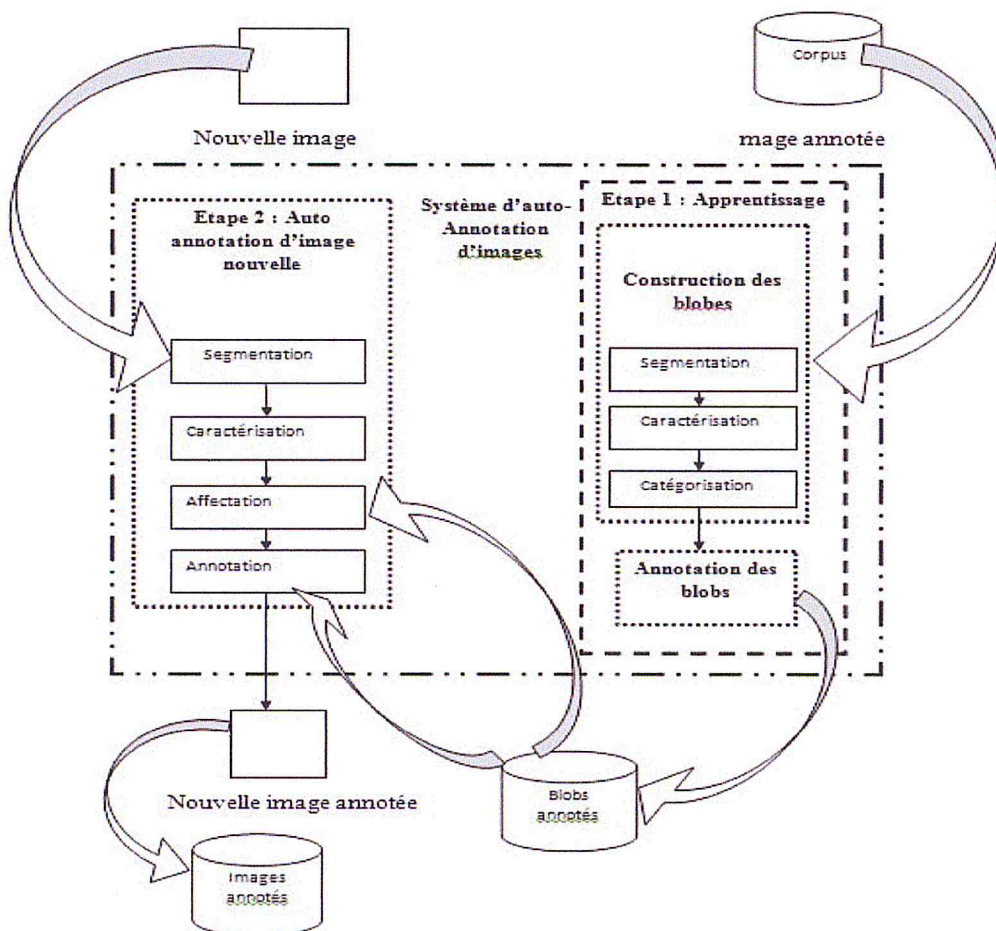


Figure 1.1:Présentation des phases du système d’auto-annotation [14].

5.1. Apprentissage :

L'apprentissage a pour but la construction et l'auto-annotation des blobs. Nous segmentons les images d'un corpus d'images bien annotées. Les régions visuellement similaires sont regroupées dans des blobs, qui seront annotés.

L'étape d'apprentissage à partir d'un corpus d'images annotées comprend deux phases : construction des blobs et l'auto-annotation des blobs [14].

5.1.1. Construction des blobs

La première phase d'apprentissage regroupe :

- Chargement et prétraitement des images du corpus.
- la segmentation d'images du corpus en régions.
- l'extraction des caractéristiques visuelles des régions.
- la catégorisation de ces dernières dans des blobs [14].

5.1.2. Le filtrage des images

Le but de filtrage est de distinguer entre les pixels blancs dans une région de celles de l'arrière-plan. Pour cela, nous allons diminuer la couleur d'un pixel dans l'image par un degré, c'est-à-dire sa valeur RGB (255, 255, 255) devienne RGB (254, 254, 254). Par conséquent, lors de parcourt de la matrice, tout pixel dont la valeur est (255, 255, 255) sera ignoré.

5.1.3. La segmentation de la base d'images

La segmentation est un traitement de bas niveau qui consiste à créer une partition de l'image A en sous-ensembles R_i , appelés régions tels qu'aucune région ne soit vide. L'intersection entre deux région soit vide et l'ensemble des régions recouvre toute l'image. [8]

La segmentation joue un rôle important dans l'annotation d'image basée sur les régions car elle sépare les objets dans les images; mais cela reste théorique. Cependant, elle ne fournit pas une correspondance exacte entre un concept et une région. En général, une bonne segmentation conduit à une bonne annotation [15].

La segmentation s'applique pour chaque image de la base de corpus d'image. Elle consiste à extraire les régions de l'image.

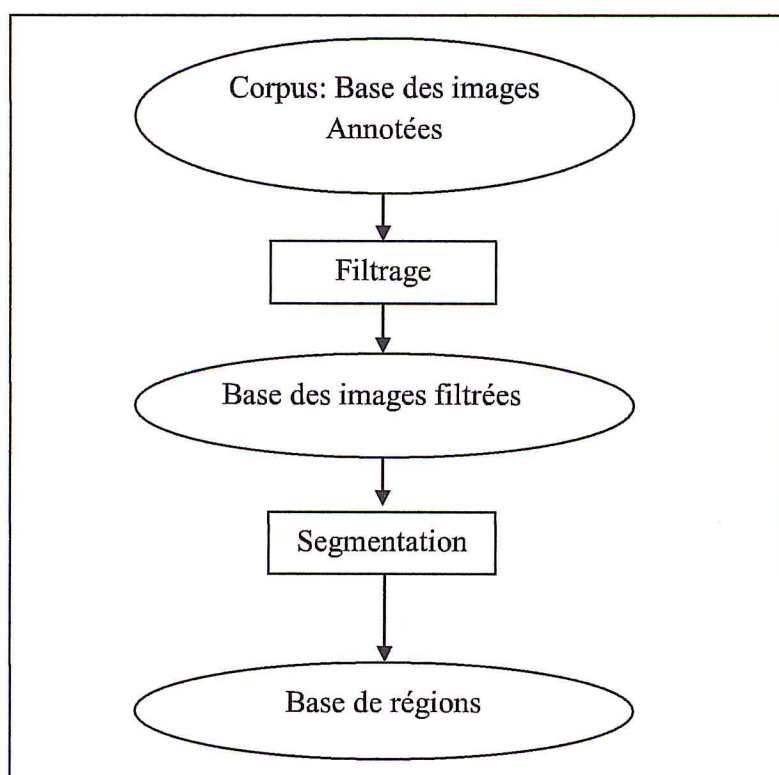


Figure 1.2 : Architecture de module de segmentation

5.1.4. Caractérisation

Dans cette étape, nous extrayons les caractéristiques visuelles de chacune des régions qui constituent l'image afin de les catégoriser dans des blobs. Les caractéristiques choisies sont : les moments statistiques de couleur (la moyenne, la variance et les moments d'ordre trois) de chaque composante d'espace RGB et la texture (la variance, l'homogénéité et l'entropie).... [15]

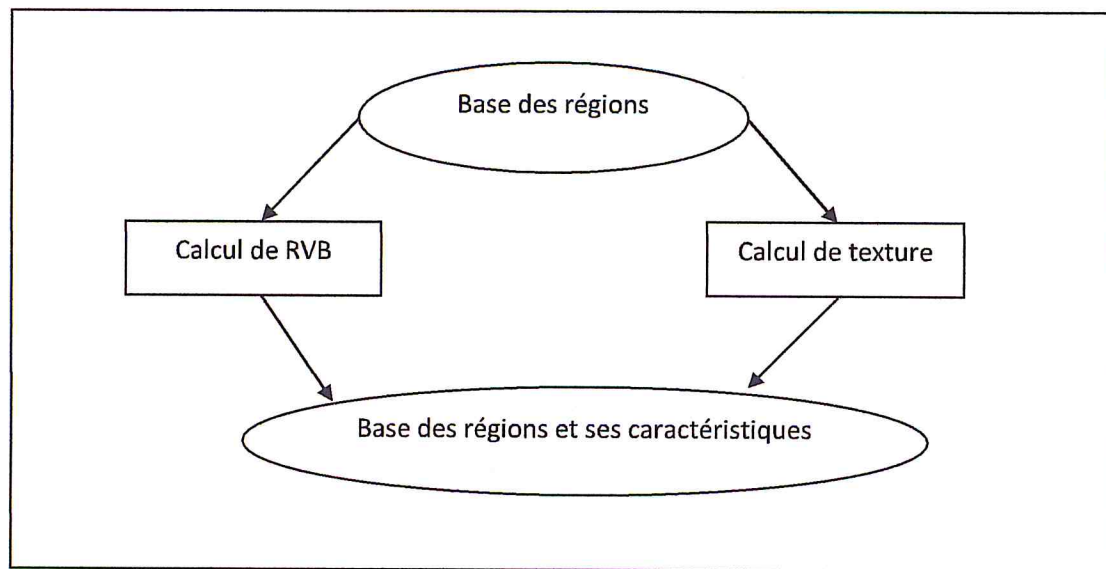


Figure 1.3 : Les modules des calculs des caractéristiques des régions

5.1.5. Catégorisation

La catégorisation est une étape primordiale dans l'apprentissage du processus d'auto-annotation d'image. Elle est a pour but d'améliorer la qualité d'annotation d'images [15].

Une fois l'ensemble des caractéristiques choisi, il faut donc choisir une méthode de classification. Cette méthode nous permet de catégoriser les régions selon leurs caractéristiques visuelles en blobs.

La figure suivante représente les différentes étapes de la phase d'apprentissage

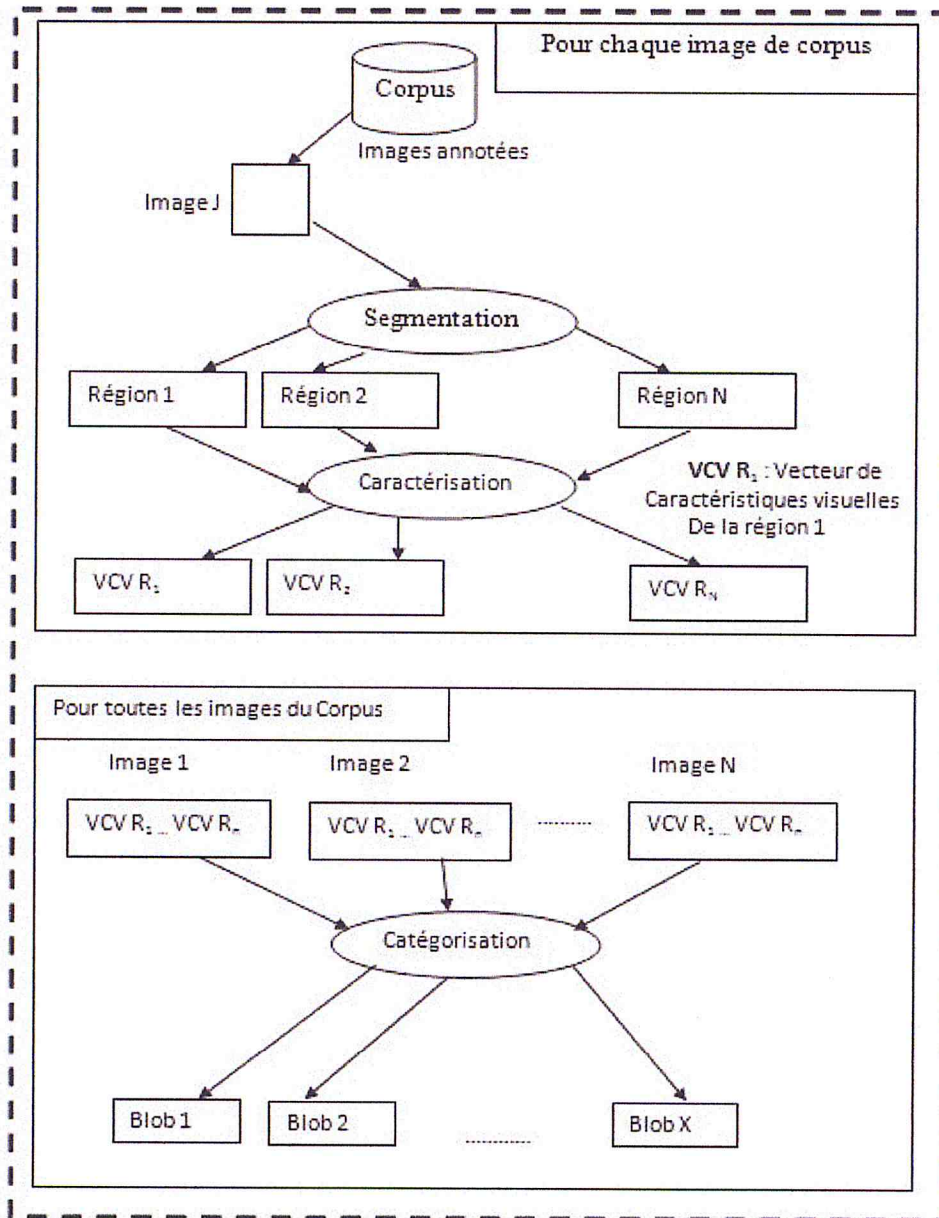


Figure 1.4 : Schéma d'apprentissage : Construction de blobs [14].

5.1.6. Annotation automatique des blobs

La deuxième étape de l'apprentissage permet de corrélérer le contenu visuel avec le contenu sémantique en annotant les blobs regroupant les régions visuellement similaires, par les mots clé d'images du corpus correspondant. Ceci permettra d'annoter les nouvelles images.

5.2. Traitement d'une nouvelle image

Une nouvelle image peut alors être annotée automatiquement comme suit [14] :

1. Chargement et prétraitement d'une nouvelle image.
2. Segmenter la nouvelle image en régions.
3. L'extraction des caractéristiques visuelles des régions de la nouvelle image.
4. Affecter les régions de l'image aux blobs construits précédemment.
5. Annoter l'image par les mots des blobs auxquels les régions de cette image appartiennent.

6. Conclusion :

L'annotation automatique d'images est un moyen d'exprimer la sémantique d'une image. Cette technique permet d'associer à l'image de façon automatique des mots clés, qui dénote son contenu sémantique. Elle a comme objectif de combler les lacunes de l'annotation manuelle et semi-automatique respectivement et d'améliorer la recherche sur le Web.

Dans ce chapitre nous avons donné l'essentiel sur l'image numérique, leurs caractéristiques de bas niveau (Couleur, forme, texture, ...) et l'annotation d'image. Ainsi qu'une description sur le processus d'auto-annotation des images qui rentre dans le cadre de notre projet de recherche.

Le prochain chapitre concerne la classification et ces différentes méthodes.

Chapitre : 11

Techniques de classification

1. Introduction

La classification est une discipline reliée de près ou de loin à plusieurs domaines, elle est connue aussi sous noms variés (catégorisation, clustering...) selon les objets qu'elle traite et les objectifs qu'elle vise à atteindre.

La classification est une étape importante dans le processus d'auto-annotation d'image. Elle vise à développer un partitionnement optimal, c'est à dire le regroupement des données en classes qui partagent des caractéristiques similaires où les données sont généralement représentées par des vecteurs de mesures. En d'autres termes, l'objectif des méthodes de classification est d'identifier les groupes d'un ensemble non étiqueté de vecteurs de données qui partagent des similarités sémantiques. Cela permet à l'utilisateur de construire un modèle cognitif, favorisant ainsi la détection de la structure inhérente d'un ensemble de données.

Nous présentons, dans la suite de ce chapitre les différentes approches et méthodes de classification.

2. Classification

La classification permet de grouper (classifier) les individus en exploitant leur similarité. Les individus peuvent être : les pixels, une sous image de taille fixe ou une région [16].

D'une manière plus formelle, soit un ensemble X de N objets décrits chacun par leurs P attributs. La classification consiste à créer une partition ou une décomposition de cet ensemble en groupes telle que :

- Les objets appartenant au même groupe se ressemblent.
- Les objets appartenant à deux groupes différents soient peu ressemblants.

Cette vision contraint donc à disposer d'une distance définie sur le langage de description des objets (dans notre cas l'espace de description des images qui sera un espace vectoriel numérique dans lequel chaque dimension correspond à un attribut distinct). La figure suivante présente différents regroupement selon différentes mesure distances

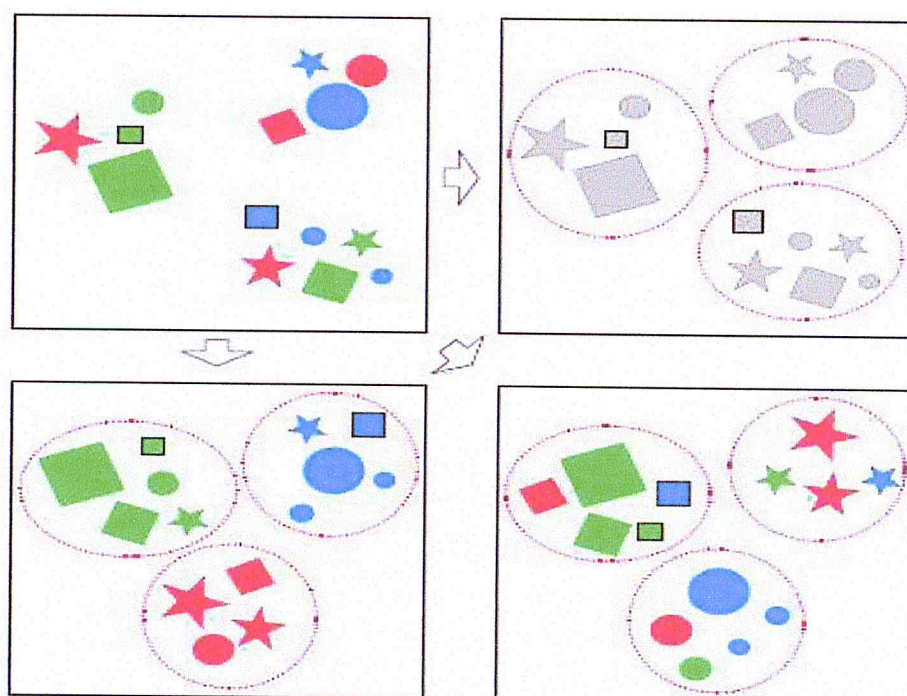


Figure 2.1 : Différents regroupements selon différentes mesures de distance

La classification est résumée en trois étapes :

1. Choix :

- Des données (ensemble des individus $E = (e_1, e_2, \dots, e_n)$).
- De la mesure d'éloignement (dissimilarité, distance) entre individus (Le choix d'une mesure dépend de la nature des descripteurs. Le plus souvent, le choix se porte sur une distance euclidienne).
- Du critère d'homogénéité des classes à optimiser (généralement inertie).

2. Choix d'un algorithme de classification et exécution.

3. L'interprétation des résultats :

- Évaluation de la qualité de la classification.
- Description des classes obtenues [16].

3. Les critères de classification :

L'homogénéité et la séparation sont deux critères proposés par [25].

L'homogénéité est calculée en tant que la distance moyenne entre chaque objet et le centre du cluster dont il appartient. La formule de l'homogénéité est comme suit :

$$H_{ave} = \frac{1}{N} \sum_i d(i, C(i))$$

Où i est un objet et $C(i)$ est le centre du cluster qui contient l'objet i , N est le nombre total d'objets; d est la fonction de distance.

La séparation est calculée comme la distance moyenne des poids entre les centres des groupes [26]. La formule de la séparation est comme suit :

$$S_{ave} = \frac{1}{\sum_{i \neq j} N_{ci} N_{cj}} \sum_{i \neq j} N_{ci} N_{cj} D(C_i, C_j)$$

Où C_i et C_j sont les centres du i^{eme} et j^{eme} cluster, et N_{ci} et N_{cj} sont le nombre d'objets dans le i^{eme} et j^{eme} cluster. Ainsi H_{ave} reflète la compacité des clusters tandis que S_{ave} reflète la distance globale entre les clusters. Décroître H_{ave} où accroître S_{ave} permet d'améliorer les résultats de regroupement. Il est à noter également que H et S ne sont pas indépendants l'un de l'autre, H est étroitement lié à la variance dans le cluster, S est étroitement lié à la variance entre les clusters. Pour un ensemble de données quelconque, la somme de la variance intra-clusters et la variance inter-clusters est une constante.

4. Les méthodes de classification

Selon le nombre d'attributs considérés, les méthodes de classification peuvent être monodimensionnelles ou multidimensionnelles [16].

4.1. Les méthodes monodimensionnelles

Elles consistent à classer les individus en fonction d'un seul attribut (niveau de gris). Ces méthodes déterminent des seuils qui constitueront les limites des différentes classes. Il existe trois catégories de méthodes de seuillage : seuillage globale, seuillage locale et seuillage dynamique [16].

4.2. Les méthodes multidimensionnelles

Elles exploitent plusieurs attributs afin de classifier les individus. Nous présentons deux types de méthodes de classification multidimensionnelles : la classification supervisée et la classification non supervisée (clustering) [16].

4.2.1. Classification supervisée

La classification supervisée (appelé classement ou inductive) a pour objectif « d'apprendre » par exemple. Elle cherche à expliquer et à prédire l'appartenance des individus à des classes connues a priori [17].

A partir d'une classification connue a priori d'un certain nombre d'individus, la classification supervisée permet de construire une fonction d'identification ou de discrimination pour les autres individus afin de découper l'espace de représentation en zones. Chaque zone est affectée à une classe de la classification a priori. Les autres individus sont ensuite classifiés en fonction de leur position dans l'espace de représentation [18].

Parmi les méthodes de classification supervisée, nous citons les K plus proche voisins (K-ppv), les séparateurs à vaste marges...etc.

4.2.1.1. Les K plus proche voisins (K-PPV)

La méthode de k plus proche voisins noté parfois *k*-NN (en EN : *Nearest Neighbor*) ou *k*-PPV consiste à déterminer pour chaque nouvel individu que l'on veut classer, la liste des *k* plus proches voisins parmi les individus déjà classés. L'individu est affecté à la classe qui contient le plus d'individus parmi ces *k* plus proches voisins. Cette méthode nécessite de choisir une distance (la plus classique est la distance euclidienne). Et le nombre *k* de voisins à prendre en compte [19].

La méthode utilise donc quatre paramètres : le nombre de classe, valeur de K, exemples initiaux et la fonction de similarité pour comparer le nouveau cas aux cas déjà classés. La formule suivante présente la distance euclidienne entre x_i et x_j .

$$d(x_i, x_j) = \sqrt{\sum_{i=0}^n (a_r(x_i) - a_r(x_j))^2}$$

Algorithme de K-PPV

L'algorithme de *k-ppv* est résumé en deux étapes essentielles :

1. initialisation, choix de :

- Nombre de classes.
- Valeur de *k*.
- exemples initiaux.
- mesure de similarité.

2. pour chaque vecteur d'objet à classer :

- mesurer la distance du vecteur avec tous les autres déjà classés.
- déterminer la liste des *k* vecteurs les plus proches de lui (*k-ppv*).
- déterminer la classe la plus représentée dans la liste des *k-ppv* et affecter notre vecteur à cette classe [19].

Algorithme 2.1 : Algorithme de classification *k-ppv*

Avantage :

Cette méthode supervisée est souvent performante. De plus, son apprentissage est assez simple.

Inconvénient :

Le temps de prédiction est très long, car il nécessite le calcul de la distance avec tous les exemples.

4.2.1. 2. Les Séparateurs à Vastes Marges (SVM)

Les Support Vector Machines souvent traduit par l'appellation de Séparateur à Vaste Marge, SVM est une méthode de classification binaire par apprentissage

supervisé. Elle peut être utilisée dans le cas où l'espace de représentation des individus est linéairement séparable. La méthode à base de SVM cherche l'hyperplan séparateur le plus éloigné des individus du corpus d'apprentissage la figure suivant représente le choix d'un hyperplan séparateur.

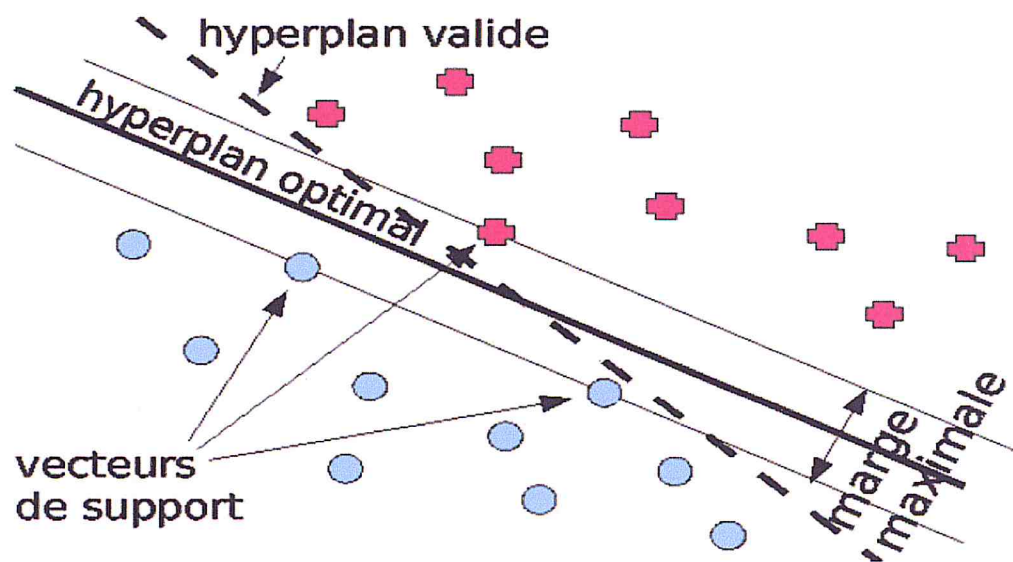


Figure 2.2 : Hyperplan séparateur

Notions de base : Hyperplan, marge et support vecteur

Pour deux classes d'exemples donnés, le but de SVM est de trouver un classificateur qui va séparer les données et maximiser la distance entre ces deux classes. Ce modèle ne suppose pas que les descripteurs sont indépendants contrairement aux méthodes bayésiennes. Avec SVM, ce classificateur est un classificateur linéaire appelé hyperplan. Dans la figure 2.3, nous déterminons un hyperplan qui sépare les deux ensembles de points. [23]

Les points les plus proches, seuls sont utilisés pour la détermination de l'hyperplan, sont appelés vecteurs de support.

Il est évident qu'il existe une multitude d'hyperplan valide mais la propriété remarquable des SVM est que cet hyperplan doit être optimal. Nous allons donc en plus chercher parmi les hyperplans valides, celui qui passe « au milieu » des points des deux classes d'exemples. Intuitivement, cela revient à chercher l'hyperplan optimal «le plus

sûr ». En effet, cela revient à chercher un hyperplan dont la distance minimale aux exemples d'apprentissage est maximale. Cette distance est appelée « marge » entre l'hyperplan et les exemples. L'hyperplan séparateur optimal est celui qui maximise la marge. Comme nous allons chercher à maximiser cette marge, nous parlerons de *séparateurs à vaste marge*.

Maximisation de marge

Intuitivement, le fait d'avoir une marge plus large procure plus de sécurité lorsque l'on classe un nouvel exemple. De plus, si l'on trouve le classificateur qui se comporte le mieux vis-à-vis des données d'apprentissage, il est clair qu'il sera aussi celui qui permettra au mieux de classer les nouveaux exemples. Dans le schéma qui suit, la partie droite nous montre qu'avec un hyperplan optimal, un nouvel exemple reste bien classé alors qu'il tombe dans la marge. Nous constatons sur la partie gauche qu'avec une plus petite marge, l'exemple se voit mal classé.

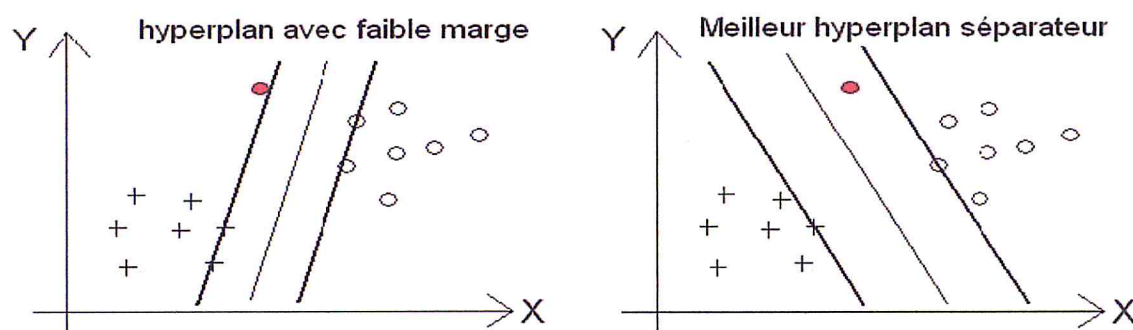


Figure 2.3 : La relation entre l'hyperplan optimal et l'hyperplan valide.

En général, la classification d'un nouvel exemple inconnu est donnée par sa position par rapport à l'hyperplan optimal.

Linéarité et non-linéarité

Parmi les modèles des SVM, nous constatons les cas linéairement séparable et les cas non linéairement séparable. Les premiers sont les plus simples de SVM car ils permettent de trouver facilement le classificateur linéaire. Dans la plupart des problèmes réels il n'y a pas de séparation linéaire possible entre les données, le

classificateur de marge maximale ne peut pas être utilisé car il fonctionne seulement si les classes de données d'apprentissage sont linéairement séparables.

Avantage :

- Les SVM possèdent des fondements mathématiques solides.
- Les exemples de test sont comparés juste avec les supports vecteur et non pas avec tous les exemples d'apprentissage.
- Décision rapide. La classification d'un nouvel exemple consiste à voir le signe de la fonction de décision $f(x)$.

Inconvénients :

- SVM est une classification binaire, qui nécessite d'utilisation de l'approche un-contre-un.
- Grande quantité d'exemples en entrées implique un calcul matriciel important.
- Temps de calcul élevé lors d'une régularisation des paramètres de la fonction noyau.

4.2.2. Classification non supervisée

Elle consiste à découper l'espace de représentation en zones homogènes et distinctes selon un critère de ressemblance entre les individus, c'est-à-dire selon un critère de proximité dans l'espace des attributs [16]. Ces zones sont des classes (cluster).

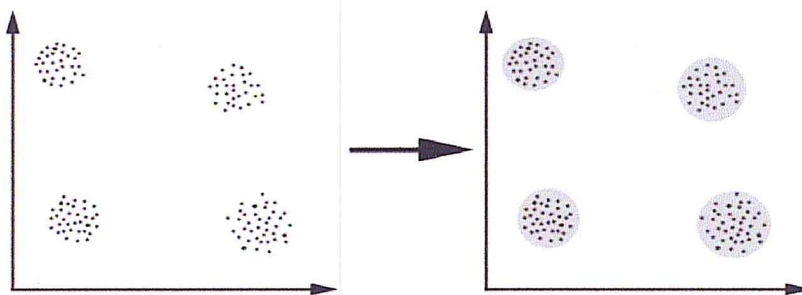


Figure.2.4 : Illustration de regroupement en clusters.

Un «Cluster» est donc une collection d'objets qui sont «similaires» entre eux et qui sont dissemblables par rapport aux objets appartenant à d'autres groupes. Cette définition clairement graphiquement dans l'exemple de La figure 2.5, elle représente L'illustration de regroupement en clusters.

Contrairement à la classification supervisées, elle ne possède pas des connaissances a priori sur les classes prédéfinies des éléments. Donc la division des objets dans les différents groupes (clusters) se procède en se basant sur le calcul de similarité entre les éléments.

Alors que l'objectif des méthodes du Clustering est de grouper des éléments proches dans un même groupe de manière à ce que deux données d'un même groupe soient le plus similaires possible et que deux éléments de deux groupes différents soient le plus dissemblables possible.

Les principales exigences qu'un algorithme de clustering doit répondre sont les suivantes :

- Évolutivité des clusters.
- Traiter les différents types d'attributs.
- Découvrir les clusters de forme arbitraire
- Exigences minimales pour la connaissance du domaine afin de déterminer les paramètres d'entrée.
- Capacité de composer avec le bruit et les valeurs manquantes traitées les dimensionnalités élevées.
- L'intelligibilité et la convivialité.

4.2.2. 1. La classification par K-moyenne (K-mean)

La méthode des K-moyennes (en anglais : K-means) est un outil de partitionnement des données non-hiérarchique qui permet de répartir les données en K clusters homogènes.

K-means est l'un des algorithmes de classification non supervisées les plus simples et les plus utilisés. La procédure de classification d'un ensemble de données en un certain nombre de clusters (k clusters) fixé a priori est aussi simple et facile (Le nombre de clusters K est déterminé par l'utilisateur à l'initialisation).

L'idée principale est de définir k centres initiaux, un pour chaque cluster. Ces centres devraient être placés d'une manière rusée, car différents emplacements causes des résultats différents. Donc, le meilleur choix est de les placer loin les uns des autres autant que possible. La prochaine étape est de prendre chaque point appartenant à l'ensemble de données et de l'associer au centre le plus proche. Après avoir affecté tous les points aux clusters représentés par les centres initiaux, la première étape est terminée.

L'étape suivante consiste à recalculer k nouveaux centres comme barycentre des groupes issus de l'étape précédente. Après avoir calculé ces k nouveaux centres, une nouvelle phase d'affectation de chaque objet de données au nouveau centre le plus proche. A chaque itération, les k centres changent leurs localisations étape par étape. Ou en d'autres termes, les centres se déplacent à chaque itération. Ce processus se poursuit jusqu'à ce qu'il n'y ait aucun mouvement des centres. Par conséquence, les k clusters qui représentent l'ensemble de données sont alors identifiés, la Figure 2.7 illustre le déroulement étape par étape d'un tel algorithme [28].

L'algorithme des K-moyennes peut être résumé dans les 4 étapes suivant :

1. choisir aléatoirement k objets de la base de données comme centroïdes initiaux représentant les k clusters recherchés.
2. assigner chaque objet au cluster dont le centroïde est le plus proche;
3. puis tant qu'au moins un objet change de cluster d'une itération à l'autre:
 - mettre à jours les centroïdes des clusters.
4. réitérer les étapes (2) et(3) jusqu'à ce que l'algorithme converge vers un minimum local, optimisation itérative.

Algorithme 2.2 : Algorithme de classification K-moyennes.

Le déroulement de l'algorithme K-means peu être représenté dans la figure suivant :

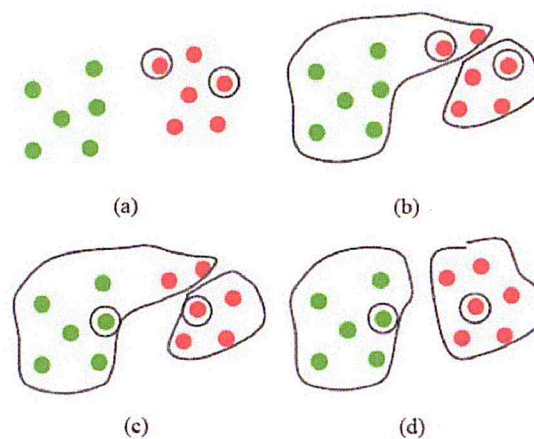


Figure 2.5: déroulement de K- means.

4.2.2.2. Algorithme de classification Fuzzy C-Means (FCM)

Cet algorithme est une extension de l'algorithme des K-means en introduisant une notion de floue dans la définition du degré d'appartenance. Son principe est de regrouper des individus dans K classes qui soient le plus homogènes et naturelles

possibles. Les groupes obtenus doivent contenir des individus les plus semblables, et entre groupes différents les individus doivent être le plus différents possible. En fait, la modélisation de l'imprécision s'effectue en considérant des frontières graduelles au lieu de frontières nettes entre les classes. L'incertitude s'exprime par le fait qu'un individu possède bien des attributs qui l'assignent à une classe qu'à une autre, la classification floue assigne donc son degré d'appartenance à chacune des classes. Le degré d'appartenance se situe dans l'intervalle $[0, 1]$ et les classes obtenues ne sont pas forcément disjointes. Dans ce cas l'individu x_j n'est plus assigné à une classe unique mais à plusieurs classes, par l'intermédiaire de degré d'appartenance u_{ij} du vecteur x_j à la classe V . Le but des algorithmes de classification est non seulement de calculer les centres de classe B mais aussi l'ensemble des degrés d'appartenances des vecteurs aux classes. L'algorithme de classification fuzzy C-Means est comme suit :

Données : K classes, centre v des classes, partition initiale, m degré de flou

Résultat : Partition finale

- Fixer arbitrairement un nombre de classes K .
- Initialiser la matrice de degré d'appartenance $U_{K \times N}[u_{ij}]$. Cette matrice représente la partition floue des données et doit vérifier la condition de normalisation

$$\sum_{i=1}^K u_{ij} = 1, \forall j = 1, \dots, N;$$

Repeter

Calculer les centres de classes

$$v_i = \left(\sum_{k=1}^N u_{ik}^m x_k \right) / \left(\sum_{k=1}^N u_{ik}^m \right) \quad i = 1, 2, \dots, K$$

Mettre à jour la matrice d'appartenance

$$\mu_{ij} = \frac{1}{\sum_{j=1}^K \left(\frac{1}{d_j^2(x_i, v_j)} \right)^{(1/m-1)}} \quad i = 1, 2, \dots, K \text{ et } j = 1, 2, \dots, N$$

Calcul de critère de convergence ;

Jusqu'à Convergence ;

Algorithme 2.3 : Algorithme FCM [36].

La condition d'arrêt n'est pas la stabilisation des centres mais la stabilisation de la matrice d'appartenance.

Avantage :

L'avantage de ces algorithmes est avant tout leur grande simplicité mais aussi leur complexité algorithmique qui reste raisonnable.

Inconvénients :

Les deux algorithmes K-means et FCM souffrent aussi de tous les inconvénients habituels :

- La partition finale obtenue est très dépendante du choix des centres initiaux.
- Le processus est sensible aux valeurs aberrantes.
- Plus le nombre de données est grand plus l'utilisateur aura des problèmes lors de la détermination du nombre de clusters.

4.2.2.4. Les cartes auto-organisatrices (SOM)

Les cartes auto-organisatrices (*self organizing map* SOM) ou carte de *Kohonen* (1982)) [34] appartiennent aux techniques de clustering par partitionnement des données. Cette technique est un cas particulier de réseau de neurones non-supervisé. Elle considère deux espaces indépendants :

- L'espace de données généralement de grande dimension.
- L'espace de représentation (la carte) de dimension réduite.

La carte est constituée d'un ensemble de neurones organisés selon une certaine structure topologique (carré, rectangle, cylindre, carte 3D...) fixée a priori. La topologie de la carte introduit une notion de voisinage (rectangulaire, hexagonal ...) et de distance entre les neurones. Chaque neurone J de la carte possède des coordonnées fixes sur la carte, et des coordonnées adaptables dans l'espace d'entrée. Les coordonnées adaptables sont les poids W_{ij} qui le relie à chacune de ses entrées x_i .

Ces réseaux sont composés d'une Grille de neurones (ou nœuds, ou unités) auxquels seront présentés des stimuli. Un stimulus est un vecteur, de dimension d , qui

décrit un objet à classer. Ce vecteur peut aussi bien être une description des caractéristiques physiques des objets stimuli qui fait référence à des caractéristiques telle que la présence ou l'absence d'un mot-clé dans un document. Chaque unité de la grille est reliée au vecteur d'entrée (stimulus) par l'intermédiaire de d synapses de poids. En fait, à chaque unité est associé un vecteur de dimension d qui contient les poids w .

Algorithme de SOM :

Au début, la couche de sortie doit être initialisée de manière aléatoire. Un cycle d'apprentissage est constitué des étapes suivantes :

1. Présenter un vecteur d'entrée choisi au hasard associé à un stimulus à la grille (images).
2. Trouver le nœud gagnant (ou *winner*) : C'est l'unité dont le vecteur associé est le plus similaire, en terme de distance Euclidienne au vecteur d'entrée, en utilisant la formule suivante :

$$\| \text{entrée} - \text{neurone winner} \| = \text{Min}_i \| \text{entrée} - \text{neurone}_i \|$$

3. Modifier les poids W_i du nœud gagnant, ainsi que ceux de son entourage, de manière à ce que les vecteurs associés (les vecteurs de poids) «se rapprochent d'avantage» du vecteur d'entrée présenté à la grille. La règle de modification est la suivante :

$$\begin{cases} w_i(t+1) = w_i(t) + h(r, t)(p_i - w_i(t)) & \text{si } i \in \text{voisinage} \\ w_i(t+1) = w_i(t) & \text{si } i \notin \text{voisinage} \end{cases}$$

$$\text{Avec : } h(r, t) = \alpha(t) \cdot v(t)$$

($\alpha(t)$: Le taux d'apprentissage. $v(t)$: la fonction de voisinage).

4. Faire décroître la taille de la zone de voisinage des nœuds gagnants (la zone qui contient les neurones subissant la transformation).
5. Faire décroître le coefficient d'apprentissage, $\alpha(t)$, qui contrôle l'importance des modifications appliquées aux vecteurs de poids.
6. Arrêter l'apprentissage si le coefficient d'apprentissage est nul, sinon, présenter un autre stimulus à la grille

Algorithme 2.4 : Algorithme SOM.

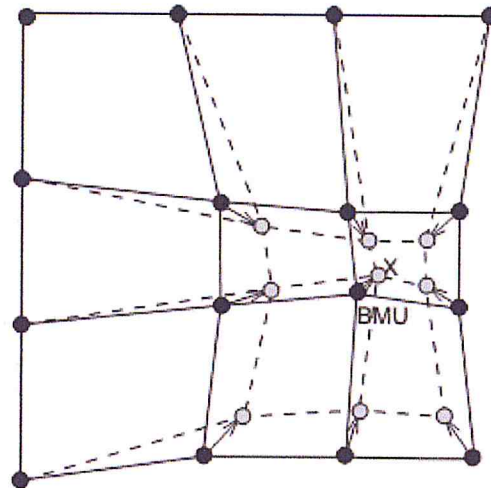


Figure 2.6 : Mise à jour de la (BMU) et ses voisins

La figure précédente montre la mise à jour du neurone gagnant (BMU) et ses voisins vers l'échantillon d'entrée marqué avec x. Les lignes continues et en pointillés correspondent respectivement, à la situation avant et après mise à jour.

La modification des vecteurs associés aux unités se fait de manière différente selon la distance des nœuds par rapport à l'unité gagnante. Le nœud gagnant sera celui dont le vecteur subira le plus de modifications, tandis que les unités les plus éloignées seront moins affectées. La fonction de voisinage $v(r)$ va être maximale pour $r = 0$ et décroître quand r croît (quand on s'éloigne du nœud gagnant). La fonction de voisinage la plus couramment employée est la courbe gaussienne.

La fonction de voisinage :

Cette fonction décrit comment les neurones voisins de l'unité gagnante corrigent leurs poids. Son rôle est de forcer les neurones du voisinage de l'unité gagnante à rapprocher leurs vecteurs de poids du vecteur d'entrée correspondant. Elle est présentée par la fonction gaussienne suivante :

$$v(t) = e^{-\frac{d_j^2}{2r^2}}$$

Tel que :

d_j : est la distance entre le nœud j et le nœud gagnant(BMU).

r : est le rayon du voisinage de nœud gagnant(BMU).

La fonction de voisinage dans le processus d'auto-organisation commence avec un rayon de voisinage $N_c(0)$ grand (on peut commencer par un rayon égal à la moitié de la plus grande dimension du réseau). La figure ci-dessous, explique la notion du rayon de voisinage d'un neurone.

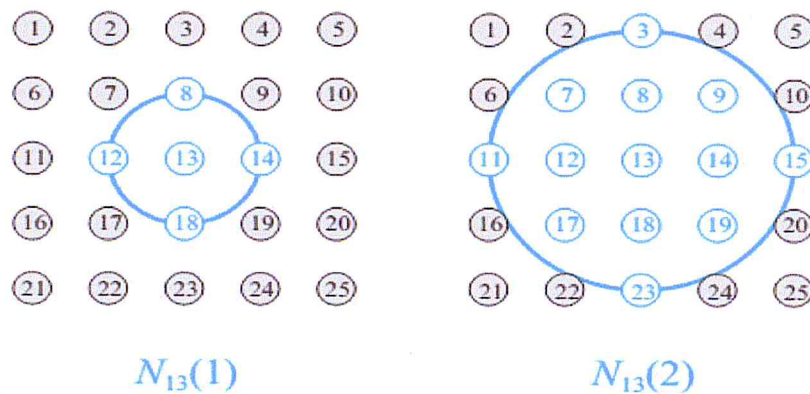


Figure 2.7 : Le rayon de voisinage

b) La forme du de la carte :

Pour obtenir des cartes stables et non ambiguës, selon [35], il est préférable de tenir comptes des points suivants :

- La forme hexagonale de la carte est préférée, car elle ne favorise pas beaucoup les directions horizontales et verticales.
- Les bords du réseau doivent être rectangulaires plutôt que carrés, parce que les vecteurs m_i qui forment la carte doivent être stabilisés dans le processus d'apprentissage.
- Dans le cas où le réseau est circulaire, il n'existe pas d'orientation préférée.

La figure suivante présente deux formes de voisinage, rectangulaire et hexagonal, que peut prendre une carte de *Kohonen* (couche de sortie).

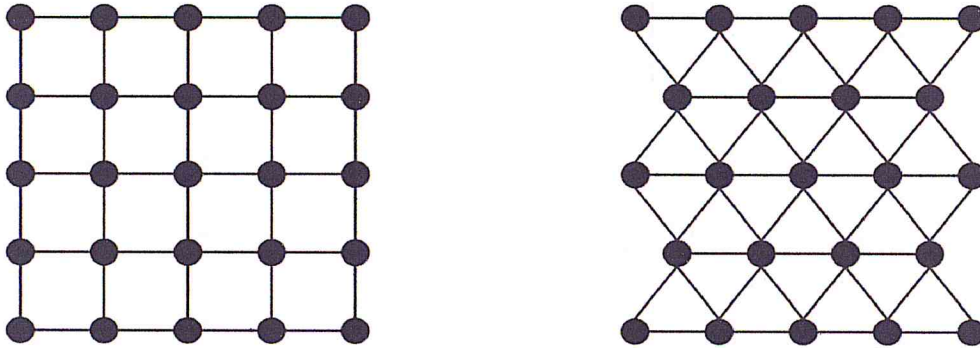


Figure 2.7 : Grille SOM rectangulaire (gauche) et grille SOM hexagonale (droite)

5.2.2.5. La classification Ascendante Hiérarchique (CAH)

La classification hiérarchique ascendante procède successivement par fusionnement de plus petits clusters dans les plus grands. Le résultat de l'algorithme est un arbre de clusters, appelés le dendrogramme, qui montre comment les clusters sont reliés.

Cette méthode est la plus ancienne et la plus utilisée dans la classification automatique. Supposons que nous avons N objets à classer. Les algorithmes agglomératifs suivant cette approche, définissent d'abord une partition initiale en N classes unitaires. Par la suite, ils fusionnent successivement les classes jusqu'à ce que toutes les entités soient dans la même classe. Dans chaque étape de fusion des classes, le recalcul des dissimilarités entre les nouvelles classes est nécessaire. Le choix des classes se fait selon le critère qui caractérise la méthode. Les méthodes de cette catégorie diffèrent selon le critère local choisi et selon la méthode de calcul des dissimilarités interclasses [35].

Le principe est de fournir un ensemble de partitions de moins en moins fines obtenues par regroupement successifs de parties. Une fois cet algorithme terminé, on ne récupère donc pas directement une partition, mais une hiérarchie de partitions en $n, \dots, 1$ classes, avec diminution de l'inertie inter-classes à chaque agrégation. Ce principe est résumé dans les étapes suivantes :

- 1) Initialisation : les classes initiales sont les n singletons-individus. Calculer la matrice de leurs distances deux à deux.
- 2) Itérer les deux étapes suivantes jusqu'à l'agrégation en une seule classe.
- 3) Regrouper les deux éléments (classes) les plus proches au sens de la distance entre groupes choisie.
- 4) Mettre à jour le tableau de distances en remplaçant les deux classes regroupées par la nouvelle et en calculant sa distance avec chacune des autres classes [35].

Algorithme 2.5 : Algorithme de classification hiérarchique ascendante

La figure présente un dendrogramme pour la classification hiérarchique ascendante

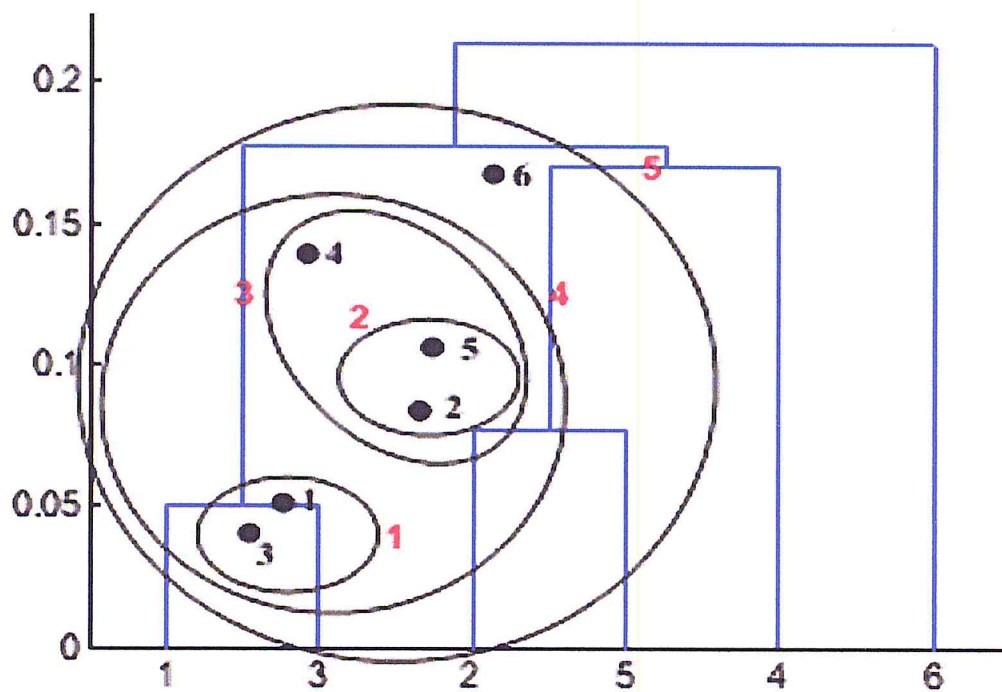


Figure 2.8 : dendrogramme Classification hiérarchique ascendante.

Avantage :

L'un des résultats est le dendrogramme, qui permet de visualiser le regroupement progressif des données. Il est possible d'avoir une idée sur le nombre adéquat de classes dans lesquelles les données peuvent être regroupées

Inconvénients :

La classification hiérarchique ascendante est coûteuse en temps de calcul.

6. Conclusion

Nous venons de présenter dans ce chapitre les différentes méthodes de classification avec ces algorithmes. Plusieurs méthodes sont proposées dans la littérature afin de répondre au problème général de la classification des données.

Selon le nombre d'attributs considérés, il existe deux types de méthodes de classification : monodimensionnelles ou multidimensionnelles. Pour les méthodes de classification multidimensionnelles, il existe des méthodes supervisées comme les K-plus proches voisins (K-PPV), la classification bayésienne et les Séparateurs à Vastes Marges (SVM)...et des méthodes non supervisées comme la Classification Hiérarchique Ascendante (CHA), K-means...

Le prochain chapitre présente un état de l'art sur l'auto-annotation des images à la base de la classification.

Chapitre : 171

État de l'art

1. Introduction

L'annotation automatique des images est un domaine de recherche qui a suscité beaucoup d'intérêt dans la communauté du traitement d'images, Les premières publications sur l'auto-annotation d'images sont apparues récemment vers 1999, L'annotation d'image automatique tente de remplacer un énorme d'efforts manuels pour l'annotation d'image. Les chercheurs proposent et expérimentent plusieurs approches d'annotation d'image afin d'améliorer la qualité d'annotation.

Dans ce chapitre nous abordons l'état de l'art sur les travaux déjà fait pour l'annotation automatique des images à base de classification.

2. Annotation avec les réseaux bayésiens :

Une approche bayésienne semi-naïf d'auto-annotation des images est présentée par *Sabrina Tollari* [20] incorporant la catégorisation avec les contraintes *pair-wise* (deux à deux). Cette approche se divise en deux phases : apprentissage et annotation des images.

➤ Apprentissage : le déroulement de cette phase se fait en deux étapes :

- Segmentation des images en région.
- Catégorisation des régions dans des clusters de région par l'incorporation des contraintes deux à deux, qui sont dérivées du modèle de langage.

➤ Annotation : Au cours de l'étape d'annotation, elle utilise les probabilités de modéliser la relation entre annotations des clusters de région, et à résoudre à l'aide d'un modèle bayésien semi-naïf en 4 étapes comme suit :

1. Formuler comme un problème de probabilité.
2. Calculer la probabilité à posteriori des concepts à partir des clusters de région.
3. Calculer la probabilité à posteriori des clusters de région à partir des régions.
4. Annoter nouvelles images.

Les auteurs ont démontré par expérimentation que l'approche qui combine la catégorisation avec les contraintes *pair-wise* et le modèle bayésien semi-naïf améliore considérablement la performance d'annotation.

3. Annotation avec la méthode de K plus proches voisins

L'idée dans [24] est l'utilisation d'un algorithme génétique 'real coded chromosome' et la classification de K plus proches voisins comme fonction de fitness pour optimiser le poids des descripteurs des caractéristiques des images. Trois schémas sont utilisés :

- L'optimisation du poids avec un algorithme génétique 'real coded chromosome'.
- La sélection de descripteur de caractéristiques optimal avec le code 'binary one'.
- La simultanéité d'optimisation du poids et la sélection de descripteur de caractéristiques optimal avec le 'bi-coded one'.

Les auteurs *Sheridan, S et al* [24] traitent le problème d'annotation d'image comme un problème de classification. L'annotation d'image se fait par un seul mot. Les auteurs ont travaillé avec l'image entière sans segmentation. Le processus d'annotation se fait comme suit :

- Calculer le vecteur de caractéristiques bas niveaux de l'image avec les résultats de l'algorithme génétique.
- Classer l'image dans la catégorie avec classificateur K plus proche voisin.
- Finalement, le mot clé correspondant à la catégorie de l'image est propagé afin d'annoter automatiquement l'image.

Les expérimentations menées par les auteurs ont montré l'efficacité du système d'annotation.

4. Annotation avec séparateur à vaste marges SVM

Dans [21], *Sabrina Tollari* propose un Framework d'auto-annotation qui se résume comme suit :

- Extraction automatique des caractéristiques visuelles multimodales, à partir d'une image de test, selon différentes tailles de grille
- Classement des grilles d'images multi-résolution dans les classes d'objet les plus pertinentes.
- Fusion des images de grilles voisines de la même classe d'objet comme une seule image grille.
- Classement de l'image test dans le concept d'images le plus pertinent en connaissant des classes d'objet.

- Annotation automatique d'image multi niveaux en connaissant les classes d'objet et les concepts d'images.

Les résultats d'expérimentation obtenus, en utilisant des images naturelles, sont assez encourageants.

5. Annotation avec K-means

Les auteurs *Boser.b et al* dans [22], présenté une approche d'auto-annotation des images organisée en trois étapes :

1. Préparation : l'objectif de cette étape est l'extraction et caractérisation des régions d'intérêt pour chaque image.
2. Apprentissage : exécuté un algorithme d'apprentissage permet de regrouper les régions similaire et affect à chaque classe ces annotation correspondants.
3. Annotation automatique : pour annoter automatiquement une nouvelle image utilisant les résultats de la phase d'apprentissage.

Pour expérimenter cette approche les chercheurs utilisés la collection VOC 2009 qui contient 14743 images divisée en deux ensembles la premier pour l'apprentissage et la deuxième pour le test, et un vocabulaire visuel de 4000 mots.

A la fin d'appliquée cette approche pour annoter les images de test, ils ont obtenu des résultats varie entre 62% et 64.6%.

Les problèmes de cette approche est que le classifieur K-means rend l'apprentissage plus difficile et lent à cause de redondance, ainsi la justification du choix de valeur K.

6. Auto-annotation avec Les cartes auto-organisatrices (self organizing map SOM)

Les auteurs *CM. Huang, et al* [27] proposent un Framework incorporant stratégie de pondération avec concurrente carte d'auto-organisation (CSOM) classificateur, Deux classificateurs sont au centre de cette étude: GCSOM (Grille CSOM) et WGCSOM (image entière CSOM). Les deux taches suivantes décrivent le regroupement de SOM algorithme, apprentissage et test pour GCSOM et WGCSOM.

- **Phase d'extraction des caractéristiques**

Telles que la couleur, la texture, forme, et etc. les caractéristiques sont extraits et regroupées dans un vecteur.

- **Phase de classification (CSOM)**

Algorithme de SOM

Le nombre de cartes SOM inclus dans un classificateur de CSOM dépend du nombre de classes dans la base de données. Les étapes de l'algorithme de clustering SOM sont :

Étape 1 : SOM Initialisation: Pour chaque neurone de la carte SOM, le vecteur de poids W sera initialisé de manière aléatoire avec une valeur variant entre 0 et 1.

Étape 2 : BMU identification: Pour la grille d'entrée, le BMU (best matching unit) sur la carte SOM sera identifié en raison de similarité maximale.

Étape 3 : Mise à jour de BMU et voisins: Après l'identification de la BMU de la grille d'entrée X , le poids vecteurs et ses voisins sont mis à jour.

Étape 4 : Analyse histogramme: L'idée principale de l'analyse de l'histogramme est d'analyser et d'affecter un facteur de pertinence R approprié pour le vecteur de poids dans chacun des neurones affectés par des données d'entrée. Par conséquent, le poids dominant dans un neurone sera identifié.

- **Phase d'auto-annotation**

Pour lier un mot-clé à un blob, une table de correspondance sera construite comme suit :

L'image sera décrite par une matrice où chaque ligne représente l'image, les g premières colonnes correspondent les g mots clés, et les b colonnes correspondent à b blobs.

Après les images de test sont classés, une table avec sa classe correspondant sera utilisé pour effectuer la sélection des mots-clés comme annotation d'image.

Les auteurs (*CM. Huang, et al*) montrent des bons résultats avec un taux de correspondance est de 33,33% à 100% et le taux moyen de l'annotation correcte atteint 78,44% si l'image est affectée à la bonne classe.

6. Conclusion

Nous avons vu dans ce chapitre les différents travaux d'annotation automatique des images déjà réalisée. Chaque travail utilise une approche et méthode de classification différente dans le but d'améliorée la qualité d'annotation. Les résultats obtenus sont différent d'un travail à autre. Mais les approche qui utilise la segmentation sont les plus performants.

Dans le prochain chapitre, nous passons à la conception et l'implémentation de notre application d'auto-annotation des images basée sur plusieurs méthodes de classification.

Chapitre : IV

Conception et mise en œuvre des systèmes AIA

1. Introduction :

Dans ce chapitre nous allons présenter la conception et l'implémentation d'un système d'annotation automatique des images. Avec le langage de modélisation UML. Ce dernier est composé de deux phases : apprentissage et annotation d'une image de test, nous donnons dans ce chapitre une explication de chaque étape.

Nous présentons au départ les outils utilisés et par la suite les différents modules de notre système, avec une description détaillée de chaque étape du processus d'auto-annotation, nous montrons des captures d'écrans des différentes interfaces et les résultats obtenus.

2. Conception :

2.1. Définition d'UML :

UML (Unified Modeling Language) est un langage de modélisation orienté objet, possède 9 diagrammes parmi ces derniers nous avons choisi le Diagramme de cas d'utilisation et diagramme de classe qui répondent à nos besoins [28].

2.2 Diagramme de cas d'utilisation:

Les cas d'utilisation représentent les fonctions que le système doit faire sans spécifier comment sont-ils faits.

Le diagramme suivant représente les tâches principales pour notre application :

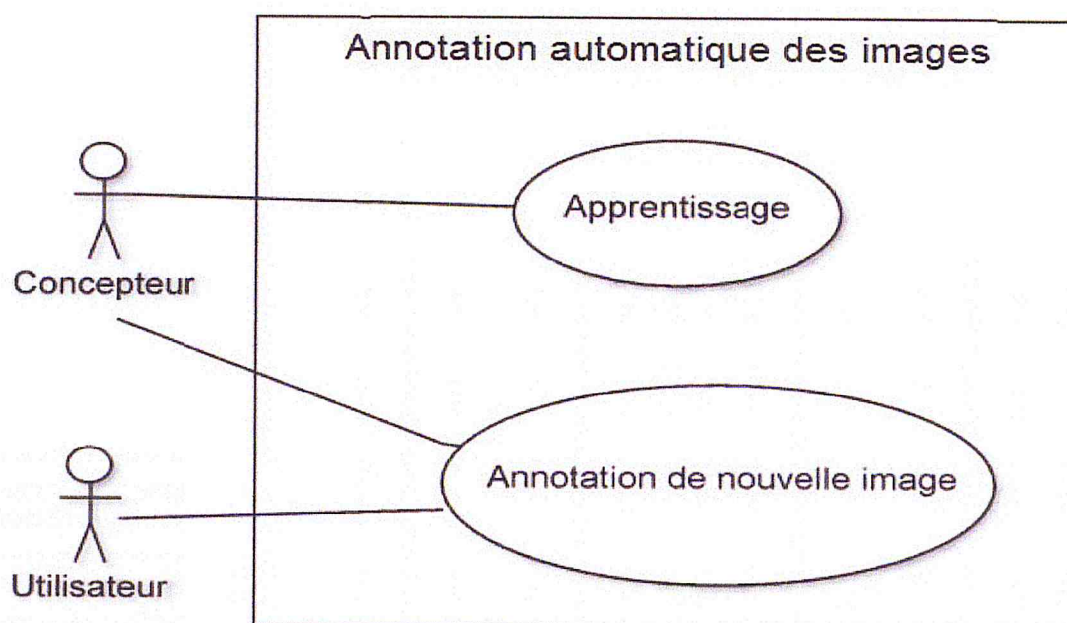


Figure 4.1: Diagramme de cas d'utilisation global.

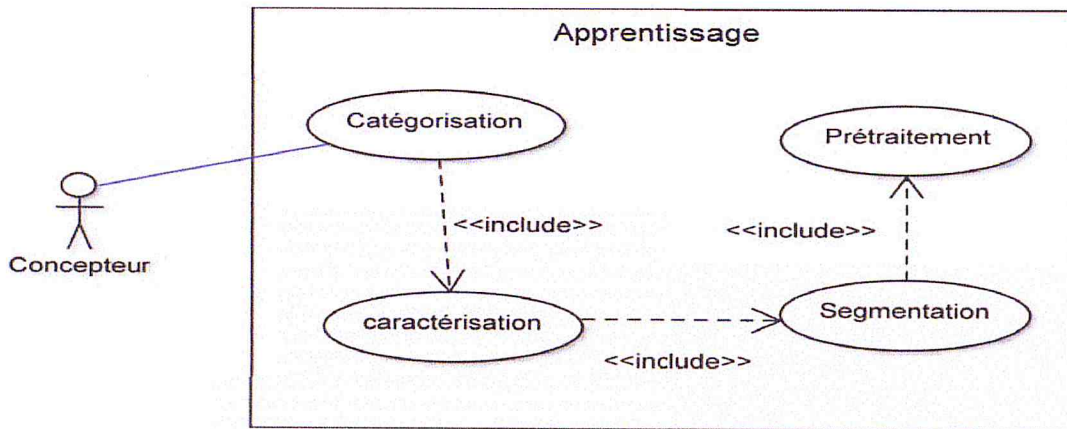


Figure 4.2: Diagramme de cas d'utilisation apprentissage.

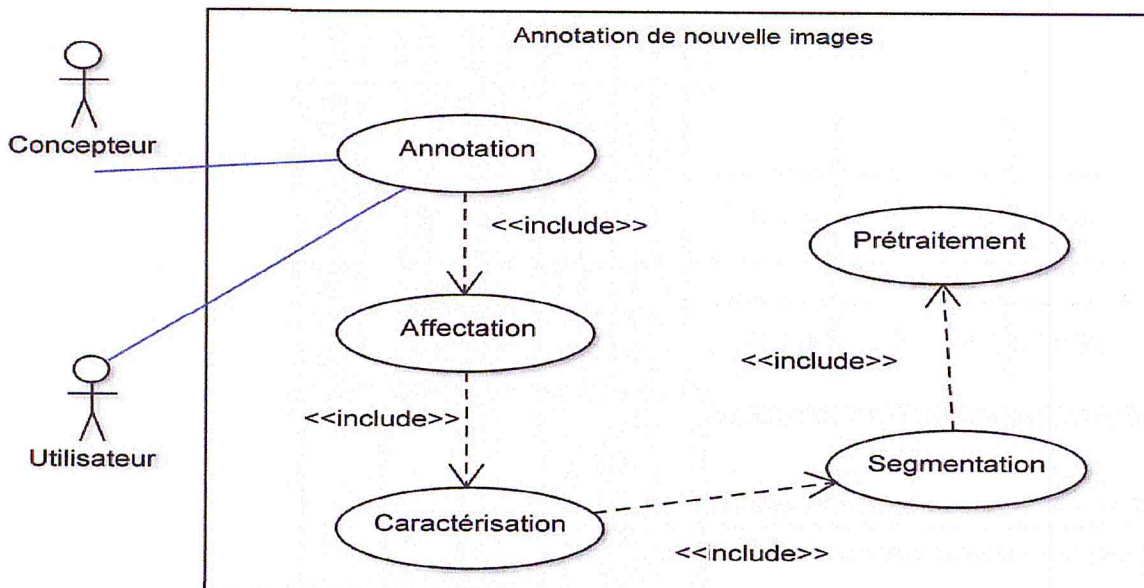


Figure 4.3 : Diagramme de cas d'utilisation annotation de nouvelle images

2.3. Diagramme des Classes

Le diagramme de classes est considéré comme le plus important de la modélisation orientée objet, il est le seul obligatoire lors d'une telle modélisation.

Alors que le diagramme de cas d'utilisation montre un système du point de vue des acteurs, le diagramme de classes en montre la structure interne. Il permet de fournir une représentation abstraite des objets du système qui vont interagir pour réaliser les cas d'utilisation.

La figure suivant représente le diagramme de classe de la base de données de notre application.

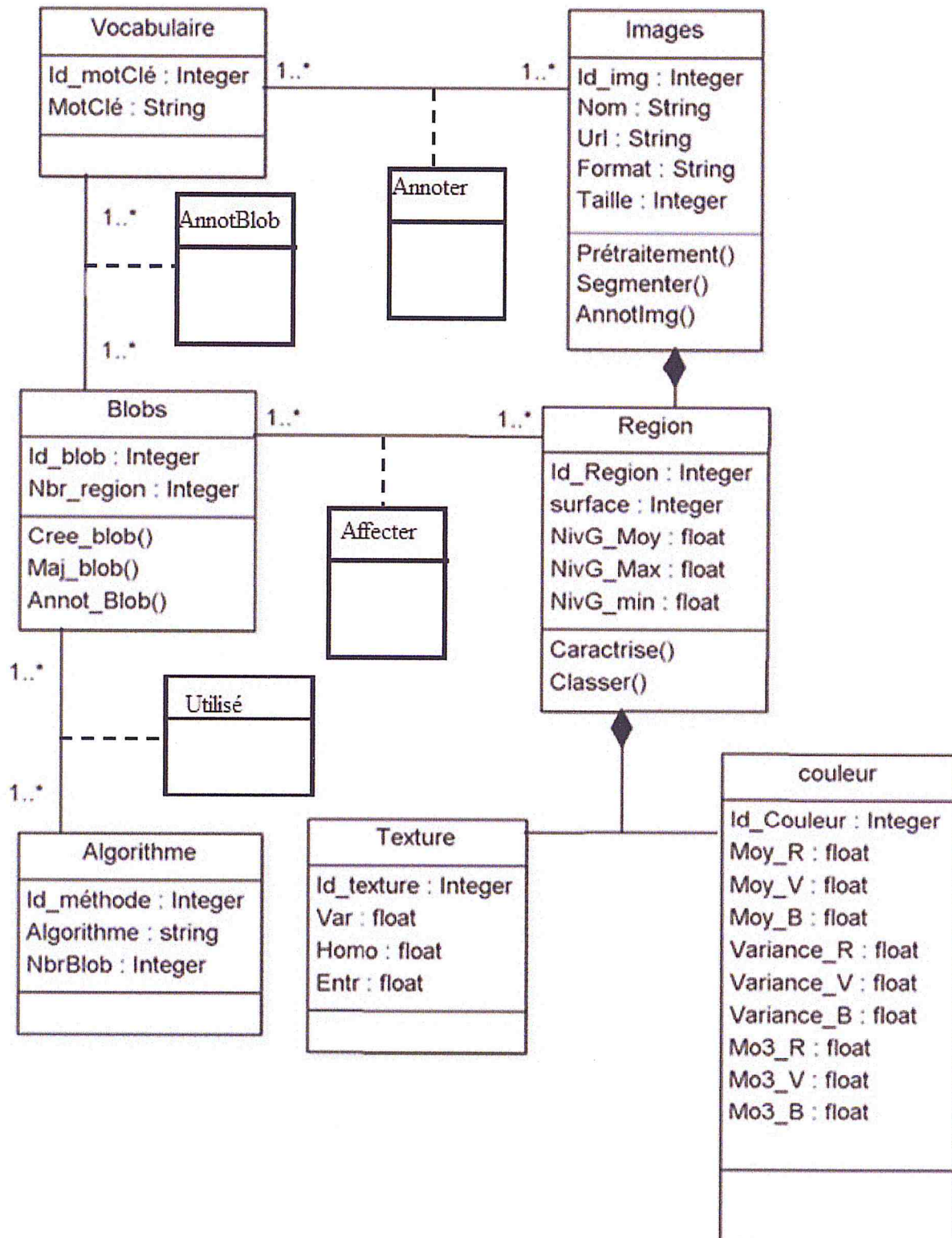


Figure 4.4: Diagramme des classes au niveau logique.

2.2.1. La description des class:

Classe	Attributs	Code Attribut	Type
Images	Identifiant de l'image	Id_img	Numérique
	Nome de l'image	Nom	Alphabétique
	Format de l'image	Url	Alphabétique
	Taille de l'image	Format	Numérique
Région	Identifiant de région	Id_region	Numérique
	Surface de région	Surface	Numérique
	Niveau de gris moyen	NivG_Moy	Numérique
	Niveau de gris max	NivG_Max	Numérique
	Niveau de grise min	NivG_Min	Numérique
Blobs	Identifiant de blob	Id_Blob	Numérique
	Nombre de région	Id_region	Numérique
Algorithme	Identifiant de méthode	Id_méthode	Numérique
	Nom de l'algorithme	Algorithme	Alphabétique
	Nombre de blob	NbrBlob	Numérique
Vocabulaire	Identifiant de mot clé	Id_motClé	Numérique
	Mot clé	MotClé	Alphabétique
Couleur	Identifiant de couleur	Id_Couleur	Numérique
	Moyen rouge	Moy_R	Numérique
	Moyen vert	Moy_V	Numérique
	Moyen bleu	Moy_B	Numérique
	Variance rouge	Variance_R	Numérique
	Variance vert	Variance_V	Numérique
	Variance bleu	Variance_B	Numérique
	Moment d'ordre 3 de Rouge	Mo3_R	Numérique
	Moment d'ordre 3 de Vert	Mo3_V	Numérique
Moment d'ordre 3 de bleu	Mo3_B	Numérique	
Texture	Identifiant de textureur	Id_texteur	Numérique
	Variance	Var	Numérique
	Homogénéité	Homo	Numérique
	Entropie	Entr	Numérique

Tableau 4.1: Description des classes

2.2.2. Description des méthodes :

Les méthodes sur les classes sont expliquées comme suit :

La classe *Image* contient trois méthodes sont :

- *Prétraitement* : Elle permet de charger une image et faire des traitements de base.
- *Segmenter ()* : Elle permet de segmenté l'image en régions.
- *AnnotImg ()* : Elle permet d'annoter une image.

La classe *Région* contient deux méthodes sont :

- *Caractrise ()* : Elle permet d'extraire les caractéristiques (couleur, texture) pour chaque région.
- *Classer ()* : Affecter une région aux blob similaire.

La classe *Blobs* contient trois méthodes sont :

- *Cree_blob ()* : Elle permet de créer un blob.
- *Maj_blob ()* : Elle permet d'ajouter des régions aux blobs.
- *Annot_Blob ()* : Elle permet d'annoter les blobs.

3. Mise en œuvre un système d'auto-annotation des images :

A fin d'implémenter le système d'auto-annotation d'image à base de classification, nous avons opté à travailler avec builder C++, qui est un logiciel de développement rapide des applications conçu par borland en 2006 en raison de sa facilité de développement et de ses possibilités de traitement d'images.

Les données du processus d'apprentissage sont sauvegardées dans une base de données relationnelle sous le SGBD relationnel Access, qui est un SGBD relationnel édité par Microsoft. Il fait partie de la suite bureautique MS Office Pro. La base des images annotées est également sauvegardée avec le même SGBD

Comme nous avons déjà présenté dans le premier chapitre le système d'annotation automatique des images ce déroule en deux phases : la phase d'apprentissage qui permet de crée une base de connaissances pour le système a l'aide d'une base des images annotée manuellement (pour notre projet nous utilisons corpus de Benchmark), et la phase d'auto-annotation d'une nouvelle image.

3.1. Corpus Benchmark :

La collection d'images Benchmark se compose de 20 000 images fixes prises à partir d'emplacements naturels dans le monde entier et comprenant une section hétéroclite d'images fixes naturelles. Les 20 000 images sont de haute qualité, multi-objets tel que des

photographies de personnes, des animaux, des villes, des paysages et de nombreux autres aspects de la vie contemporaine.

Chaque image de corpus est annotée manuellement avec des mots clé en langue anglais. Ces derniers sont pris d'un vocabulaire de 267 mots. Les données sont placées dans des dossiers (40 dossiers en totalité) où chacun contient des images et un fichier qui associe chaque image à ses mots clé correspondant [29].

3.2. Système d'auto-annotation d'images à base de classification:

La figure suivante représente l'interface principale de système qui est composé de quatre menus principaux.

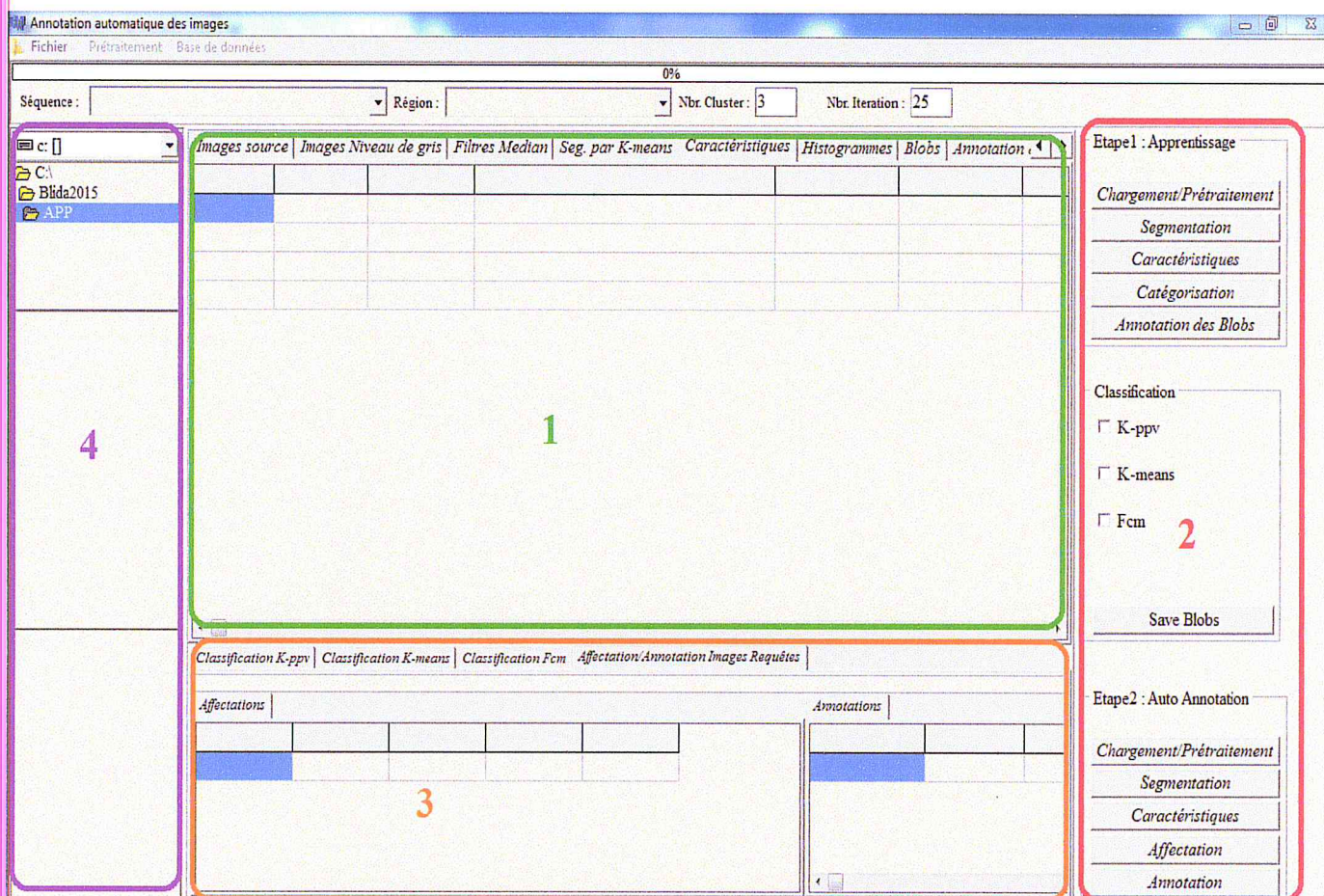


Figure 4.5: la fenêtre principale du système.

Tel que:

1. Menu d'affichage (image, segmentation, statistique,...).
2. Menu des tâches (Apprentissage, annotation).
3. Menu des paramètres (variable initiaux).
4. Menu des fichiers (pour le chargement des images).

3.2.1. Phase d'apprentissage:

Nous appliquons les différentes étapes d'apprentissage sur une image et nous exposons les résultats de chaque étape. La figure suivant présent l'interface de chargement des images.

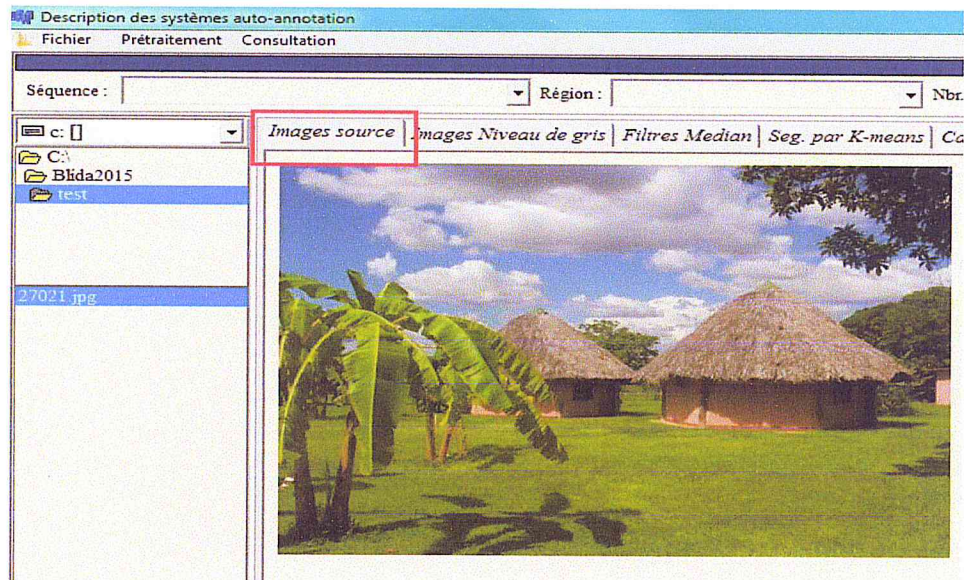


Figure 4.6: chargement de nouvelle image

3.2.1.1. Prétraitement:

Niveau de gris:

Pour convertir une image en couleur à une image niveau de gris, il faut remplacer pour chaque pixel les trois valeurs représentant les niveaux de rouge, de vert et de bleu, en une seule valeur représentant la luminosité, et enregistrer dans un fichier ou envoyer à un terminal en noir et blanc (écran). Il existe plusieurs formule pour calculé niveau de gris parmi elle nous avons utilisé la formule suivant : $Gris=(R+V+B)/3$

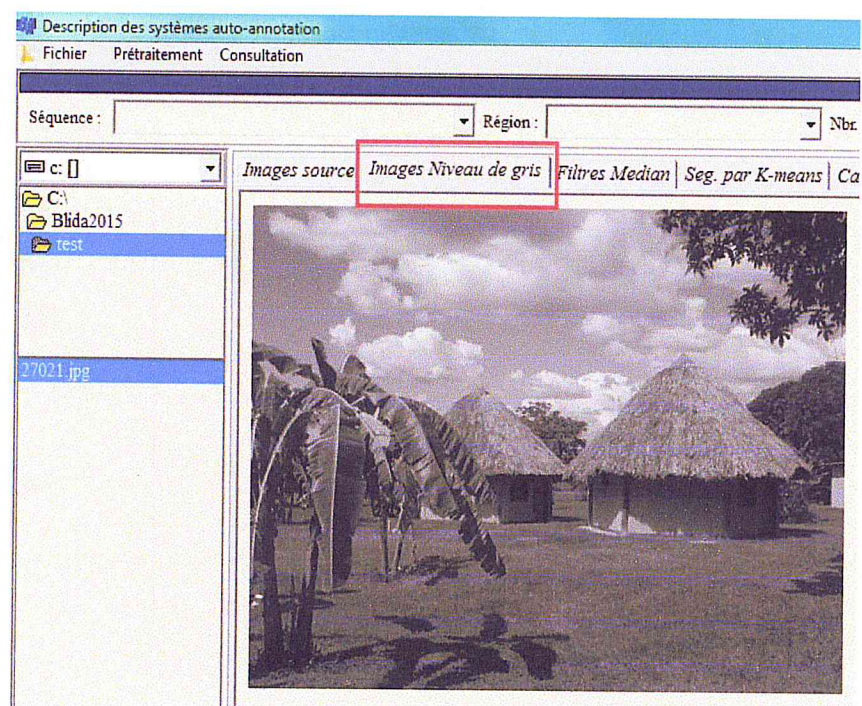


Figure 4.7: résultat de niveau de gris.

Filtre médian

Le filtre médian est un filtre numérique non linéaire, souvent utilisé pour la réduction de bruit. La réduction de bruit est une étape de prétraitement classique visant à améliorer les résultats de traitements futurs (détection de bords par exemple). La technique de filtre médian est largement utilisée en traitement d'images numériques car il permet sous certaines conditions de réduire le bruit tout en conservant les contours de l'image. Se dernier est calculé par la méthode suivant :

- 1) Trier les valeurs du voisinage par ordre croissant.
- 2) Sélectionner la médiane des valeurs.
- 3) Attribuer cette valeur médiane au niveau de sortie [30].

3.2.1.2. Segmentation:

Cette étape permet de diviser les images du corpus en régions. Pour cela nous avons utilisée l’algorithme *K-means* qui se résume en quatre étapes :

1. Initialisation de chaque centre u_k ;
2. Pour chaque pixel(x,y) calculer la distance $d(f(x,y),u_k)$ aux différents centres des classes u_k et affecter à la classe la plus proche $C_j = \text{argmin}_k d(f(x,y),u_k)$ avec $d(f(x,y),u_k) = |f(x,y) - u_k|$
3. Mise à jour de nombre de pixels et des centres u_k des class ;

$$u_k = \frac{\sum_{(x,y) \in C_k} f(x,y)}{N_k}$$
4. Arrêt si $N_k = N_{k+1} \in C_k$, sinon retour à l’étape 2.

Algorithme 4.1: Algorithme de segmentation *k-means* [31].

L'algorithme K-means est un algorithme basé sur la classification non supervisée qui ne nécessite pas la présence d'une base d'apprentissage, de ce qui fait, cet algorithme permet d'organiser les pixels d'image sous forme de classes, c'est à dire que chaque pixel doit appartenir à une classe bien déterminée sous l'intention de plusieurs critères.

Nous appliquons cette algorithme avec nombre de cluster =5, le résultat d'exécution de ce module est présenté dans la *figure 4.8* :

Une image x avant la segmentation..

Le résultat d'une segmentation K-means sur l'image x peut être représenté comme suit :

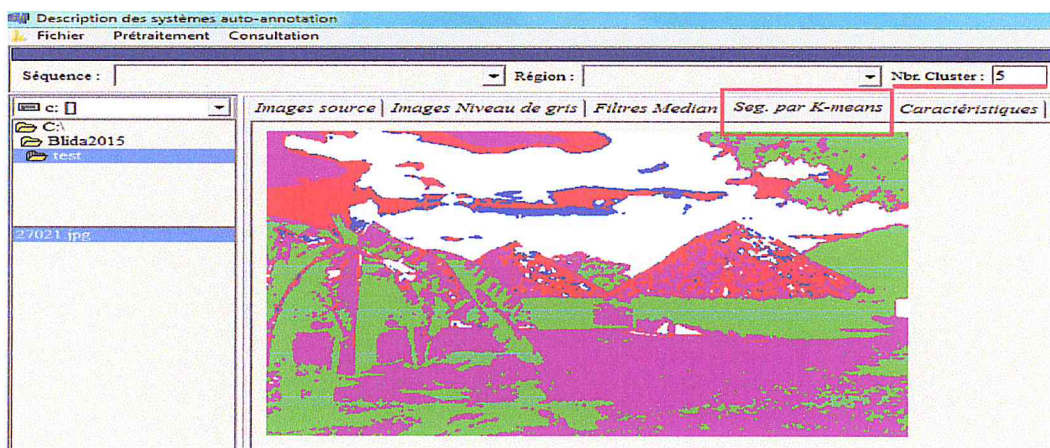


Figure 4.10 : résultat de la segmentation.

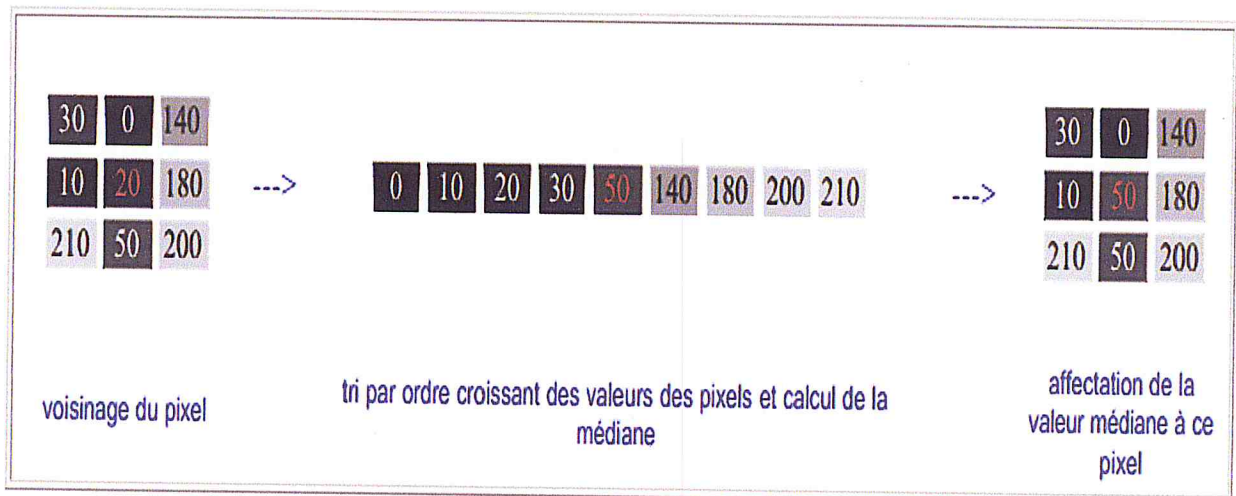


Figure 4.8: calcul de filtre médian.

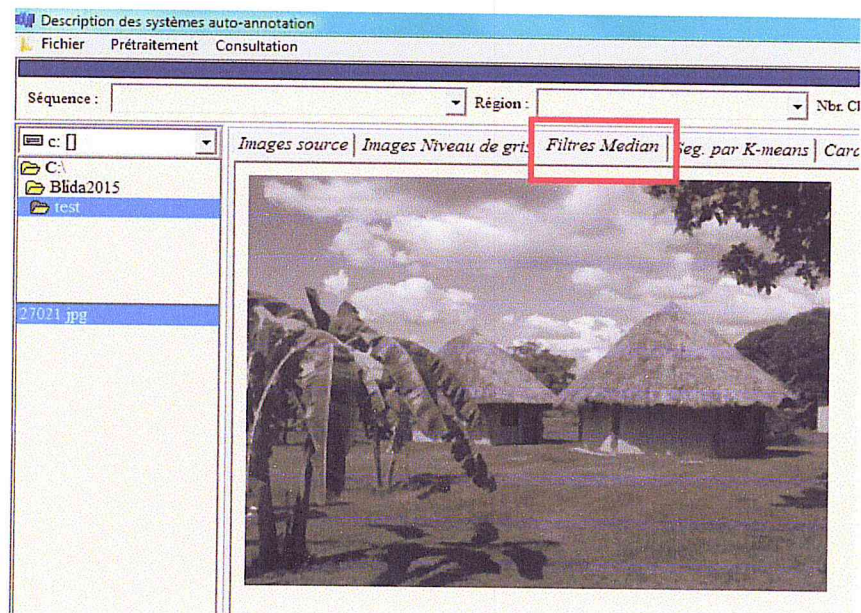


Figure 4.9: Résultat de filtre médian.

3.2.1.3. Caractérisation :

Cette phase de système permet de caractériser les régions obtenues par segmentation. Pour chaque région obtenue nous extrayons les moments statistiques de la couleur : la moyenne, la variance et les moments d'ordre trois de chaque composante d'espace RGB. Nous extrayons également les trois paramètres de la caractéristique visuelle Texture : La variance, l'homogénéité, le contraste. La *figure 4.9* présente les résultats de la caractérisation d'une image. Ces dernier doivent être sauvegardées dans la base de données pour l'utilisé dans la prochaine étape (catégorisation)

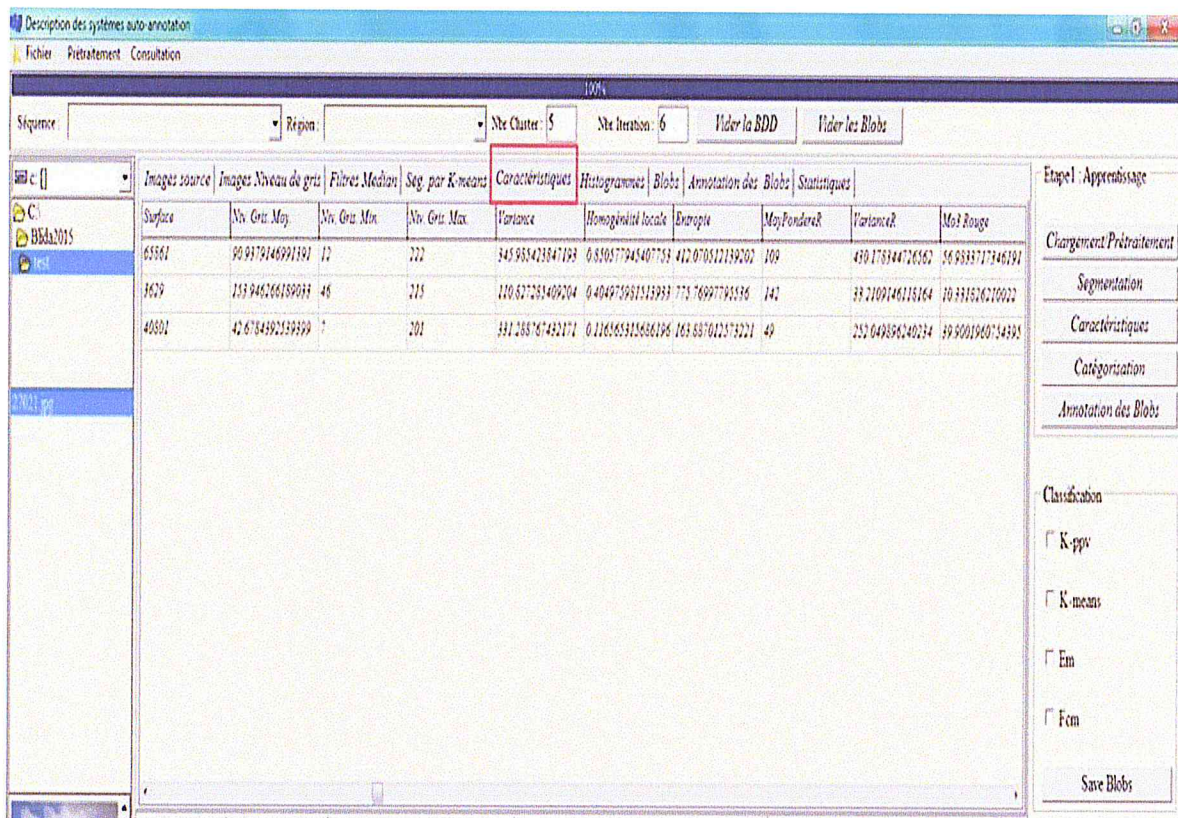


Figure 4.11: résultat de la caractérisation.

3.2.1.4. Catégorisation:

Cette étape permet de regrouper les régions visuellement similaires à des classe appelée blobs. Pour cela et selon les études que nous avons déjà fait sur les méthodes de classification dans chapitre 2 (l'avantage et l'inconvénient de chaque méthode), et selon les résultats présentés par les auteurs dans les travaux de l'état de l'art (chapitre 3) nous décidons d'utilisée les trois méthodes suivant pour classifier les K-ppv, K-means et FCM

La méthode de classification choisie doit être appliquée sur l'ensemble des vecteurs de caractéristique des régions afin de construire des groupes visuellement similaires appelé blobs. La figure suivante représente pour chaque blob l'ensemble de ces régions.

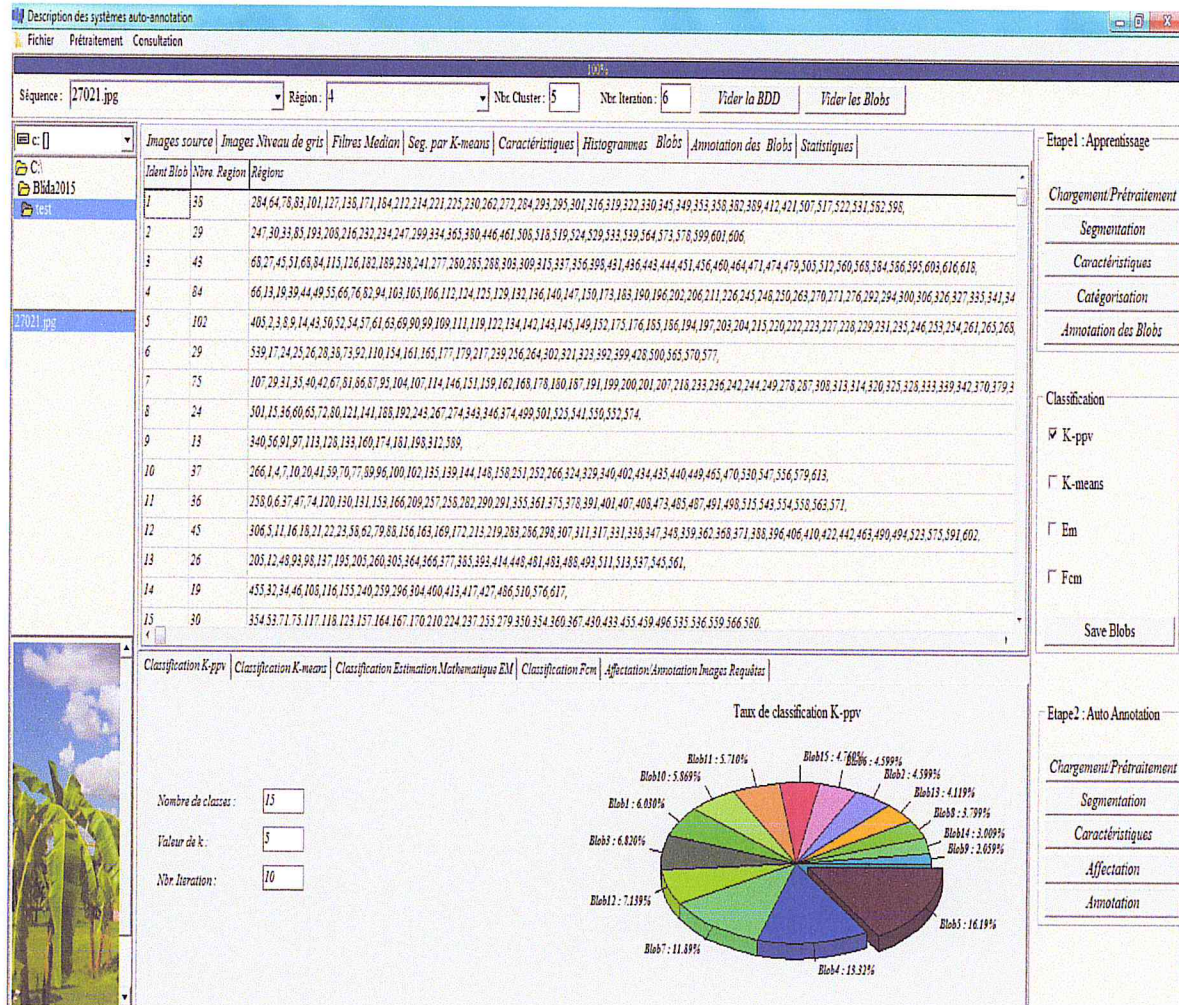


Figure 4.12: résultat de la catégorisation.

3.2.1.5. Annotation des blobs:

Dans cette phase le système exécute l'algorithme *Annot_Blob()*, l'idée de base de cet algorithme est la suivante : pour annoter un blob, nous faisons hériter les mots clé des images auxquelles les régions contenues dans ce blob appartiennent. la fréquence d'apparition de mot clé relative aux nombre des images qui sont annotées manuellement avec ces mots clé.

La figure suivant représente l'annotation des blobs avec la fréquence de chaque mots clé.

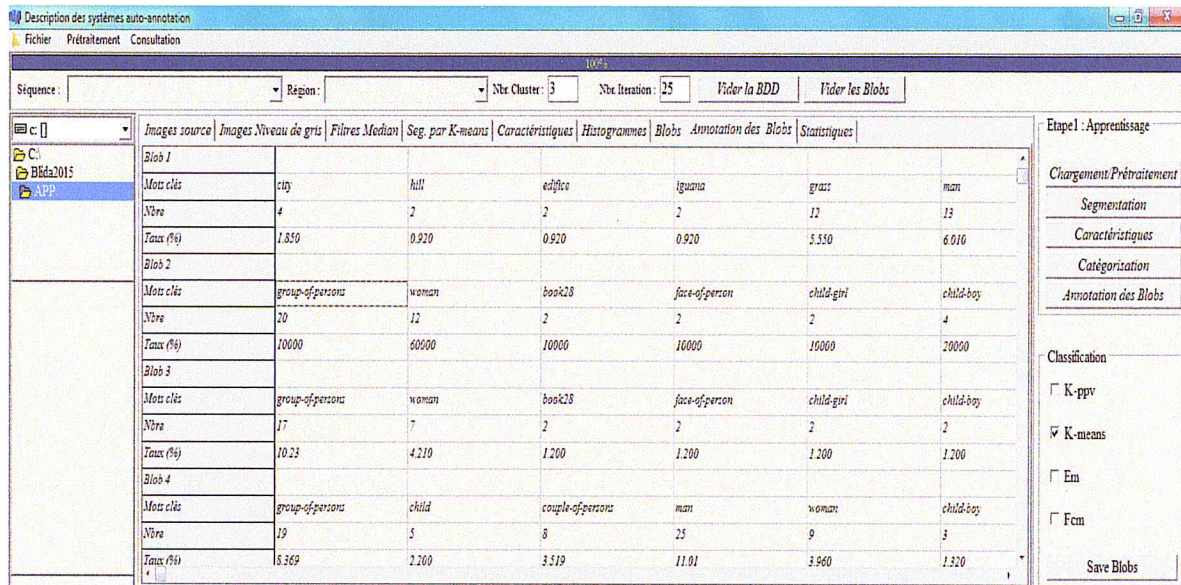


Figure 4.13: résultat d’auto-annotation des blobs.

3.2.2. Phase d’auto-annotation de nouvelle image:

Cette phase permet de système d’annoter automatiquement une nouvelle image, elle est similaire à l’étape d’apprentissage dans les modules prétraitement, segmentation et caractérisation. La déférence est dans les modules affectation et auto-annotation d’une nouvelle image.

3.2.2.1. Affectation:

Cette étape permet d’affecter les régions de la nouvelle image aux blobs pour chaque méthode de classification appliquée,

La Figure suivante représente l'affectation des régions pour chaque méthode de classification.

Images	Regions	Blob/Kppv	Blob/Kmeans	Blob/Em	Blob/Fcm
13579.jpg	1	4	1		3
13579.jpg	2	8	2		2
13579.jpg	3	5	3		2

Figure 4.14: résultat d'affectation des régions au blobs.

3.2.2.2. Annotation :

L'objectif de ce module est d'annoter automatiquement une nouvelle image à partir des blobs annotés. Nous sélectionnons pour chaque blob dont la région appartient un sous ensemble des mots clé de ce blob tel que l'ensemble d'intersection de ces derniers représente l'annotation de cette image. Les figures suivantes présentent les résultats d'annotation selon les différentes méthodes de classification.

Images	Regions	Kppv	Kmed
13579.jpg	1	rock, group-of-persons,	group
13579.jpg	2	group-of-persons, man,	group
13579.jpg	3	man, woman,	man,

Figure 4.15: Résultat d'annotation des régions selon k-ppv

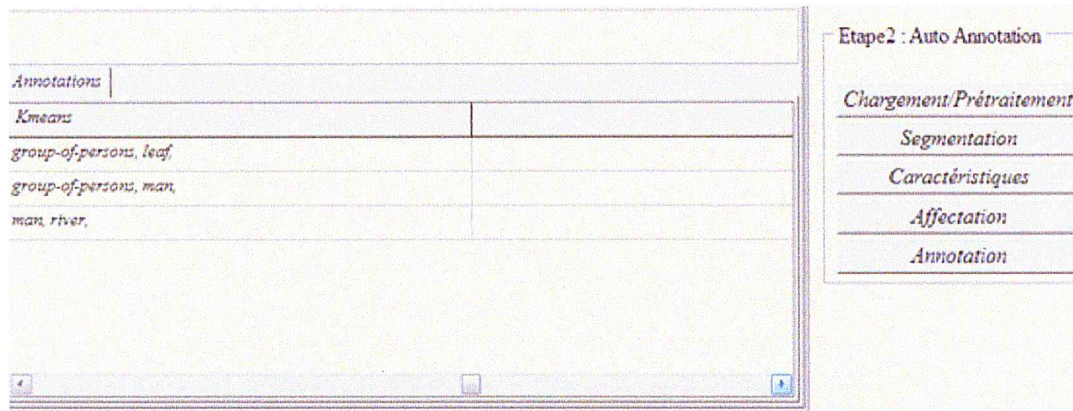


Figure 4.16: Résultat d'annotation des régions selon k-means

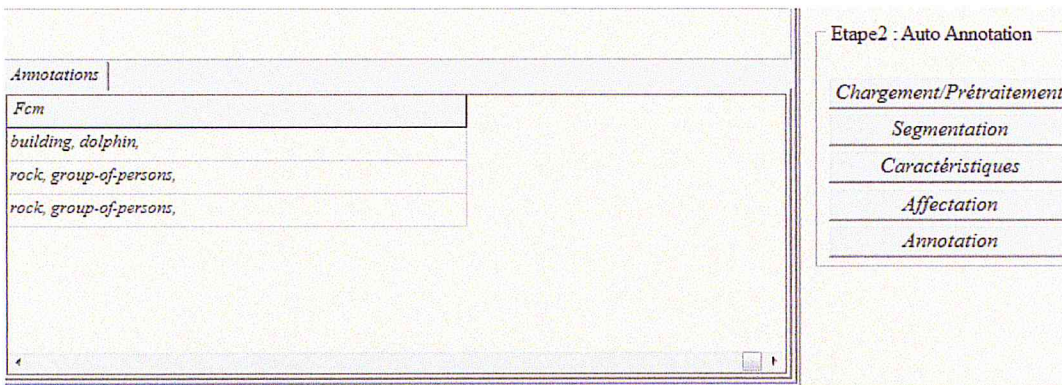


Figure 4.17: Résultat d'annotation selon FCM

La figure suivante montre les résultats d'annotation automatique d'une image pour chaque méthode

Annotation de l'Image requête :


Image Requête	Annotations Kppv	Ann	K-means	FCM	Ann
	wall		face-of-person	wall	
	bed		child-girl	bed	
	face-of-person		ground	face-of-person	
	window		group-of-persons	window	
	child-boy		cloud	child-boy	
	lamp		leaf	lamp	
	wooden-furniture		palm	wooden-furniture	
	curtain		cup	curtain	
	child-girl		swimming-pool	child-girl	
	floor-other		kitchen-pct	floor-other	
	group-of-persons		glass	group-of-persons	
	non-wooden-furniture		mountain	non-wooden-furniture	
			wall		
			window		

Figure 4.18: Résultat final d'annotation.

4. Conclusion :

Nous avons présenté à travers ce chapitre la conception et l'implémentation de système d'auto-annotation des images qui utilise les méthodes de classification k-ppv, k-means et FCM, le système est composé de deux partie, apprentissage et auto-annotation d'une nouvelle image.

Nous utilisons un apprentissage sur un corpus benchmark des images annotées. Selon la méthode de classification choisie, le système va classifier les régions visuellement similaires dont les régions appartiennent. Pour une nouvelle image, Après segmentation en région et l'extraction des caractéristiques nous affectons chaque région de cette nouvelle image au blob le plus similaire pour chaque méthode de classification pour déduire l'annotation de cette image.

Dans le chapitre qui se suit, nous expérimentant se système pour pouvoir comparer les méthodes de classification utilisée.

Chapitre : V

Tests et validation des résultats

1. Introduction :

Nous avons vu dans le chapitre précédent la conception et l'implémentation de système d'annotation automatique des images à base des méthodes de classification.

Dans ce chapitre, nous allons les tester afin de mettre en chaque méthode de classification et comparé les résultat obtenu pour savoir quelle est la meilleur méthode de classification qui donne des meilleurs résultat dans le processus d'auto-annotation des images.

2. Apprentissage :

2.1. Corpus :

Pour tester les systèmes réalisés nous avons utilisé pour l'apprentissage une base des images annotées manuellement (Benchmark). Nous sélectionnons 1000 images naturelles.

2.2. Segmentation :

Dans cette partie nous utilisons l'algorithme de segmentation *K-means* pour segmentée les images d'apprentissage. Afin de chargée la base des images nous appliquons un algorithme qui permet de changer les images RGB vers images niveau de gris, puis nous appliquons le filtre médian ces deux tache permettant de réduire le travaille de segmentation et amélioré ses qualité. D'après plusieurs tests que nous avons faits nous décidons de mettre $k=3$.

2.3. Caractérisation :

Dans cette étape nous extrayons pour chaque région un vecteur des caractéristiques contient des informations concernent la texture et la couleur. Puis sauvegardée tout ces informations dans la base de données.

2.4. Catégorisation :

C'est l'étape la plus importante pour nous, chaque méthode de classification doit être appliquée pour l'ensemble des régions pour construire des blobs. Pour cela et d'après les études que nous avons fait dans le chapitre II (avantages et inconvénients de chaque méthode) et selon les travaux déjà fait dans ce domaine, nous avons décidé d'appliqué les trois méthodes de classification suivantes : k-ppv, k-means et FCM, et pour la raisons de faire une étude comparative entre ces méthode nous choisissons nombre de classe de départ $k=15$.

2.5. Annotation des blobs :

A la fin de classifier les régions dans des blobs nous appliquons la méthode d'annotation des blob présenté dans chapitre IV pour chaque blob. Et ver la fin nous sauvegardons ses résultat dans la base de données. Afin de cette étape le système termine l'étape d'apprentissage.

3. Annotation des images de test :

Dans cette phase nous avons appliquée le système d'auto-annotation des images sur un échantillon de dix (10) images différentes qui ont été annotée manuellement. En changeant à chaque fois la méthode de classification par une des trois méthodes choisie, et en compare les résultats obtenu par rapport au résultat d'annotation manuelle.

Les tableaux suivants présentent quelque aperçu des résultats de tests que nous avons fait :



Annotation automatique avec K ppv	Annotation automatique avec K-means	Annotation Automatique avec FCM	Annotation manuelle
<ol style="list-style-type: none"> 1. <u>bed</u> 2. <u>wall</u> 3. curtain <u>wooden-furniture</u> 4. lamp 5. face-of-person 6. painting 7. window 8. group-of-persons 9. building 10. <u>floor-carpet</u> 11. sky-blue 12. child-girl 13. floor-other 14. child-boy 15. non-wooden-furniture 	<ol style="list-style-type: none"> 1. <u>wall</u> 2. window 3. lamp 4. <u>bed</u> 5. ground 6. group-of-persons 7. chimney 8. cabin 9. branch 10. rock 11. tree 12. statue 13. <u>wooden-furniture</u> 14. man 15. person 16. can 17. roof 18. mountain 19. bench 	<ol style="list-style-type: none"> 1. <u>wall</u> 2. <u>bed</u> 3. face-of-person 4. window 5. child-boy 6. lamp 7. <u>wooden-furniture</u> 8. curtain 9. child-girl 10. floor-other 11. group-of-persons 12. non-wooden-furniture 	<ol style="list-style-type: none"> 1. bed 2. floor-carpet 3. telephone 4. wooden furniture 5. bottle 6. wall



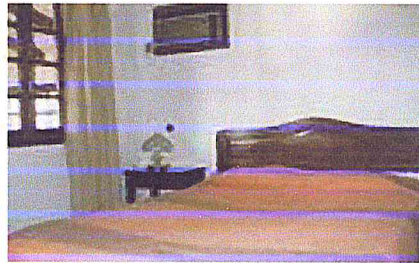
Annotation automatique avec K ppv	Annotation automatique avec K-means	Annotation Automatique avec FCM	Annotation manuelle
1. bed 2. wall 3. curtain 4. wooden-furniture 5. lamp 6. face-of-person 7. painting 8. window 9. group-of-persons 10. building 11. floor-carpet 12. sky-blue 13. child-girl 14. cloth 15. child-boy 16. ground 17. rock	1. wall 2. window 3. lamp 4. bed 5. ground 6. group-of-persons 7. chimney 8. cabin 9. branch 10. rock 11. tree 12. statue 13. face-of-person 14. child-boy 15. wooden-furniture 16. child-girl 17. woman 18. cloud 19. leaf 20. palm 21. cup 22. swimming-	1. wall 2. bed 3. face-of-person 4. child-girl 5. man 6. curtain 7. lamp 8. group-of-persons 9. window 10. wooden-furniture 11. woman 12. building 13. cloth 14. child-boy 15. cloud 16. grass 17. rock	Wall Child-girl



Annotation automatique avec K ppv	Annotation automatique avec K-means	Annotation Automatique avec FCM	Annotation manuelle
1. wall 2. bed 3. face-of-person 4. wooden-furniture 5. lamp 6. <u>child-girl</u> 7. window 8. floor-other 9. child-boy 10. curtain 11. non-wooden-furniture 12. <u>group-of-persons</u> 13. building 14. painting 15. floor-carpet 16. sky-blue	1. wall 2. wooden-furniture 3. bed 4. man 5. person 6. can 7. lamp 8. roof 9. mountain 10. bench 11. window 12. ground 13. <u>group-of-persons</u> 14. chimney 15. cabin 16. <u>branch</u> 17. <u>rock</u> 18. tree 19. statue	1. wall 2. bed 3. face-of-person 4. window 5. child-boy 6. lamp 7. wooden-furniture 8. curtain 9. <u>child-girl</u> 10. floor-other 11. <u>group-of-persons</u> 12. non-wooden-furniture	1. Groupe-of-persons 2. Child-girl 3. rock 4. branch 5. Woman 6. Public-sign 7. Sky-light 8. construc-tion



Annotation automatique avec K ppv	Annotation automatique avec K-means	Annotation Automatique avec FCM	Annotation manuelle
1. wooden-furniture 2. lamp 3. <u>face-of-person</u> 4. painting 5. window 6. group-of-persons building 7. floor-carpet 8. sky-blue 9. child-girl 10. floor-other 11. child-boy 12. non-wooden-furniture	1. wall 2. window 3. lamp 4. bed 5. <u>ground</u> 6. group-of-persons 7. chimney 8. cabin 9. branch 10. <u>rock</u> 11. tree 12. statue 13. wooden-furniture 14. man 15. person 16. can 17. roof 18. mountain 19. bench	1. wall 2. bed 3. <u>face-of-person</u> 4. window 5. child-boy 6. lamp 7. wooden-furniture 8. curtain 9. <u>child-girl</u> 10. floor-other 11. group-of-persons 12. non-wooden-furniture	1. Rock 2. Child-girl 3. Face-of-person 4. Ground



Annotation automatique avec K ppv	Annotation automatique avec K-means	Annotation Automatique avec FCM	Annotation manuelle
<ol style="list-style-type: none"> 1. <u>bed</u> 2. <u>wall</u> 3. <u>curtain</u> 4. <u>wooden-furniture</u> 5. <u>lamp</u> 6. face-of-person 7. painting 8. <u>window</u> 9. group-of-persons 10. building 11. floor-carpet 12. sky-blue 13. child-girl 14. floor-other 15. child-boy 16. <u>non-wooden-furniture</u> 	<ol style="list-style-type: none"> 1. <u>wall</u> 2. <u>wooden-furniture</u> 3. <u>bed</u> 4. man 5. person 6. can 7. <u>lamp</u> 8. roof 9. mountain 10. bench 11. face-of-person 12. child-boy 13. group-of-persons 14. child-girl 15. ground 16. <u>window</u> 17. woman 18. rock 	<ol style="list-style-type: none"> 1. <u>wall</u> 2. <u>bed</u> 3. face-of-person 4. <u>window</u> 5. child-boy 6. <u>lamp</u> 7. <u>wooden-furniture</u> 8. <u>curtain</u> 9. child-girl 10. floor-other 11. group-of-persons 12. <u>non-wooden-furniture</u> 	<ol style="list-style-type: none"> 1. bed 2. Window 3. Curtain 4. Lamp 5. Wooden-furniture 6. non-wooden-furniture 7. Wall

5. Résultats d'expérimentations:

La table suivant représente le nombre de mots clés obtenus correct pour chaque méthode de classification par rapport l'annotation manuelle qui se trouve dans le corpus.

Image	1	2	3	4	5	6	7	8
Annotation manuelle	6	3	2	4	8	7	5	8
K-ppv	5	3	2	2	2	7	1	4
K-means	3	1	2	2	2	5	3	2
FCM	3	3	2	2	2	6	1	4

Tableau 5.1: Résultats d'annotation pour huit images.

L'histogramme suivant représente le taux de correspondance pour chaque méthode de classification :

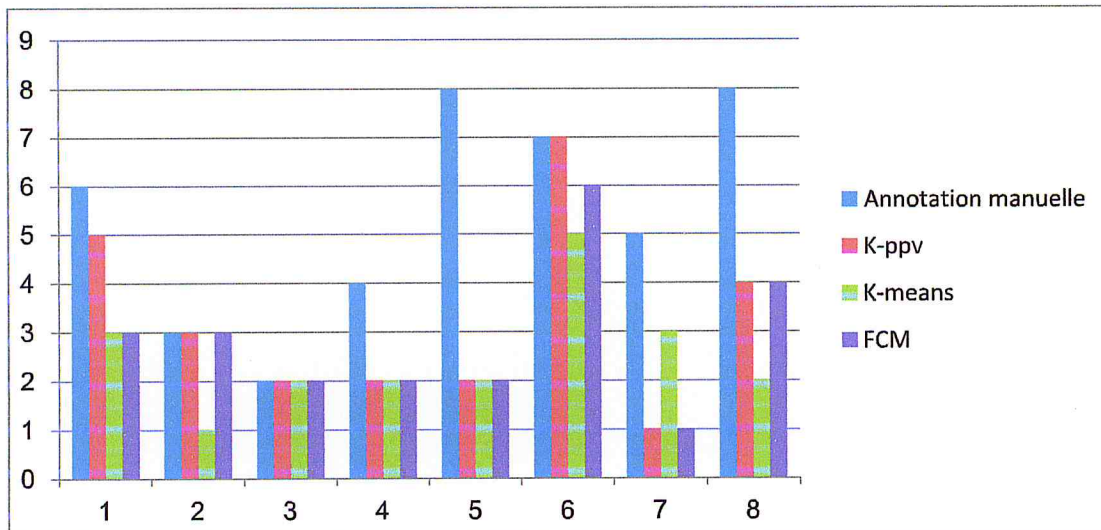


Figure 5.1 : Histogramme de comparaisons entre les différentes classifications.

5. Interprétation des résultats :

D'après les tests que nous avons faits on déduit les résultats suivant :

- K-ppv : a un taux de réussite varie entre 20 % et 100 %.
- K-means : a un taux de réussite varie entre 33.33 % et 100 %.
- FCM : a un taux de réussite varie entre 33.33 % et 100 %.

Ces résultats ne permettent pas de favoriser une méthode à autre, car les résultats de comparaison entre les trois méthodes varie selon les images est pour deux image déferent on trouve taux de réussit déferent.

Mais globalement, nous avons 43 mot clé utilisé pour l'annotation manuelle des images de test pour chaque image, nous avons obtenu pour chaque méthode un taux de réussit global comme suit :

- K-ppv : à obtenir 26 mot clé correct avec un taux = 60.46 %.
- K-means : à obtenir 20 mot clé correct avec un taux = 46.51 %.
- FCM à obtenir 23 mot clé correct avec un taux = 53.48 %.

6. Temps d'exécution de l'algorithme de segmentation k-means :

Pour testée la fiabilité et la rapidité d'algorithme de segmentation k-means nous avons exécutée le système sur un PC avec 4Go de RAM et CPU Intel Core 2 Duo 2.0 Ghz et un carte graphie ATI 1.5 Go, nous utilisons un ensemble des images de corpus de même taille (90 x 120) on change a chaque fois le nombre des images et on compte le temps nécessaire pour l'exécution. Le tableau 5.2 le nombre des images et le temps nécessaire pour la segmenté.

<i>Nombre d'images</i>	<i>Temps d'exécution (Sec)</i>
1	0.42
5	2.2
10	4.2
20	8.5
50	22.46
100	51.4
200	97.45
500	281.067
1000	688.71
2000	1177.51

Tableau 5.2: Résultat de temps d'exécution K-means.

7. Conclusion :

Nous ne pouvons pas déduire quelle est la meilleure méthode de classification qui peut donner les meilleurs résultats pour l'annotation automatique des images, mais pour ce cas on peut dire que k-ppv donne les meilleurs résultats, mais cela ne permet pas de les favoriser par rapport aux autres, car une autre collection de test peut donner des résultats différents.

Les résultats que nous avons obtenus sur un échantillon d'images sont encourageants. Pour faire encore de test.

Conclusion Générale

Les bases d'images numériques connaissent un essor considérable. Leur facilité d'acquisition et de stockage les rendent très attractives pour des applications diverses. Ils en résultent une production permanente et croissante d'images numériques dans différents domaines.

Par conséquent, et pour assurer une exploitation efficace de ces bases d'images par des machines, la description de telles bases est une tâche qui s'impose. Sinon, ces images peuvent demeurer pratiquement inexploitable et impossible de les retrouver et utiliser efficacement.

L'annotation automatique des images est un domaine de recherche très actif. Elle consiste à désambiguïser et interpréter les images d'une façon qui assure leur exploitation adéquate par la machine. Cette technique consiste à assigner la sémantique à une image en associant des mots clés qui décrivent cette sémantique.

La qualité d'annotation d'images dépend de plusieurs critères. Dans notre projet de fin d'étude, nous sommes intéressés à la classification.

Notre objectif principal est d'améliorer la qualité d'annotation d'images, en cherchant la bonne technique de classification qui donne une meilleure annotation d'image. Pour ce faire, nous avons implémenté les méthodes Kppv, Kmeans et FCM dans un système d'annotation automatique.

Notre travail s'articule autour de deux phases principales suivantes :

- Apprentissage : Cette phase est basée sur un corpus d'images annotées. Où les images de corpus doivent être segmentées en régions afin de les grouper dans des classes des régions visuellement similaires appelées blobs. Finalement, les blob construits précédemment doivent être annotés.
- Traitement d'une nouvelle image : est une tâche qui consiste à annoter automatiquement d'une nouvelle image de test.

La mise en œuvre de notre système conduit à des bons résultats après des tests, comparaison et validation mais le K ppv à donner des résultats que les deux autres méthodes.

Perspective

Concernant les méthodes de classification considérées pour la phase de catégorisation les images, nous souhaiterions utiliser d'autres méthode (par exemple SVM, SOM) et évaluer leur impact dans le système de l'annotation automatique d'images.

Concernant les expérimentations réalisées, nous souhaitons expérimenter nos approches sur d'autres collections plus volumineuses, a pour but d élargir la couverture des collections de tests considérées.

Bibliographie



- [2] N. Ben Cheikh et R. Ben Bezziane, la recherche d'images par la sémantique, mémoire de master, Université Kasdi Merbah Ouargla, 2011.
- [3] M. Elyes, modélisation statistique des collections à d'images en utilisant les modèles de copules et application à l'indexation d'images, thèse de doctorat, université de Québec, 2011.
- [4] L. Jayaratne, Z. Jiang et A. Ginige, A unified approach to indexing multimedia from text accompanying pictures. Pages 793-798, National conférence on Artificial intelligence (AAAI), 1994.
- [5] M. Iggane, D. Mammass, A. Khenchaf et J. Christophe, Segmentation d'images sonar par matrice de co-occurrence, International Conference: Sciences of Electronic, Université Ferhat Abbas Sétif, 2009.
- [7] B. le roux, Recherche dans les bases de données images une approche de description et une mesure de similarité combinant les aspects physique set sémantiques de l'image, Université Kasdi Merbah Ouargla, 2004.
- [8] H. Hacid, Annotation Semi-automatique de Grandes BD Images : Approche par Graphes de Voisinage, Université Lumière Lyon 2, Conférence en Recherche d'Informations et Applications, 2006.
- [9] M. MITRAN, Annotation d'images via leur contexte spatio-temporel et les métadonnées du Web, thèse de doctorat Paris, 2014.
- [10] F. Amourache, Construction d'une ontologie pour l'annotation des CVS/Offres d'emploi, Décembre 2008.

- [11] Kucuktunc O et Can F, Automatic photo tag expansion using visual information for photo sharing web sites, Proceedings of the 3rd International Conference on Semantic and Digital Media Technologies, Berlin, 2008.
- [12] A.W.Smeuldres, M. Worring, S. Santini, A. Gupta and R. Jain. Content based image retrieval at the end of the early years, IEEE Transaction on Pattern Analysis and Machine Intelligences, December 2000.
- [13] Wenyin L, Zhang H et Czerwinski M. Semiautomatic image annotation. In Proceedings of Interact, Conference on Human-Computer Interaction (HCI), 2001.
- [14] M. Bakalem, Indexation textuelle sémantique pour un moteur de recherche d'images sur le web, Mémoire de magistère, Université Saad Dahleb Blida, 2010.
- [15] Ph. Bolon et al, Analysed'image : Filtrage et segmentation, Paris, 1998.
- [16] Ph. Bolon, J-M. Chassery, J-P. Cocquerez, D. Demigny, C. Graffigne, A.Montanvert, S. Philipp, R. Zeboudj and J. Zerubia. Analyse d'image : Filtrage et segmentation. Paris. 1998
- [17] G. Gouvaert, Analyse de données IC2, Hermes Lavoisier, page 151.168, Paris 2003
- [18] K. Sauvagnat. Modèle flexible pour la recherche d'information dans des corpus de documents semi-structurée. Thèse de doctorat, université Paul Sabatier. Toulouse. France. 2005.
- [19] Berrani *et al.*. Recherche par similarités dans les bases de données multidimensionnelles. Panorama des techniques d'indexation. Ingénierie des systèmes d'information. 65-90. France. 2002.
- [20] Sabrina Tollari, Indexation et recherche d'image par fusion d'information textuelle et visuelle, Thèse de doctorat, université du sud Toulon-var mention informatique, France, 2006
- [21] Sabrina Tollari, Indexation et recherche d'image par fusion d'information textuelle et visuelle, Thèse de doctorat, université du sud Toulon-var mention informatique, France. page 187, 2006
- [22] Boser.b, Guyon. I et Vapnik, V. A training algorithm for optimal margin classifiers, page 144-152 of : workshop on computational learning theory, 1992.
- [23] Mohamadally Hasan, Fomani Boris, BD Web, ISTY3 Versailles St Quentin, France 2006

- [24] Sheridan, S. C, The redevelopment of a weather-type classification scheme for North America, *International Journal of Climatology*, page 51-68, 22(1), 2002.
- [25] Chen, G., Jaradat, S. A., Banerjee, N., Tanaka, T. S., Ko, M. S., & Zhang, M. Q, Evaluation and comparison of clustering algorithms in analyzing ES cell gene expression data. *Statistica Sinica*, page 241-262, 12(1), 2002.
- [26] Strehl A., Relationship based Clustering and Cluster Ensembles for High dimensional Data Mining, Thèse de doctorat, université de Texas, May 2002.
- [17] Kanungo, T., Mount, D. M., Netanyahu, N. S., Piatko, C. D., Silverman, R., & Wu, A. Y. An efficient K-means clustering algorithm: Analysis and implementation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, Page 881-892, 2002.
- [28] Boubou. M, Contribution aux méthodes de classification non supervisée via des approches pré-topologiques et d'agrégation d'opinions, Thèse de doctorat, Université Claude Bernard-Lyon I, 2007.
- [29] Kaufman. L, & Rousseeuw. P, Clustering Large Applications (Program CLARA). *Finding Groups in Data: An Introduction to Cluster Analysis*, 1990.
- [30] Singh G., Kumar V, An Efficient Clustering and Distance Based Approach for Outlier Detection, *International Journal of Computer Trends and Technology (IJCTT) Vol.4*, page 2067-2072, 2013.
- [31] Ng, R. T. et Han, J. *Efficient and effective clustering methods for spatial data mining*, 1994
- [32] Singh G. Kumar V, An Efficient Clustering and Distance Based Approach for Outlier Detection, *International Journal of Computer Trends and Technology (IJCTT) Vol.4*, page 2067-2072, 2013.
- [33] kohonen. T, *self-organising map*, 3^{ed} ed. Springer-verlag, Springer series inf information Sciences.
- [34] Carl. F, *classification supervisée avec rejet(l) approche Probabiliste*. Laboratoire d'informatique-image-interaction, EA 2118 Université de la Rochelle, 2004.
- [35] L.Malha, *segmentation d'images texturées à partir des attributs fractals*, UMM Tizi-Ouzou.
- [20] W. Jin, R. Shi, et T-S. chua, A Semi-Naïve Bayesian Method Incorporating Clustering with Pair-wise Constraints for Auto Image Annotation, *ACM 1-58113-893-8/04/0010,MM'04*,New York, USA, 2004.

- [21] Y. Gao, J. Fan, H. Luo, ..., Automatic image annotation by incorporating feature hierarchy and Boosting to scale up SVM Classifiers, ACM 1-581133-737-0/03/0008, October 23-27, California, USA, 2006.
- [22] R. Albat, P. Mulhem, Y. Chiamella, Phrases Visuelles pour l'annotation automatique d'images, Laboratoire d'informatique de Grenoble (LIG).
- [24] J. Lu, T. Zhao et Y. Zhang. Feature selection Based-On Genetic Algorithm for Image Annotation, Knowledge-Based Systems, 0950-7051, 2008.
- [27] CM. Huang, CC. Chang et CT. Chen, Automatic Image Annotation by Incorporating Weighting Strategy with CSOM Classifier, Department of Information Management National, Yunlin University of Science & Technology, Taiwan, 2011.
- [31] A. Nakib, conception de méta-heuristiques d'optimisation pour la segmentation d'images, thèse de doctorat, université de Paris, 2007.

Webographie :

- [1] http://www.memoireonline.com/12093040m_La-liaison-automatique-des-plusieurs-images-percues-sur-un-scanner4.html, visité en Mai 2015.
- [28] http://www.journaldunet.com/solutions/0304/030415_faq_modelisation.shtml, Visité septembre 2015
- [29] <http://imageclef.org/photodata>, visité Septembre 2015
- [30] https://fr.wikipedia.org/wiki/Filtre_m%C3%A9dian, visité Septembre 2015.