

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE

SCIENTIFIQUE

UNIVERSITE AMAR THELIDJI -LAGHOUAT-



Faculté des sciences

Departement de mathématique et d'informatique

2ème année master informatique

MEMOIRE DE FIN D'ETUDES

THEME :

**CLASSIFICATION DE REQUETE POUR LA
FRAGMENTATION VERTICALE D'UNE BDD**

Présenté par :

- GUERIDA NASSIMA
- KORICHI KHADIDJA

Encadré par :

Mr: ZIANI AMEUR

Année Universitaire :2017/2018

Table des matières :

introduction générale.....	2
1.Chapitre1 :la segmentation des images... ..	3
1.2. la segmentation d'image	3
1.2.1 la segmentation par region.....	3
1. 2.1 la methode des k-means	3
1.2.2 la méthode des k-medoids.....	4
1.3. segmentation hierarchique.....	4
1.3.1 classification descendante hiérarchique :DIANA.....	5
1.3.2 classification ascendante hiérarchique :AGNES.....	7
4. segmentation par densité.....	8
4.1. l'algorithme DBSCAN.....	8
5.conclusion.....	9

2. chapitre 2:L'imagerie	10
2.1.'image numérique	11
2.1.1 caractéristiques d'une image numerique pixel.....	11
2.2 image couleurs (RVB).....	11
2.2.1système colorimétriques (RVB).....	12
2.3type d'images.....	12
2.3.1image bitmap.....	12
2.3.2image vectorielle	13
2.4les formats de fichiers images	13
2.4.1bmp	13
2.4.2 jpeg	13
2.4.3gif	13
2.4.4png	14
2.4.5DXF.....	14
2.4.6tiff	14
2.4.7EPS	14
2.4.8PSD	14
2.5 le spectre d'une image	14
2.6. passage aux niveaux de gris	15
2.7.binarisation.....	15

2.8. la segmentation d'images (paragraphe)	16
2.9.conclusion	16

3.Chapitre3 : résultats et discussion	17
3/ introduction générale sur l'algorithme d'AGNES.....	18
3. 1.c'est quoi agnes.....	18
3. 1.1 adapter agnes pour les images	18
3 1.2 calcul des distances	19
3.2. choix du langage (matlab).....	19
3.2.1 facilité.....	19
3 2.2 communauté existante	20
3 2.3 présence d'un tool kit.....	20
3.2.4 matlab pour le traitement	20
3.3 prétraitement	21
3.3.1 introduction sur matlab	21
3.3.2 rappels sous matlab	21
3.3.3 lire et écrire des images sous matlab	21
3.3.3.1 quelques fonctions sous matlab	21
3.4 l'algorithme d'AGNES	22
3.4.1 application de l'algorithme AGNES sous matlab	23
4. Conclusion générale	27

dédicace

A ma mère Celle qui attendue avec patience les fruits de sa bonne éducation et qui a toujours été là pour moi j'espère qu'elle trouvera sa récompense de tous les sacrifices qu'elle a consentis.

A mon père En témoignage de ma sincère gratitude pour ses conseils et son soutien dans les moments difficiles .

A mes frères Avec toute mon affection et ma considération profonde pour leur présence de tous les jours.

A mes formateurs Tous mes professeurs et la direction de l'université.

A tous ceux qui ont contribué de près ou de loin à l'accomplissement de ce modeste travail ,mes amis Et surtout les étudiants de la promotion Avec tous mes sentiments de reconnaissance et gratitude.



Dédicace

Je dédie ce travail

A ma chère mère

En lui exprimant mon amour, mon respect et vive gratitude

Pour sa patience, son amour et ces prières.

A mon père

Auquel je souhaite prompt rétablissement et longue vie.

A mes frères et sœurs et surtout MUSAFAH mon frère aîné

Je leur souhaite une vie pleine de réussite et de bonheur

Je les remercie pour leurs sacrifices avec toute ma gratitude.

A toute ma grande famille.

*A toute mes chers amis de la promotion surtout ceux de l'option
informatique « systèmes et réseaux »*

A tous ceux qui m'aiment et que je n'ai pas cités

Kalthoum



Introduction générale

Le traitement d'images sert à désigner une discipline de l'informatique et des mathématiques appliquées qui étudie les images numériques et leurs transformations, dans l'objectif de perfectionner leur qualité ou d'en extraire de l'information.

Il s'agit d'un sous-ensemble du traitement du signal dédié aux images ainsi qu'aux données dérivées comme la vidéo (par opposition aux parties du traitement du signal consacrées à d'autres types de données : son et autres signaux monodimensionnels surtout), tout en opérant dans le domaine numérique (par opposition aux techniques analogiques de traitement du signal, comme la photographie ou la télévision respectant les traditions).

Dans le contexte de la vision artificielle, le traitement d'images se place après les étapes d'acquisition et de numérisation, assurant les transformations d'images et la partie de calcul permettant d'aller vers une interprétation des images traitées. Cette phase d'interprétation est d'ailleurs de plus en plus intégrée dans le traitement d'images, en faisant appel surtout à l'intelligence artificielle pour manipuler des connaissances, essentiellement sur les informations dont on dispose à propos de ce que représentent les images traitées (connaissance du *domaine*). La compréhension du traitement d'images débute par la compréhension de ce qu'est une image. Le mode et les conditions d'acquisition et de numérisation des images traitées conditionnent beaucoup les opérations qu'il faudra réaliser pour extraire de l'information.

CHAPITRE I

La segmentation D'images

CHAPITRE II :

L'Imagerie

Chapitre 1

1. La segmentation d'images

La segmentation d'images est une étape fondamentale et importante dans beaucoup d'applications de vision par ordinateur. C'est une étape primordiale pour l'interprétation d'images. Beaucoup de méthodes de segmentation ont été développées, mais il n'y a pas encore de mesures de performance satisfaisante. Concevoir une bonne mesure pour la qualité de segmentation est un problème dur. Quelques chercheurs pensent que c'est impossible. ¹

2. Segmentation par région :

2.1 Méthodes des k-means (centres mobiles)

Les objets sont réparties en k classes de façon à minimiser la fonction objective

$$C = \sum_{j=1}^k \sum_{i \in I_j} p_i d_2(i, g_j)^2$$

Algorithme k means

1. Choisir k centres mobiles (en général au hasard)
2. Affecter chaque objet au centre le plus proche de façon à minimiser C.
3. Calculer les centres de gravités des classes définissant de nouveaux centres mobiles.
4. On reprend les étapes 1 à 3 jusqu'à obtenir une partition stable.

Avec la méthode de Forgy, les centres mobiles initiaux sont tirés au sort parmi les n objets (voir le cours). Avec la méthode de MacQueen, les centres mobiles sont recalculés à chaque réaffectation d'un objet. Avec **R**, on utilise la fonction :

kmeans(x, centers, iter.max = 10, nstart = 1,

algorithm = c("Hartigan-Wong", "Lloyd", "Forgy", "MacQueen"))

x est un tableau numérique : objet × variables numériques

kmeans(x,k)\$cluster est un vecteur définissant la classe de chaque objet

¹Mémoire de magister (traitement d'images et reconnaissances de formes)

²Cours (Segmentation d'images : principes)

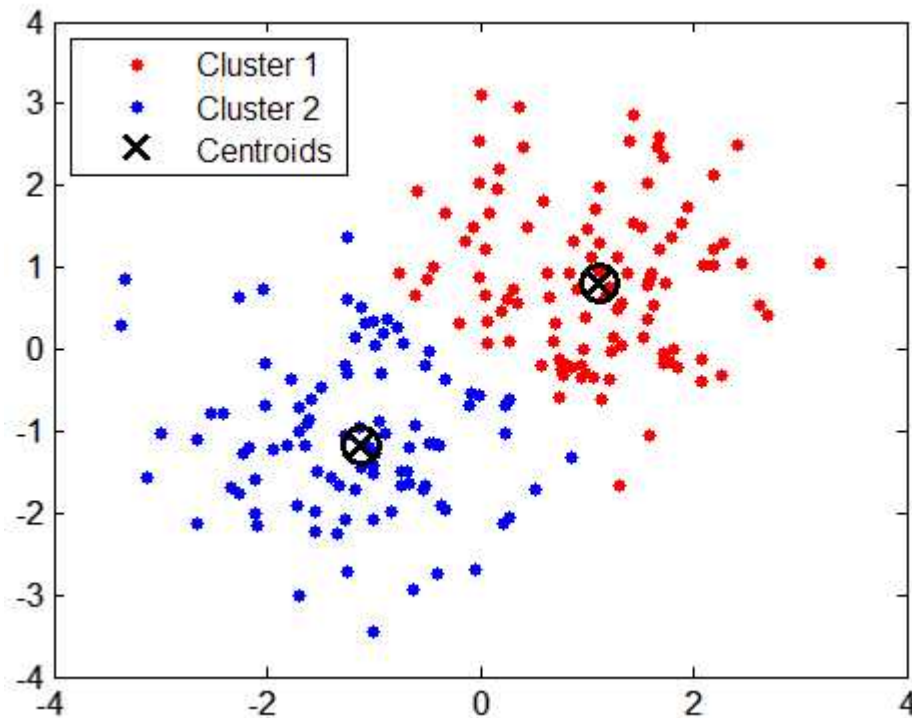


Figure 1-Méthodes des k-means

C=

Dans la figure -1- on a choisi k centres mobiles et on a affecté chaque objet au centre le plus proche de façon à minimiser C . en fin on va calculer les centres de gravites des classes définissant de nouveau x centres mobiles

2.2 Méthode PAM (partition around medoids)

Comme les k -moyennes, l'algorithme des k -médoids minimise l'erreur quadratique moyenne qui est la distance entre les points de la classe et le point central (ou médoid).

2.2.3 Algorithme

Etape 1 :

- Le premier médoid est l'objet minimisant la fonction objective
- Le second médoid est l'objet, autre que m_1 , minimisant la même fonction objective
- On détermine ainsi les k premiers médoids.
- Chaque objet est ensuite affecté au groupe dont le médoid est le plus proche.

Etape 2 :

- Pour tout couple d'objet (i,j) tel que i soit un des k médoids et j un objet autre qu'un médoid. On permute i et j si la fonction objective baisse.
- On répète l'étape précédente jusqu'à stabilisation de la partition.

Dans la figure -2- suivante on a la méthode des k-médoides qui minimise l'erreur quadratique moyenne qui est la distance entre les points de la classe et le point central (ou médoides) .

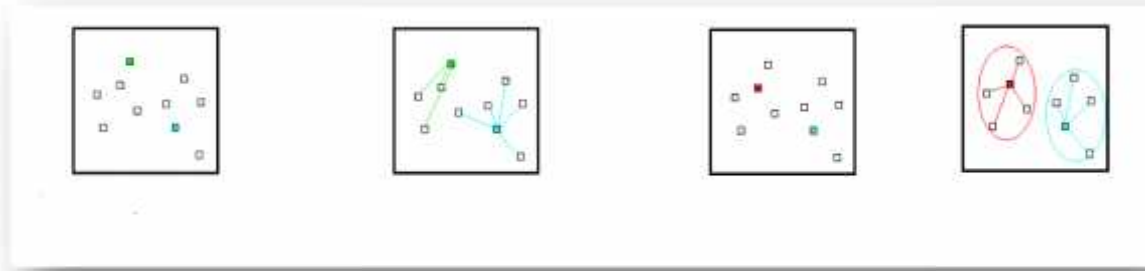


Figure 2-partition around medoids

3. Segmentation hiérarchique :

3.1 .1Classification descendante hiérarchique : DIANA

DIANA est conçu par Kaufman et kousseu .c'est une méthode par division récursive , à l'inverse de celui d'Agnes ,tous les objets sont placés dans un cluster , à chaque itération divise de manière hiérarchique les clusters, on se bassant sur notion de distance.

3.1.2 Algorithme :DIANA

Entrées : une matrice de proximités D

Sortie : une hiérarchique P

Début

- 1- Tous les objets appartiennent à un même cluster**
- 2- Chaque cluster est divisé .selon un critère de dispresion**
- 3- La division se fait par récursivité**
- 4- Stopper quand le nombre de clusters visé est attend ou si chaque cluster ne contient qu'un seul objet.**

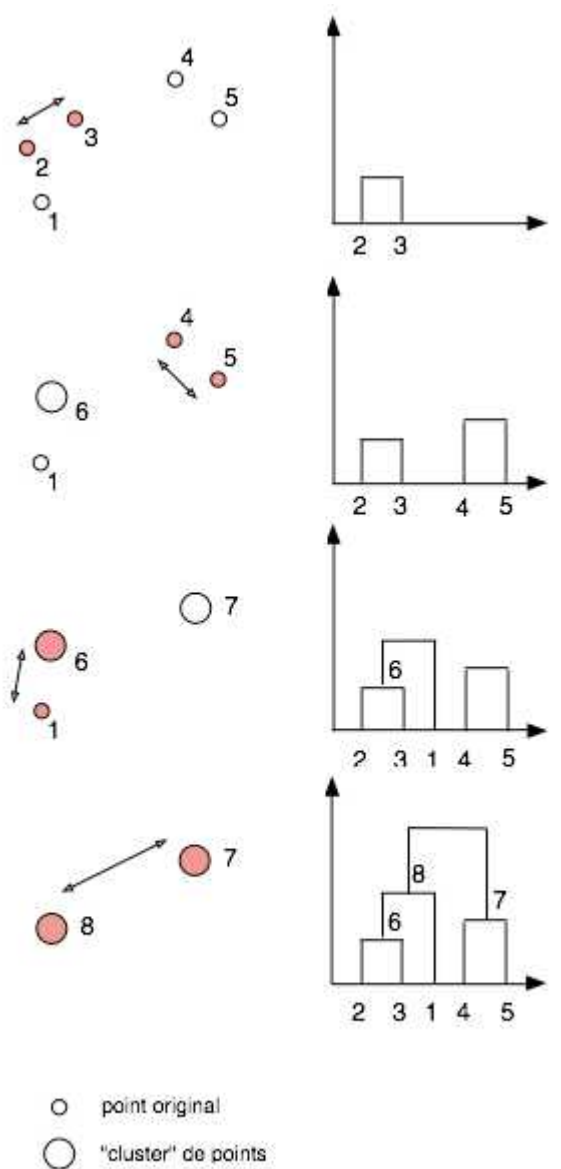


Figure 3 -Classification descendante hiérarchique : DIANA

A la figure -3- : Tous Les objets sont regroupés au sein d'une même classe . en suite Dans la classe C présentant la plus grande dissimilarité entre deux objets, on sépare ces objets en deux classes A et B ,Les objets de la classe C scindée en deux sont affectés à l'un ou l'autre des deux classes A ou B.

3.2 A Classification ascendante hiérarchique : AGNES

-) Entrées : une matrice de proximité D
-) Sortie : une hiérarchie P

Début

- Calculer la matrice des proximités D des données ;
 - Construire la partition la plus fine : chaque point est un cluster ;
 - Répéter
 - Trouver les plus proches clusters selon un critère de fusion ;
 - Les fusionner dans un seul cluster ;
 - Recalculer la matrice réduite D des proximités des clusters ;
- Jusqu'à ce que (les clusters soient fusionnées dans un cluster unique) ou (une condition d'arrêt devienne vraie)

Fin.

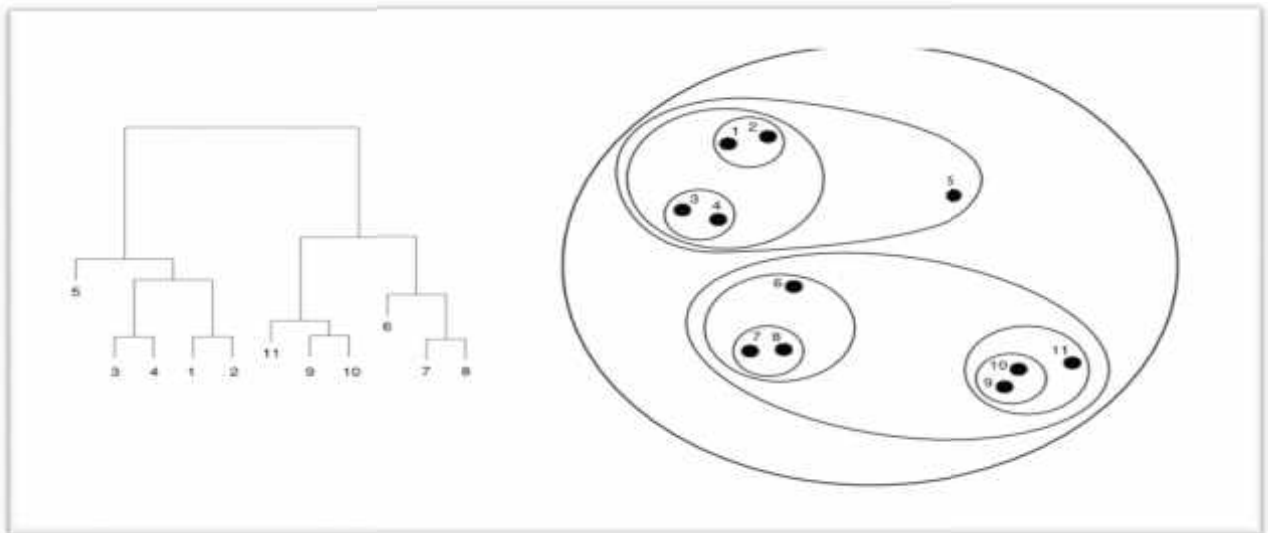


Figure 4-Classification ascendante hiérarchique : AGNES

dans la figure - 4 – on va regrouper les 2 objets ou classes les plus proches en une nouvelle classe . La distance des autres objets ou classes par rapport à cette nouvelle classe sont calculés à l'aide d'un critère d'agrégation.

4 . segmentation par densité :

4.1 L'algorithme DBSCAN

Utilisé 2 paramètres : la distance ϵ et le nombre minimum de points *MinPts* devant se trouver dans un rayon ϵ pour que ces points soient considérés comme un cluster. Les paramètres d'entrées sont donc une estimation de la densité de points des clusters. L'idée de base de l'algorithme est ensuite, pour un point donné, de récupérer son ϵ -voisinage et de vérifier qu'il contient bien MinPts points ou plus. Ce point est alors considéré comme faisant partie d'un cluster. On parcourt ensuite l' ϵ -voisinage de proche en proche afin de trouver l'ensemble des points du cluster.³

Algorithme

```

DBSCAN(D, eps, MinPts)
C = 0
pour chaque point P non visité des données D
marquer P comme visité
PtsVoisins = epsilonVoisinage(P, eps)
tailleDe(PtsVoisins) < MinPts
mark P as NOISE
sinon
    C++
    expandCluster(D, P, PtsVoisins, C, eps, MinPts)

expandCluster(D, P, PtsVoisins, C, eps, MinPts)
ajouter P au cluster C
pour chaque point P' de PtsVoisins
si P' n'a pas été visité
marquer P' comme visité
PtsVoisins' = epsilonVoisinage(D, P', eps)
    si tailleDe(PtsVoisins') >= MinPts
PtsVoisins = PtsVoisins U PtsVoisins'
si P' n'est membre d'aucun cluster
ajouter P' au cluster C

epsilonVoisinage(D, P, eps)

```

³cours master2(traitement et reconnaissances d'image)

retourner tous les points de D qui sont à une distance inférieure à epsilon de P^4

Conclusion

Dans ce chapitre On va choisir parmi les 3 algorithmes (k-means, k-medois, Diana) l'algorithmes Agnes au traitement mais avant on va présenter quelque notions de traitement d'image .

⁴Université de Nice - Sophia Antipolis (Traitement du Signal et des Images)

CHAPITRE III :

Résultats et discussion

Chapitre 2

1-Image numérique :

1-1 Caracteristiques d'une image numerique Pixel :

Lorsqu'on agrandit une image numérique, on voit que celle-ci est composée d'un ensemble de « points », appelés pixels le pixel (abréviation venant de l'anglais : pictureelement est l'élément de base d'une image ou d'un écran, c'est-à-dire un point. L'ensemble de ces pixels est contenu dans un tableau à deux dimension constituant l'image (k).la lettre A, par exemple, peut être affichée comme un groupe de pixels comme l'indique la figure suivante

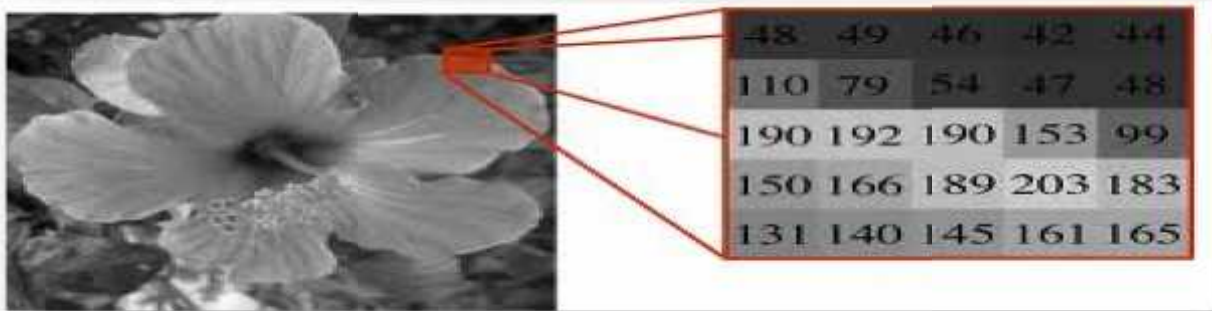


Figure 1 : image numérique

2-Les images couleurs (R,V ,B) :

Il existe plusieurs modes de représentation de la couleur, le plus utilisé pour les images numériques est le mode ROUGE, VERT, BLEU (R, V, B) basé sur la synthèse additive des couleurs (le mélange de trois composantes R, V, B donne une couleur) la combinaison de ces couleurs avec des quantités égales donne la couleur blanche). ce modèle est utilisé pour l'affichage des couleurs à l'écran.

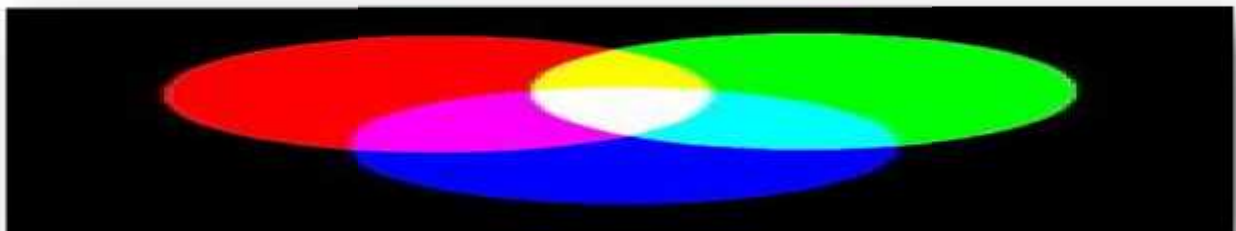


Figure2 : système RVB

Figure2 (on dit que le rouge ,le vert et le bleu , sont les couleurs primaires en synthèse additive, c'est-à-dire l'addition des trois couleurs tend vers le blanc (observer le centre de l'image)

2-1Systèmes colorimétriques RVB :

Il existe une infinité de systèmes RVB (la dénomination internationale étant RGB pour Red, Green, Blue). Ceux-ci dépendent du choix de la teinte, caractérisée par la longueur d'onde, et de la saturation (pureté) de chaque composante. Dans le diagramme de chrominance (x,y) du système CIE xyY, les trois points R, V, B correspondants définissent un triangle. Seules les couleurs situées dans ce triangle peuvent être reconstituées à partir des trois couleurs primaires choisies. Un système RVB ne peut donc pas décrire toutes les couleurs visibles par l'œil humain. Plus précisément il peut faire apparaître toutes les teintes mais pas toutes les saturations.

En 1931 la Commission internationale de l'éclairage a étudié de manière approfondie le système particulier CIE RGB avant d'en déduire CIE XYZ qui couvre toutes les couleurs visibles par l'homme. Ultérieurement, les systèmes sRGB et Adobe RGB ont été définis pour faciliter une gestion de la couleur cohérente dans un ensemble informatique.

3.Type d'images :

Les images produites et traitées par les ordinateurs sont de deux types: les images *bitmap* et les images "vectorielles" :

3.1. Les images bitmap (ou "pixellisées") :

Une image bitmap est composée en mode point. Le système de codage le plus universel consiste en effet à décomposer la représentation graphique, l'image, en un certain nombre de points élémentaires caractérisés par leurs coordonnées spatiales et leur couleur. Il s'agit donc d'une représentation graphique définie par l'ensemble des points qui la composent.

Dans le cas d'une photographie, l'opération de digitalisation, d'acquisition ou encore de numérisation, consiste à transformer des tons et des couleurs continus - en nombre infini- en un nombre fini de points élémentaires.

3.2. Images vectorielles :

Ces images sont formées d'objets. Des lignes, des carrés, des polygones, des cercles. Ces objets sont définis par leurs coordonnées vectorielles ou par leurs équations mathématiques. Une ligne est par exemple définie par les coordonnées X et Y de ses deux extrémités. On peut facilement déplacer ces objets ou les faire pivoter ou les agrandir sans perte de qualité (calcul matriciel).

On peut aussi leur attribuer des propriétés : épaisseur, couleur, texture, ombre portée, ombre propre, reflets... Bref, elles permettent de faire des images de synthèse.

4. les formats de fichiers images :

Un format d'image est une représentation informatique de l'image, associée à des informations sur la façon dont l'image est codée et fournissant éventuellement des indications sur la manière de la décoder et de la manipuler. La plupart des formats sont composés d'un en-tête contenant des attributs (dimensions de l'image, type de codage, LUT, etc.), suivi des données (l'image proprement dite). La structuration des attributs et des données diffère pour chaque format d'image. De plus, les formats actuels intègrent souvent une zone de métadonnées (metadata en anglais) servant à préciser les informations concernant l'image comme : la date, l'heure et le lieu de la prise de vue, les caractéristiques physiques de la photographie (sensibilité ISO, vitesse d'obturation, usage du flash...)

4.1BMP (bitmap) :une matrice de bits, codé en couleur jusqu'à 24 bits par pixel lu uniquement sur des logiciel Windows (image non compressée).

4.2 Le format Jpeg :(Joint Photo Expert Group) :C'est le format le plus courant (extension **.jpg**). On le rencontre sur Internet. Il occupe peu d'espace disques. C'est le format développé par les photographes pour transmettre des images de qualité photographique professionnelle. Il est libre de droits, gère les millions de couleurs mais il ne possède pas de palette couleur associée et donc les couleurs peuvent être différentes sur des machines et des systèmes différents.

4.3 Le format Gif : (GraphicsInterchange Format) :On le rencontre aussi sur l'Internet (extension **.gif**). Il possède une palette de couleurs associée et occupe peu d'espace disque mais est limité à 256 couleurs. Il appartient à CompuServe qui perçoit des droits d'utilisation.

4.4 Le format Png : (Portable Network Graphics): Les graphistes s'orientent vers ce nouveau format (extension Pc **.png**) qui est libre de droits, gère les millions de couleurs et possède une palette couleurs associé compression sans perte de données, gère 256 niveaux pour chaque couleurs et la transparence.

4.5 DXF :Le format DXF est un format vectoriel créé par la compagnie AutoDesk pour son logiciel de CAO AUTOCAD. Bien qu'étant un format très répandu dans le monde de la conception et du dessin assisté par ordinateur, le format DXF est très peu répandu en d'autres domaines.

4.6 TIFF :(Tagged Image File Format):représentation d'images ; de 1 à24 bits. Images binaires (compression parfaite).

4.7 EPS : (Encapsulated PostScript):format PostScript avec codage couleur sur 8, 16, 24 ou 32 bits.

4.8 PSD:format correspondant au logiciel Adobe Photoshop format très répandu, peut coder la couleur sur 8, 16, 24 ou 32 bits.

5- le spectre d'une image:

On entend par spectre d'une image l'histogramme des niveaux de couleur c'est-à-dire la représentation du nombre de pixels ayant un niveau de couleur en fonction de celui-ci. Pour une image en couleur ce spectre peut être fait sur chaque couleur fondamentale (r-v-b) et/ou sur la luminosité globale (niveau de gris équivalent)

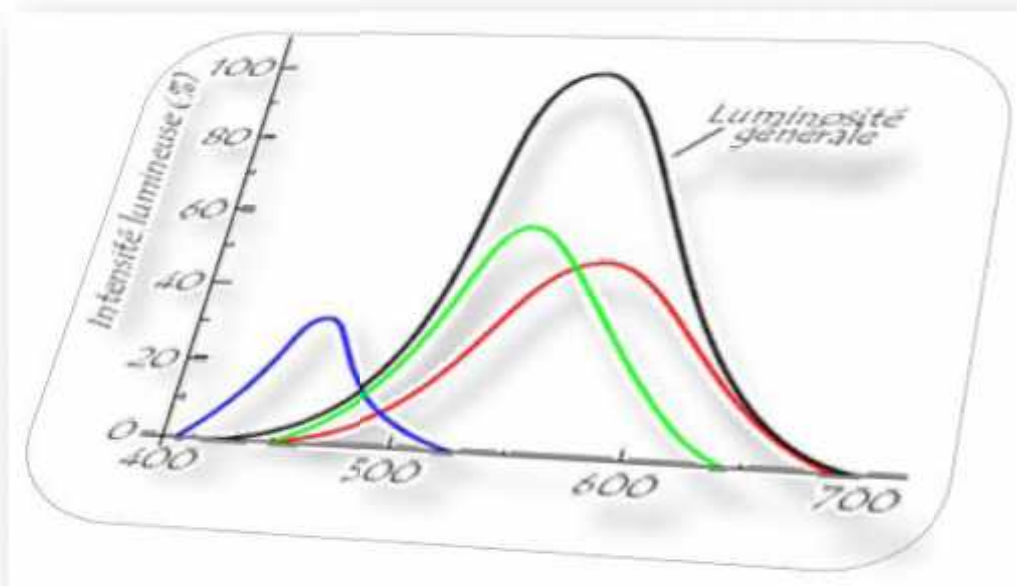


Figure 3 :Spectre RVB.

Il existe plusieurs caractéristiques que l'on peut déduire du spectre (la couleur dominante dans une image, les couleurs constituant l'image, la nature de l'image (claire ou sombre...)).

6. Passage aux niveaux de gris :

Une image numérique en couleur est codée généralement en RVB (pour Rouge, Vert et Bleu) qui utilise 24 bits dont chaque couleur est représentée sur un octet, soit 2^{24} couleurs (plus de 16 millions de couleurs), alors qu'une image en niveaux de gris comporte seulement 256 niveaux de gris (un seul octet) [27]. Par conséquent, il est plus aisé d'utiliser des images en niveaux de gris que des images couleurs. Cette phase est très utile dans le traitement des dégradations car une image de ce type contient toujours des couleurs très proches, et par conséquent, la transformation en niveaux de gris portera toujours l'information. (voir figure 4).

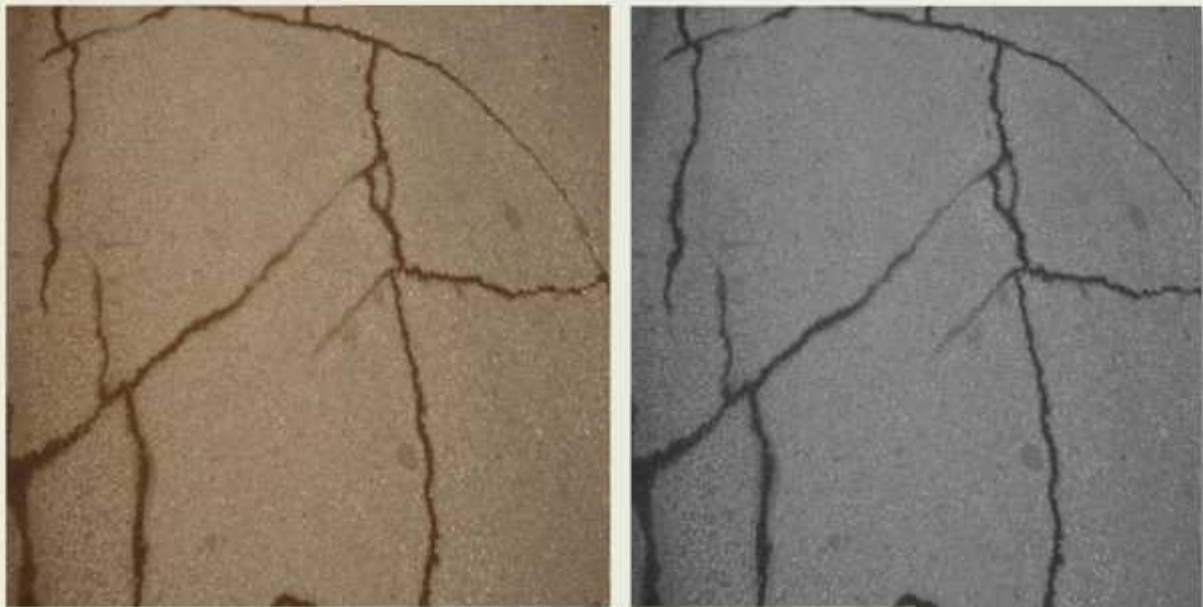


Figure 4 : Transformation d'une image de fissure en niveaux de gris

7 .binarisation :

En traitement d'image, la binarisation est une opération qui produit deux classes de pixels, en général, ils sont représentés par des pixels noirs et des pixels blancs.

La différence entre la binarisation et la segmentation est que la binarisation produit toujours deux classes alors que la segmentation peut en produire plusieurs. Cependant, ces deux termes sont souvent confondus par abus de langage



Figure 5 :binarisation

8.La segmentation d'images :

Est l'un des problèmes phares du traitement d'images. Elle consiste à partitionner l'images en un ensemble de régions connexes. L'intérêt de ces régions est de pouvoir être manipulées ensuite via des traitements de haut niveau pour extraire des caractéristiques de forme, de position, de taille, etc. Le problème est évidemment très mal posé, car on ne sait jamais dire quelle est la segmentation idéale. L'idée est bien sûr que la région se rapproche de la notion d'objet, au sens courant du terme. Néanmoins, on peut dégager des propriétés plus raisonnables qu'on cherche à obtenir dans un algorithme de segmentation, en particulier :

- Stabilité: la segmentation obtenue ne doit pas varier beaucoup lorsque les conditions d'acquisition varie légèrement (bruit, illumination, point de vue...)
- Régularité: les régions obtenues doivent être simples' à manipuler (taille Suffisante, forme régulière,...)

conclusion :

la segmentation d'image compte plusieurs techniques à savoir la détection de contours la reconnaissance de formes ...ext.

Notre objectif dans ce chapitre n'est pas d'étudier les techniques existantes pour les images (choix de Markov ...) mais d'essayer d'adapter un des algorithmes de clustering largement utilisés dans divers domaines.

Dans le chapitre suivant, on va prendre comme exemple l'algorithme AGNES, et on va l'adapter pour la segmentation d'images.

1.c quoi AGNES ?

agnes :est conçu pour kaufman et rousseeuw en 1990 cet algorithme est basé sur la notion de distance pour rapprocher les groupes (au début de l'algorithme chaque pixel est un groupe) il fusionne les nœuds (groupes) qui ont la plus faible dis-similarité l'algorithme itère jusqu'à ce que tous les éléments soient en un seul groupe (le nœud racine).

1-1 adapter agnes pour le traitement d'image :

On a choisit l'algorithme d'agnes pour la segmentation d'image par ce qu'il est facile a manipuler ,efficace , et il utile plus d'une solution il nous permet à sgmenter entre les images hiérarchiquement par la classification ascendante hiérarchique , a des avantages :

Amélioration, élimination du bruit, rehaussement du contraste.

1-3 Définition de la distance entre les images :

La distance Δ est une distance entre images :

représentées par des arbres quaternaires. La distance Δ entre deux images i et j est définie par une somme de distance $\delta_k(i,j)$ entre les noeuds des arbres quaternaires représentant les images i et j , pondérées par des coefficients $C_k > 0$: $\Delta(i, j) = \sum C_k \delta_k(i, j) / \sum C_k$ (1)

$\delta_k(i,j)$ est une distance normalisée entre les nœuds homologues k des arbres quaternaire i et j définie précédemment.- k est l'identificateur d'un noeud pris parmi l'union des identificateurs de noeuds apparaissant dans les arbres quaternaires des images i et j .- C_k est un coefficient positif représentant le poids du noeud k dans le calcul de la distance C_k .

Chaque poids C_k est choisit selon l'importance qu'on souhaite donner à certains quadrants d'image par rapport à d'autres dans le calcul de la distance Δ (on peut même donner à l'utilisateur la possibilité de choisir lui-même les poids de certains quadrants). Par exemple, si certains quadrants ne doivent pas apparaître dans le calcul de Δ (le cas de la distance par région ou par niveau), alors ils peuvent être associés à un poids nul. Si la surface des quadrants doit entrer en jeu dans le calcul de Δ , alors chaque coefficient C_k doit être

proportionnel à la surface représentée par le quadrant par rapport à l'image entière.

2/choix du langage :

MATLAB (matrix laboratory) est un langage de programmation de quatrième génération et un environnement de développement ; il est utilisé à des fins de calcul numérique. Développé par la société The MathWorks, MATLAB permet de manipuler des matrices, d'afficher des courbes et des données, de mettre en œuvre des algorithmes, de créer des interfaces utilisateurs, et peut s'interfacer avec d'autres langages comme le C, C++, Java, et Fortran.

2-1 Facilité :

À l'occasion de la version 2013 a des produits matlab et simulink, traditionnel alignement des évolutions des outils qui composent cette famille ,mathwork met l'accent sur la facilité pour les utilisateurs à passer facilement de matlab (outil de programmation avec un langage de calcul) à simulink (outil de modélisation graphique), et donc de créer les flots hybrides optimisés. exemple il est possible désormais de passer directement et automatiquement d'un calcul en virgule fixe à l'aide d'un seul et même outil ,fixed point designer ,qui combine les fonctionnalités des ex-outils fixed point toolbox (pour matlab) et simulinkfixed point (pour simulink).

On notera aussi que la notion d'aide à la conception qui fournit au programmeur des conseils pour améliorer et optimiser son programme et qui existait au sein de matlab ,estétendue à simulink grâce à l'outil performance advisor.ce dernier oriente par exemple vers le choix d'un solveur , suggère de remplacer un bloc de calcul par un autre ,etc.

2-2 Communauté existante :

Les utilisateurs de MATLAB (environ un million en 2004) sont de milieux très différents comme l'ingénierie, les sciences et l'économie dans un contexte aussi bien industriel que pour la recherche. Matlab peut s'utiliser seul ou bien avec des toolbox (« boîte à

outils »)La société mathworks a créé un espace d'échange anglophone appelé le matlab central divisé en quatre parties .

2-3 Présence d'un toolkit :

Matlab a une série d'algorithmes déjà implémentés pour trouver les racines (root, fzero), les moindres carrés (lsqcurvefit, lsqin...), la solution de systèmes d'équations (fsolve, fzero) et la minimisation, en une et plusieurs dimensions. Pour minimiser une fonction à une variable dans un domaine on utilise fminbnd et si la fonction a plusieurs variables, on utilise fminsearch. Pour le cas de problèmes contraints on utilise linprog et quadprog pour les cas linéaires et quadratiques respectivement. La fonction fmincon permet de trouver le minimum d'un problème avec contraintes non linéaire et multi variable Matlab possède un toolkit d'optimisation (Optimization Toolbox) pour les problèmes plus compliqués, qui automatise via GUI (interface graphique) le processus de choix de l'algorithme. Matlab utilise plusieurs algorithmes selon le type de problème à résoudre

'interior reflective Newton method', 'trust-region-dogleg', 'trust-region-reflective', 'levenberg-marquardt', 'simpex', 'BFGS', 'minmax'....

2-4 Un algorithme d'optimisation :

est une procédure mathématique qui permet d'obtenir les minimums (ou maximums)1 d'une fonction réelle f (que l'on appelle fonction objective)

$$\min_{x \in \mathbb{R}} f(x)$$

2-5 pour le traitement d'image :

Matlab offre une possibilité intéressante, en effet, il est possible d'afficher plusieurs images dans la même figure. Pour ce faire, il faut utiliser la commande subplot. Elle s'utilise avec comme arguments le nombre de ligne, le nombre de colonnes et le numéro de l'image dans la figure.

3-Introduction :

L'objectif de cette introduction au traitement d'images sous Matlab est de présenter la notion d'image et d'effectuer des opérations simples d'analyse d'images telles que la détection de contours, le changement d'espace de couleur... Le traitement d'images est un thème de recherche situé entre l'informatique et le traitement du signal.

3-1 Rappels sous Matlab :

Une image Matlab est une matrice bidimensionnelle de valeurs entières ou réelles. Les principales fonctions de traitement d'images sous Matlab se trouvent dans la boîte à outils (toolbox) image processing (traitement d'images). L'aide sur cette boîte à outils est obtenue en tapant `help images` en ligne de commande de Matlab. Ensuite, l'aide sur une commande particulière est obtenue en tapant `help` suivi du nom de la commande, par exemple `help edge`

3-2 Lire et écrire des images sous Matlab :

Matlab est capable de lire et de décoder les fichiers images JPEG, TIFF, BMP, PNG, HDF, PCX ou XWD. Une image sous Matlab peut être représentée sous plusieurs formes, mais toujours sous forme d'une matrice. Avant de traiter une image dans Matlab, il faut la lire et décoder son format afin de la transformer en une matrice de valeurs.

Quelques fonctions sous matlab :

```
» img=imread('saturn.tif');
» img(3,2)
» img(1:10,30:40)
» img(1:3,31:39) = 0;
»img=imread('blood1.tif');
» img2=imread('alumgrns.tif');
» img=imread('rice.tif');
» img2=imread('alumgrns.tif');
» img=imread('rice.tif');
» img=imread('rice.tif');
» imgint=im2uint8(imgdbl);
» imwrite(imgint,'test.jpg','jpeg');
```

Algorithme d'Agnes :

Début

Min=255 ;

Pour K=1 :N

Pour i=1 : N

Pour j=i :N

D=dist (clust(i) et clust l <> -1 ,clust j <> -1,clust(j))

Si d < min

D=minn

O1 =i

O2=j

Fsi ;

finpour

finpour

clust (o1) := v(o1)

clust (o2) := v(o2)

clust (o1) := v(o1) *v(o1) +clust (o2) := v(o2)/clust (o1) := v(o2)

v(o1) = v(o1)+v(o2)

v(o2)= 0

ind (clust) = (i*n)+j

indclust /n

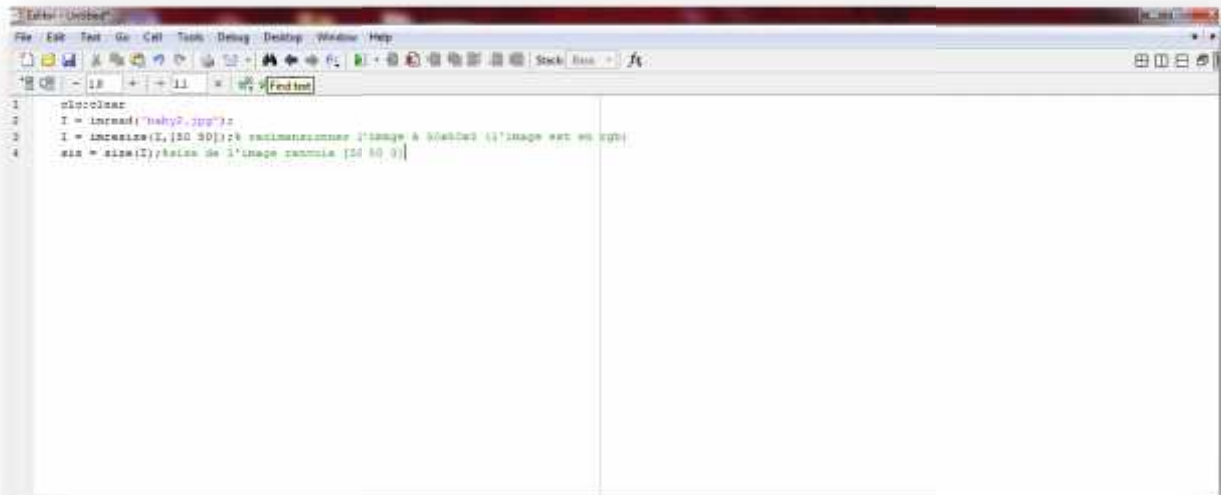
(i-1)

j

fin

```
Clc;clear
```

```
1 )redimensionner l'image à 50x50x3 (l'image est en rgb)
2 )size de l'image renvoie [50 50 3]
```

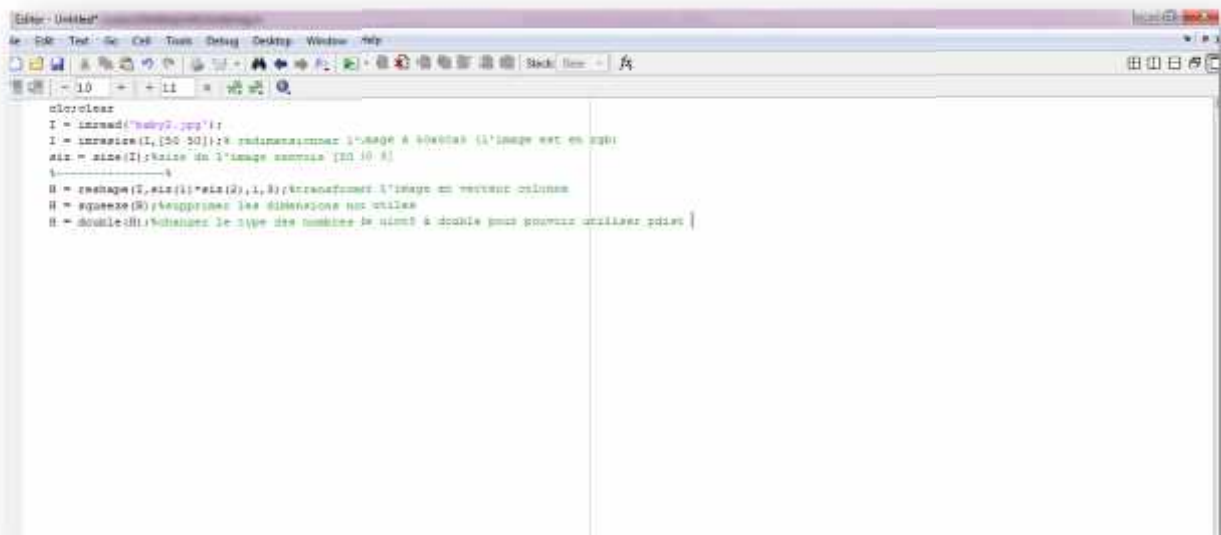


```

1 clear;clear
2 I = imread('baby2.jpg');
3 I = imresize(I,[50 50]);% redimensionner l'image à 50x50 (l'image est en rgb)
4 siz = size(I);%size de l'image renvoie [50 50 3]

```

```
3)H = reshape(I,siz(1)*siz(2),1,3);%transformer l'image en vecteur colonne
4)H = squeeze(H);%supprimer les dimensions non utiles
5)H = double(H);%changer le type des nombres de uint8 à double pour pouvoir
utiliser pdist
```



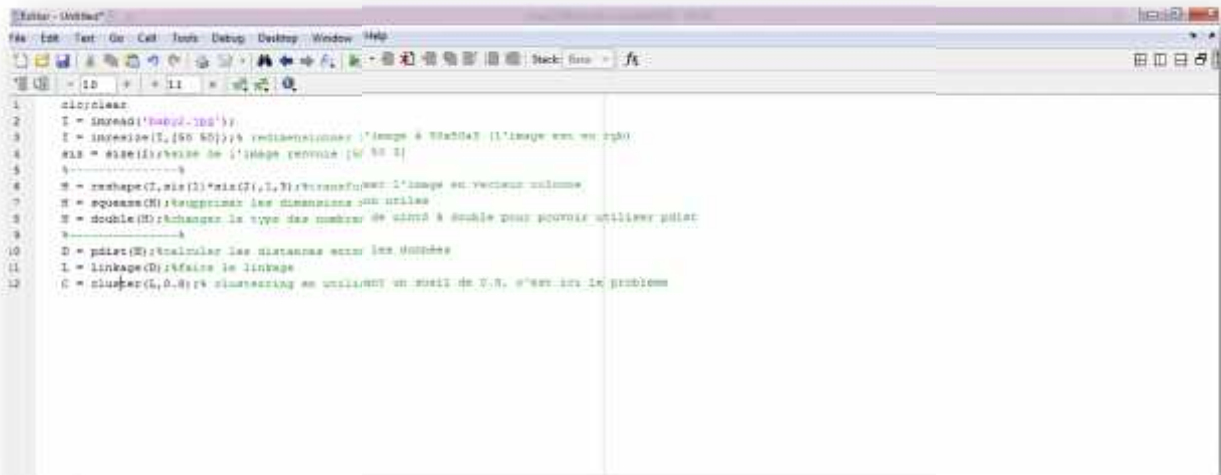
```

clear;clear
I = imread('baby2.jpg');
I = imresize(I,[50 50]);% redimensionner l'image à 50x50 (l'image est en rgb)
siz = size(I);%size de l'image renvoie [50 50 3]
%-----
H = reshape(I,siz(1)*siz(2),1,3);%transformer l'image en vecteur colonne
H = squeeze(H);%supprimer les dimensions non utiles
H = double(H);%changer le type des nombres de uint8 à double pour pouvoir utiliser pdist

```

```
6)D = pdist(H);%calculer les distances entre les données
7)L = linkage(D);%faire le linkage
```

```
8)C = cluster(L,0.8);% clusterring en utilisant un seuil de 0.8, c'est ici le probleme
```



```

1 clear;
2 I = imread('img2-100');
3 I = imresize(I,[50 50]);% redimensionner l'image à 50x50 (l'image est en rgb)
4 siz = size(I);% taille de l'image renvoie [5 50 3]
5
6 M = reshape(I,siz(2)*siz(3),siz(1));% transformer l'image en vecteur colonne
7 M = squeeze(M);% supprimer les dimensions qui n'ont
8 M = double(M);% changer le type des nombres de uint8 à double pour pouvoir utiliser plot
9
10 D = pdist(M);% calculer les distances entre les données
11 L = linkage(D);% faire le linkage
12 C = cluster(L,0.8);% clusterring en utilisant un seuil de 0.8, c'est ici le probleme

```

```
Nc = max(C);% renvoie le max de C qui est le nombre de clusters
```

```

for i = 1:Nc
    f = find(C == i);% renvoie les indices des données qui appartiennent au même cluster i
    pix_prch = H(f,:); % renvoie les données qui appartiennent au même cluster i
    meanp = mean(pix_prch);% calculer le pixel moyen de ce cluster
    meanp = round(meanp);% l'approximer au nombre naturel le plus proche (ex : 251.6 devient 252)
    for k = 1:length(f)
        Hc(f(k),:) = meanp; % chaque pixel du même cluster est remplacé par le ségment moyen du cluster
    end
end
end

```

```

1 clearvars
2 I = imread('babu.jpg');
3 I = imresize(I,[50 50]);% redimensionner l'image à 50x50 (l'image est en rgb)
4 siz = size(I);%taille de l'image renvoie [5 50 3]
5 k = 1;
6 H = reshape(I,siz(1)*siz(2),1,3);%transformer l'image en vecteur colonne
7 H = squeeze(H);%supprimer les dimensions non utiles
8 H = double(H);%changer le type des données de uint8 à double pour pouvoir utiliser pdist
9 k = k + 1;
10 D = pdist(H);%calculer les distances entre les données
11 C = linkage(D);%faire le linkage
12 C = cluster(L,S,0.5);% clustering en utilisant un seuil de 0.5, c'est ici le problème
13 k = k + 1;
14 Hc = max(C);% renvoie le max de C qui est le nombre de clusters
15 k = k + 1;
16
17 for i = 1:Hc
18     f = find(C == i); % renvoie les indices des données qui appartiennent au même cluster i
19     pts_proch = H(f,:); % renvoie les données qui appartiennent au même cluster i
20     meanpt = mean(pts_proch);% calculer le point moyen de ce cluster
21     meanp = round(meanpt);% l'approximer au nombre entier le plus proche (pas à 255.5 devient 255)
22     for k = 1:length(f)
23         Hc(f(k),:) = meanp; % chaque pixel du même cluster est remplacé par le segment moyen du cluster
24     end
25 end
26

```

- 9) `Ic = reshape(Hc,siz(1),siz(2),3);% retransformer le vecteur nouveau en matrice de dimension (50,50,3) qui est l'image ségmentée`
- 10) `Ic = uint8(Ic);% rechanger le type de données de double en uint8 pour pouvoir afficher l'image`
- 11) `imshow(Ic);% afficher l'image résultat`

```

1 clearvars
2 I = imread('babu.jpg');
3 I = imresize(I,[50 50]);% redimensionner l'image à 50x50 (l'image est en rgb)
4 siz = size(I);%taille de l'image renvoie [5 50 3]
5 k = 1;
6 H = reshape(I,siz(1)*siz(2),1,3);%transformer l'image en vecteur colonne
7 H = squeeze(H);%supprimer les dimensions non utiles
8 H = double(H);%changer le type des données de uint8 à double pour pouvoir utiliser pdist
9 k = k + 1;
10 D = pdist(H);%calculer les distances entre les données
11 C = linkage(D);%faire le linkage
12 C = cluster(L,S,0.5);% clustering en utilisant un seuil de 0.5, c'est ici le problème
13 k = k + 1;
14 Hc = max(C);% renvoie le max de C qui est le nombre de clusters
15 k = k + 1;
16
17 for i = 1:Hc
18     f = find(C == i); % renvoie les indices des données qui appartiennent au même cluster i
19     pts_proch = H(f,:); % renvoie les données qui appartiennent au même cluster i
20     meanpt = mean(pts_proch);% calculer le point moyen de ce cluster
21     meanp = round(meanpt);% l'approximer au nombre entier le plus proche (pas à 255.5 devient 255)
22     for k = 1:length(f)
23         Hc(f(k),:) = meanp; % chaque pixel du même cluster est remplacé par le segment moyen du cluster
24     end
25 end
26
27 Ic = reshape(Hc,siz(1),siz(2),3);% retransformer le vecteur colonne en matrice de dimension [ 50,50,3] qui est l'image ségmentée
28 Ic = uint8(Ic);% rechanger le type de données de double en uint8 pour pouvoir afficher l'image
29 imshow(Ic);% afficher l'image résultat

```

Conclusion :

Le traitement d'images permet de modifier le contenu des images afin de tirer l'information utile pour une application particulière. Matlab offre de nombreuses possibilités de traitement avec une palette très fournie d'outils prêts à l'emploi. L'inconvénient majeur de Matlab réside dans sa relative lenteur pour effectuer certaines opérations de calculs (par exemple la transformée de Fourier). Toutefois Matlab permet de déployer rapidement des tests pour vérifier la validité d'une méthode de traitement d'images. La manipulation d'images revient à la manipulation de matrices qui est très facile grâce au langage de haut niveau de Matlab.

Conclusion Générale

conclusion générale

Conclusion générale :

L'objet essentiel de ce travail était de segmenter des images avec l'algorithme de classification ascendante hiérarchique (AGNES).

Les méthodes de clustering comme toutes les autres méthodes de classification, ont leurs avantages, comme on a vu la classification descendante hiérarchique qui contient les deux méthodes K-MEANS et K-MEDOIDS ensuite la méthode ascendante hiérarchique DIANA et l'algorithme d'AGNES à la fin on intéressons à utiliser ce dernier dans notre projet (la segmentation d'image) elle est totalement différente de ce que nous l'avons vu jusqu'à maintenant il est facile à manipuler, efficace, et il utilise plus d'une solution il nous permet de segmenter entre les images hiérarchiquement par la classification ascendante hiérarchique.

Objectifs : -amélioration

- élimination du bruit

- rehaussement du contraste

- visualisation transmission stockage compression et reconstruction détection

- analyse

- reconnaissance (théorie de la décision)

- intelligence artificielle



Remerciements

Avant tout je remercie :

*Dieu le tout puissant de m'avoir donné le courage et la patience pour
terminer
ce travail.*

*et tiens à remercier toutes les personnes
qui m'ont aidée de près ou de loin.*

Mes remerciements à

Mon encadreur Mr. YOUNES GUELLOUMA.

*Et à tous les professeurs de département del'informatique de
Laghout.*

*Sans oublier de remercier les membres de jury qui ont bien voulu me
faire l'honneur d'examiner mon travail*

Bibliographie

- thèse :relevé et classification automatique des dégradations des routes bitumées.
- thèse :Traitement d'images médicales sous c#
- thèse : image numérique et traitement d'image.
- thèse : Université de Nice - Sophia Antipolis (Traitement du Signal et des Images).
- binarisation d'image de documents graphiques (institut de la francophonie pour l'informatique).
- morphologie mathématique système dynamique et applications au traitement des images (université de marne la vallée).

Sites web

- 1 fr.wikipedia.org/wiki/Segmentation_d'image
- 2 www.univ-sba.dz/rcam/liens/docs/traitement%20image.pdf
- 3 <http://www.sitezéro.com/>
- 4 glotin.univ-tln.fr/MCBIR/Segmentation_images_principes.pdf