Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
وزارة التعليم العالي والبحث العلمي

Badji Mokhtar Annaba University | **جامعة باجي مختار– عنابة**

Université Badji Mokhtar – Annaba

Faculté de Technologie | **كلية التكنولوجيا**

Département d'informatique | **قسم الإعلام الآلي**

Thèse

Présentée en vue de l'obtention du diplôme de Doctorat LMD

Doctorat

Spécialité : SEM (**S**ystèmes **E**mbarqués et **M**obilité)
Filière : Informatique

Par :

BOUCHEMAL Amira

Thème :

Segmentation d'Images Utilisant la Fouille de Données et Implémentation sur Terminaux Mobiles

N°	Nom et prénom	Grade	Établissement	Qualité
01	Ghanemi Salim	Prof	Université Badji Mokhtar - Annaba	Président
02	Kimour Mohamed Tahar	Prof	Université Badji Mokhtar - Annaba	Rapporteur
03	Khetatba Mourad	MCA	Université Badji Mokhtar - Annaba	Examinateur
04	Benabdallah Ahcene Youcef	MCA	Université 8 Mai 1945 - Guelma	Examinateur
05	Maazouzi Faiz	MCA	Université Mohamed Cherif Messaadia - Souk Ahras	Examinateur

Année : 2026

Remerciements

Tout d'abord, je remercie Allah le miséricordieux de m'avoir donné la puissance, le courage, et la détermination pour finaliser ce travail.

Mes vifs remerciements, ma profonde gratitude, mes sincères respects vont à mon directeur de thèse Professeur Kimour Mohamed Tahar pour son encadrement qualifié et ses conseils judicieux, merci d'avoir cru en moi, et toujours incité à aller de l'avant. Votre accompagnement infaillible, tant sur le plan scientifique que personnel, a transformé cette aventure doctorale en une expérience enrichissante et inoubliable.

J'exprime également ma profonde reconnaissance au Professeur Boudour Rachid, directeur du laboratoire LASE auquel j'ai eu l'honneur d'appartenir.

Je tiens à remercier chaleureusement Monsieur Ghanemi Salim, président du jury, pour l'honneur qu'il me fait en évaluant mon travail et pour l'enseignement inspirant qu'il m'a dispensé au cours de ma formation. Ma reconnaissance va également à Monsieur Khetatba Mourad, pour sa contribution à l'étude de ce travail doctoral et pour m'avoir enseigné avec rigueur et passion. Je tiens aussi à remercier Monsieur Benabdallah Ahcene Youcef pour l'intérêt accordé à l'analyse de ce projet scientifique. J'exprime également ma sincère reconnaissance à Monsieur Maazouzi Faiz pour l'attention portée à ce travail de recherche et pour sa participation précieuse au jury. Votre présence tous lors de cette soutenance donne à cette étape finale une dimension toute particulière.

Merci à tout le personnel du département d'informatique de l'université Badji Mokhtar Annaba, merci également à tous les enseignants qui ont contribué à ma formation depuis mon plus jeune âge.

Dédicaces

La réussite n’est jamais le fruit d’un seul individu, mais le reflet d’un chemin parcouru avec ceux qui nous inspirent, nous soutiennent et nous guident. Cette thèse est dédiée à :

Ma famille

Mes amis

Résumé

Les techniques de clustering basées sur la densité sont largement utilisées dans le data mining dans divers domaines. DBSCAN est l'un des algorithmes de clustering basés sur la densité les plus populaires, caractérisé par sa capacité à découvrir des clusters de différentes formes et tailles, et à séparer le bruit et les valeurs aberrantes. Cependant, deux limites fondamentales subsistent : le paramètre d'entrée requis du seuil de distance Eps et son inefficacité pour regrouper des ensembles de données avec différentes densités. Pour surmonter ces inconvénients, une technique basée sur les statistiques est proposée dans ce travail. Spécifiquement, la technique proposée utilise une mesure de densité appropriée basée sur les k plus proches voisins, à partir de laquelle l'ensemble de données est ordonné par ordre décroissant de densité. Elle utilise ensuite l'inégalité statistique de Chebyshev comme moyen approprié pour gérer des distributions arbitraires et déterminer automatiquement différentes valeurs Eps pour des clusters de différentes densités. Des applications menées sur des ensembles de données synthétiques et réelles ont démontré son efficacité et sa précision. Les résultats indiquent sa supériorité par rapport à DBSCAN, DPC et à leurs améliorations récemment proposées. Appliquée à la segmentation d'images, notre approche de clustering a permis d'obtenir de meilleurs résultats de segmentation.

Mots clés : Clustering, Clustering basé sur la densité, DPC, DBSCAN.

Abstract

Density-based clustering techniques are widely used in data mining on various fields. DBSCAN is one of the most popular density-based clustering algorithms, characterized by its ability to discover clusters with different shapes and sizes, and to separate noise and outliers. However, two fundamental limitations are still encountered that is the required input parameter of Eps distance threshold and its inefficiency to cluster datasets with various densities. For overcoming such drawbacks, a statistical based technique is proposed in this work. Specifically, the proposed technique utilizes an appropriate k- nearest neighbor density, based on which it sorts the dataset in descending order and, using the statistical Chebyshev's inequality as a suitable means for handling arbitrary distributions, it automatically determines different Eps values for clusters of various densities. Experiments conducted on synthetic and real datasets have demonstrated its efficiency and accuracy. The results indicate its superiority compared with DBSCAN, DPC, and their recently proposed improvements. Applied to image segmentation, our clustering approach has achieved better segmentation results.

Keywords : Clustering, Density-based Clustering, DPC, DBSCAN.

الملخص

تُستخدم تقنيات التجميع القائمة على الكثافة على نطاق واسع في استخراج البيانات في مختلف المجالات. DBSCAN هي واحدة من خوارزميات التجميع المعتمدة على الكثافة الأكثر شيوعًا، ويتميز بقدرته على اكتشاف التجمعات ذات الأشكال والأحجام المختلفة، وفصل الضوضاء والقيم المتطرفة. علي الرغم من هذا، لا يزال هناك اثنين من القيود الأساسية التي تواجهها و هما معلمة الإدخال المطلوبة لعتبة مسافة Eps وعدم كفاءتها في تجميع مجموعات البيانات ذات الكثافات المختلفة. للتغلب على هذه العيوب، يقترح أسلوب إحصائي في هذا العمل. على وجه التحديد، تستخدم التقنية المقترحة كثافة k أقرب جار المناسبة، بناءً عليه يقوم بفرز مجموعة البيانات بترتيب تنازلي وباستخدام متباينة Chebyshev الإحصائية كوسيلة مناسبة للتعامل مع التوزيعات العشوائية، فهو يحدد تلقائيًا قيم Eps مختلفة للمجموعات ذات الكثافات المختلفة. وقد أثبتت التجارب التي أجريت على مجموعات البيانات الاصطناعية والحقيقية كفاءتها ودقتها. وتشير النتائج إلى تفوقها مقارنة مع DBSCAN وDPC والتحسينات المقترحة مؤخرا. عند تطبيقه على تقسيم الصور، نجحت طريقة التجميع الخاصة بنا في تحقيق نتائج تقسيم أفضل.

الكلمات المفتاحية : التجميع، التجميع على أساس الكثافة، DBSCAN،DPC .

Table des Matières

Liste des figures

Liste des tableaux

Liste des abréviations

Eps	Rayon de voisinage (Epsilon)
MinPts	Nombre minimal de points
KNN	K-Nearest Neighbors (K plus proches voisins)
ROI	Region Of Interest (Région d'intérêt)
2D	Deux dimensions
3D	Trois dimensions
CAD	Computer-Aided Diagnosis (Diagnostic assisté par ordinateur)
CAO	Diagnostic Assisté par Ordinateur
PDE	Partial Differential Equation (Équation aux dérivées partielles)
WFA	Weighted Formula Approach
PTA	Pareto Approach
IRM	Imagerie par Résonance Magnétique
RAG	Region Adjacency Graph
IA	Intelligence Artificielle
CNN	Convolutional Neural Network
ANN	Artificial Neural Network
MLP	Multi-Layer Perceptron
FNN	Feedforward Neural Network
RNN	Recurrent Neural Network
LSTM	Long Short-Term Memory
GRU	Gated Recurrent Unit
CTC	Connectionist Temporal Classification
SAM	Spatial Access Method
ARI	Adjusted Rand Index
F1	F1-score

La segmentation d'images est un vaste sujet de recherche; de nombreuses recherches ont été réalisées dans ce contexte. La segmentation d'images est une procédure cruciale pour la plupart des tâches de détection d'objets, de reconnaissance d'images, d'extraction de caractéristiques et de classification qui dépendent de la qualité du processus de segmentation [1]. C'est la division d'une image spécifique en un nombre de segments homogènes; par conséquent, la représentation d'une image sous des formes simples et faciles augmente l'efficacité de la reconnaissance de formes [2] [3].

La segmentation d'images est la fonction la plus critique de l'analyse et du traitement d'images. C'est un mécanisme utilisé pour diviser une image en plusieurs segments. Cela rendra l'image fluide et facile à évaluer. La segmentation est le processus le plus essentiel et crucial pour faciliter la délimitation, la caractérisation et la visualisation des régions d'intérêt dans une image particulière [4] [5]. L'objectif principal est de rendre l'image plus simple et plus significative. Et ce, en divisant une image en plusieurs segments pour permettre une analyse plus approfondie et isoler les régions d'interêt [3]. Le partitionnement de l'image sera basé sur certaines caractéristiques de l'image telles que la couleur, la texture, la valeur d'intensité des pixels, etc.

La segmentation est une étape clé dans l'analyse des images médicales. Elle permet d'éviter les interférences provenant de la zone située en dehors de la région d'intérêt (ROI) et d'extraire plus précisément les caractéristiques (telles que la forme, la texture, etc.) des tissus malades. Ainsi, elle est d'une grande importance pour la prédiction de la maladie et le traitement adjuvant de la lésion [3] [5]. En raison de son importance, des recherches approfondies ont été menées sur ce problème et un certain nombre d'approches ont été proposées, telles que les méthodes de seuil, le data mining [6], particulièrement les algorithmes de clustering, les segmentations basées sur l'entropie, les réseaux neuronaux artificiels, les méthodes de croissance de région, etc.

Parmi toutes ces approches, les méthodes basées sur l'apprentissage profond ont gagné en popularité ces dernières années, en raison de la haute qualité de leurs segmentations. Cependant, ces méthodes nécessitent souvent des échantillons abondants comme données d'apprentissage, qui ne sont pas toujours disponibles pour certains types d'images médicales.

Ainsi, les algorithmes de segmentation basés sur le clustering, tels que K-means, fuzzy C-means (FCM) [7] [8] et le clustering basé sur la densité restent de bonnes alternatives, en

raison de leur nature non supervisée [9]. Les chercheurs ont mené des recherches approfondies sur la segmentation d'images et proposé diverses méthodes efficaces.

Le data mining est un processus d'extraction d'informations et de modèles utiles à partir de données volumineuses. Il est également appelé processus de découverte de connaissances, exploration de connaissances à partir de données, extraction de connaissances ou analyse de données/modèles [6]. Il est utilisé pour rechercher dans une grande quantité de données afin de trouver des données utiles [2] [6]. Le but de cette technique est de trouver des modèles jusqu'alors inconnus. Une fois ces modèles identifiés, ils peuvent ensuite être utilisés pour prendre certaines décisions.

En fonction du type de tâche du data mining, des techniques appropriées sont appliquées. Les tâches de data mining descriptives catégorisent les caractéristiques des données dans un ensemble de données cible en fonction d'événements passés ou récents. Les tâches prédictives fournissent les résultats de requêtes futures basées sur des données passées. La classification, le clustering, la régression, la détection des valeurs aberrantes, les règles d'association, les modèles séquentiels et la prédiction sont quelques-unes des techniques de data mining les plus couramment utilisées [10]. La classification, la régression et la détection externe sont des techniques de data mining prédictives. Le clustering, la découverte de modèles séquentiels et les règles d'association sont des techniques de data mining descriptives [10]. Le clustering est le processus consistant à regrouper une collection d'objets en classes d'objets similaires, c'est à dire en clusters, de sorte que les objets d'un cluster présentent une grande similitude les uns par rapport aux autres mais sont très différents des objets des autres clusters [1].

Les algorithmes de clustering peuvent être classés en algorithmes basés sur des partitions, algorithmes hiérarchiques, algorithmes basés sur la densité et algorithmes basés sur une grille. Ces méthodes varient en (i) les procédures utilisées pour mesurer la similarité (au sein et entre les clusters) (ii) l'utilisation de seuils dans la construction de clusters (iii) le mode de regroupement, c'est à dire s'ils permettent aux objets d'appartenir strictement à un seul cluster ou peuvent appartenir à plusieurs clusters à différents degrés et la structure de l'algorithme [11].

Les méthodes basées sur la densité, qui constituent la principale préoccupation de notre travail, appartiennent au clustering partitionnel. Les clusters basés sur la densité sont définis comme des clusters qui se différencient des autres clusters par des densités variables, ce qui signifie qu'un groupe qui a une région dense d'objets peut être entouré de régions de faible densité [12].

Les algorithmes de clustering basés sur la densité s'appuient sur la notion de densité pour identifier des clusters de formes arbitraires, de tailles et de densités variables. Il existe plusieurs algorithmes basés sur la densité, les plus populaires sont DBSCAN, DPC, et leurs variantes [3] [12].

DBSCAN localise les régions de haute densité qui sont séparées les unes des autres par des régions de faible densité. Pour un ensemble de données spatiales donné et des objets dans un espace bidimensionnel, nous avons besoin de deux paramètres d'entrée pour former un cluster: Eps et MinPts. Eps donne le rayon d'un seul cluster et MinPts donne le nombre minimum de points que nous souhaitons avoir dans le cluster donné [12] [13].

Cependant, DBSCAN présente aussi certaines limites, en particulier, DBSCAN n'est pas un algorithme entièrement déterministe : les points frontières accessibles depuis plusieurs clusters peuvent faire partie d'un autre cluster, selon l'ordre de traitement des données [13] [14]. Aussi, si l'échelle et les données ne sont pas compréhensibles, il peut être très difficile de choisir une distance significative par rapport au seuil Eps. Et l'inconvénient important est la procédure très laborieuse pour déterminer les paramètres requis pour la procédure d'algorithme correcte [14].

Il est difficile pour l'utilisateur de déterminer les paramètres du seuil de séparation global Eps et le nombre minimum MinPts pour chaque cluster [14]. Le résultat du clustering dépend de ces paramètres d'entrée, mais comme il n'existe pas de règles exactes pour le calcul, il est difficile pour l'utilisateur de déterminer les paramètres.

Dans cette thèse, nous avons abordé le problème de la segmentation d'images en utilisant le clustering. Le clustering ou regroupement de données est la technique clé du data mining [10]. C'est un processus de regroupement de données en classes ou clusters de sorte que les objets d'un cluster présentent une grande similarité entre eux, mais soient très différents des objets d'autres clusters. Une image peut être considérée comme un ensemble de données spatiales contenant une énorme quantité de données qui doivent être traitées pour les rendre compréhensibles. La segmentation d'images est mise en œuvre de manière à partitionner une image en un ensemble de pixels connectés afin de la regrouper en groupes homogènes et non superposés [15] [1].

La méthode proposée utilise une mesure de densité appropriée basée sur les k plus proches voisins, sur cette base, l'ensemble de données est trié par ordre décroissant et l'inégalité statistique de Chebyshev est utilisée comme méthode appropriée pour gérer les distributions

arbitraires, il détermine automatiquement différentes valeurs Eps pour des clusters de densités différentes. Appliquée à la segmentation d'images, notre approche de clustering a permis d'obtenir de meilleurs résultats de segmentation.

Nous avons donc choisi de structurer ce travail de thèse en 6 chapitres. Le premier chapitre est consacré à un état de l'art concernant la segmentation d'images et ses différentes techniques. Le deuxième chapitre présente les différents outils de segmentation, notamment le choix du type de segmentation approprié. Le troisième chapitre discute l'un des outils de segmentation les plus couramment utilisés qui est le clustering, ce chapitre est une étude des diverses techniques de segmentation. Le quatrième chapitre a pour objectif de faire le point sur les méthodes de clustering par densité, notamment l'algorithme DBSCAN qui est le sujet fondamental de notre recherche. Le cinquième chapitre présente une synthèse des travaux existants dans le domaine de segmentation d'images, en abordant notamment les algorithmes de clustering et leurs variantes basées sur la densité. Enfin, le sixième chapitre est centré sur l'évaluation de la performance de notre approche ainsi qu'aux tests et résultats obtenus. Pour conclure, nous clôturons cette thèse par une récapitulation générale des principaux résultats et perspectives.

Chapitre I.Segmentation d'Images

1 Introduction

La segmentation d'images est l'un des domaines de recherche les plus populaires en vision par ordinateur. Devenu un pôle de recherche dans le domaine du traitement d'images et de la vision par ordinateur, la segmentation d'images fait référence au processus de partitionner une image en parties significatives ayant des caractéristiques et des propriétés similaires, et constitue une étape essentielle facilitant l'analyse d'images.

La segmentation d'images est à la base de la reconnaissance de formes et de la compréhension d'images. Elle est étroitement liée à de nombreuses disciplines et domaines, par exemple, les véhicules autonomes [16], la technologie médicale intelligente [17], les moteurs de recherche d'images [18], l'inspection industrielle et la réalité augmentée.

La segmentation d'images peut être classée en deux types de base : segmentation locale (partie ou région de l'image) et la segmentation globale (concerne la segmentation de l'ensemble de l'image constituée d'un grand nombre de pixels) [19].

Les régions d'une segmentation d'images doivent être uniformes et homogènes en ce qui concerne certaines caractéristiques telles que le ton de gris ou la texture. Les intérieurs de région doivent être simples et sans beaucoup de petits trous. Les régions adjacentes d'une segmentation doivent avoir des valeurs significativement différentes en ce qui concerne la caractéristique sur laquelle elles sont uniformes. Les limites de chaque segment doivent être simples, non irrégulières et spatialement précises.

Il est difficile d'atteindre toutes ces propriétés souhaitées car les régions strictement uniformes et homogènes sont généralement remplies de petits trous et ont des limites irrégulières. Insister sur le fait que les régions adjacentes ont de grandes différences de valeurs peut entraîner la fusion des régions et la perte des frontières.

Comme il n'existe pas de théorie de la classification, il n'existe pas de théorie de la segmentation d'images. Les techniques de segmentation d'images sont fondamentalement ad hoc et diffèrent précisément par la manière dont elles mettent en valeur une ou plusieurs des propriétés souhaitées et par la manière dont elles équilibrent et compromettent une propriété désirée par une autre [20].

Les techniques de segmentation d'images peuvent être classées comme suit : regroupement spatial guidé par mesure, schémas de croissance de régions de liaison unique, schémas de croissance de régions de liaison hybride, schémas de croissance de région de liaison de centroïdes, schémas de regroupement spatial, et schémas de scission et de fusion. Comme on peut le constater à partir de cette brève typologie, la segmentation d'images peut être considérée comme un processus de clustering [21]. La différence entre la segmentation d'images et le clustering réside dans le fait que, dans le clustering, le regroupement se fait dans l'espace de mesure. Dans la segmentation d'images, le regroupement est clone sur le domaine spatial de l'image et il existe une interaction dans la classification entre les groupes (éventuellement superposés) dans l'espace de mesure et les groupes mutuellement exclusifs de la segmentation d'images.

Les systèmes de croissance de régions à liaison unique sont les plus simples et les plus sujets aux erreurs de fusion de régions non désirées. Les systèmes de croissance hybride et centroïde sont meilleurs à cet égard. La technique de scission et de fusion n'est pas aussi sujette à l'erreur de fusion de région indésirable. Cependant, il utilise beaucoup de mémoire et présente des limites excessives dans les régions bloquées. Le regroupement spatial guidé par l'espace de mesure tend à éviter à la fois les erreurs de fusion de régions et les problèmes de limites par blocs en raison de sa dépendance principale à l'espace de mesure. Mais les régions produites ne sont pas délimitées en douceur, et ils ont souvent des trous, donnant l'effet du sel et du poivre bruit [20]. Les schémas de regroupement spatial sont peut-être meilleurs à cet égard, mais ils n'ont pas été suffisamment testés. Les schémas de liaison hybride semblent offrir le meilleur compromis entre avoir des limites lisses et peu de fusions de régions indésirables.

Dans ce qui suit une brève description des idées principales qui sous-tendent les principales techniques de segmentation d'images et donne des exemples de résultats pour plusieurs d'entre elles. Certaines techniques peuvent produire de très petites régions. Lorsque de petites régions sont éliminées, une description de ce fait est faite dans la description des résultats.

2 Caractéristiques de l'image

2.1 Caractéristiques d'apparence

En général, les apparences visuelles des objectifs sont associées à des intensités individuelles en pixels ou en voxels (valeurs de gris) et à des interactions spatiales entre les intensités en termes de co-occurrences d'intensité par paires ou d'ordre supérieur dans une image.

Intensité : les intensités individuelles peuvent guider la segmentation si leurs plages ou, plus généralement, les distributions de probabilité pour un objet d'intérêt et son arrière-plan diffèrent dans une large mesure. Ensuite, l'objet entier ou au moins la plupart de ses pixels / voxels peuvent être facilement séparés de l'arrière-plan en comparant les intensités à un ou plusieurs seuils dérivés des plages ou distributions des intensités apprises. Cependant, en règle générale, des fonctionnalités discriminantes plus puissantes doivent être utilisées.

Interaction spatiale : l'apparition de certaines textures peut être associée à des motifs spatiaux de variations locales de l'intensité pixel / voxel ou à des distributions de probabilité empiriques de cooccurrences d'intensité, appelées matrices de cooccurrence dans les cas par paires. Les modèles d'interaction spatiale probabilistes considèrent les images comme des échantillons d'un certain champ d'intensités interdépendantes aléatoires, spécifié par leur distribution de probabilité conjointe.

- Le filtrage de domaine spatial constitue le moyen le plus direct de capturer des interactions spatiales locales. Les premières tentatives se sont concentrées sur les contours d'intensité, car les textures fines ont une densité spatiale des contours supérieure à celle des textures plus grossières.
- Le filtrage du domaine de fréquence découle de l'hypothèse selon laquelle le système visuel humain décompose une image pour analyse de texture en composantes de fréquence orientées.
- Les modèles fractaux sont utiles pour la modélisation de la rugosité statistique et de l'auto-similarité à différentes échelles de nombreuses surfaces naturelles. La géométrie fractale de la nature a été introduite pour la première fois dans les années 1970 par Mandelbrot et explorée plus avant par Barnsley.
- Les modèles gaussiens probabilistes supposent des signaux d'image continus dans une plage infinie (bien que les images numériques aient un ensemble fini d'intensités Q). La distribution conjointe $p\,(g)$ est unimodale et dépend des deux paramètres, l'image moyenne de la taille XY ou XYZ et la matrice de covariance symétrique de la taille

$(XY)^2$ou$(XYZ)^2$, respectivement, à apprendre des images de formation. En raison d'un espace de paramètres extrêmement grand, seuls les modèles avec des matrices de covariance simples spéciales, par exemple des champs gaussiens indépendants avec des matrices de covariance diagonales, ont été utilisés dans la pratique. Les modèles autorégressifs simultanés d'images généralisent les mêmes modèles de série chronologique 1 D : par hypothèse, les erreurs $\varepsilon(r)$ de prédiction linéaire de chaque intensité individuelle $g(r)$ à partir des intensités voisines dans une fenêtre fixe de voisins w sont considérées comme un échantillon d'un champ aléatoire indépendant avec la même distribution de probabilité 1 D, φ (.), de chaque composante en pixels ou en voxels. La probabilité d'image est égale à la probabilité conjointe des erreurs de prédiction :

$$\boldsymbol{p(g) = \prod_{r\in R} \varphi(g(r) - \sum_{\delta\in w} \alpha\delta g(r+\delta))} \qquad \textbf{(1)}$$

2.2 Caractéristiques de forme

Les caractéristiques d'apparence (c'est à dire l'intensité et l'interaction spatiale) peuvent à elles seules ne pas capturer les objectifs dans une image en présence de bruit de capteur, sous une résolution d'image médiocre et / ou de limites diffusées ou de formes occluses.

2.3 Forme combiné / Apparence bord et fonctionnalités multi-échelle

Les contours font partie des fonctions les plus populaires pour guider la segmentation d'images. L'analyse d'image nous permet de spécifier et d'extraire différents types de contours, tels que les contours d'intensité, les contours de texture, les contours paramétriques et les bassins versants. Les limites d'intensité sont associées à de grandes discontinuités ou à des modifications rapides et continues de l'intensité à travers une image. Ces changements sont fréquemment détectés par seuillage de dérivées spatiales du premier et du second ordre des intensités (c'est à dire le gradient d'intensité et le laplacien, respectivement). Les détecteurs de contour populaires, tels que les opérateurs Sobel, Prewitt et Roberts, fournissent des approximations en différences finies du gradient. La figure 1 [22] montre les résultats de l'application de plusieurs détecteurs de contour à une image de cellule. Tous les détecteurs exploitant les dérivées du premier ou du second ordre sont trop sensibles au bruit de l'image.

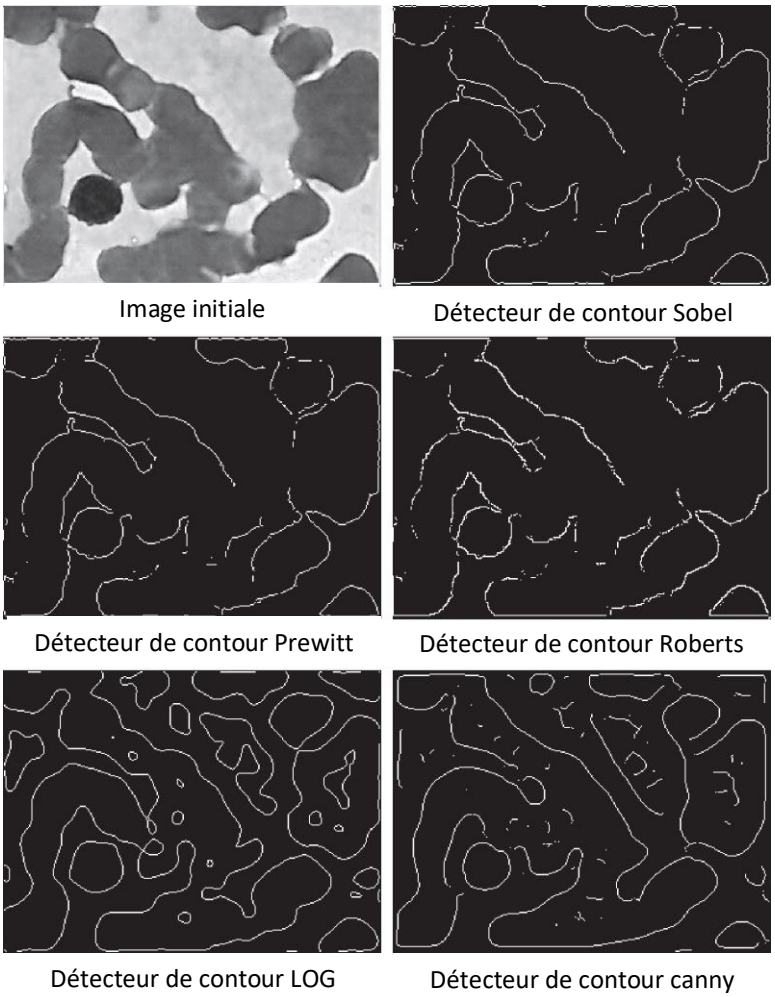


Figure 1 : Intensité des contours sur une image de cellule

Pour réduire le bruit, le lissage de l'image est généralement appliqué avant la détection des contours. Par exemple, l'opérateur Marr-Hildreth laplacien-de-gaussien (LOG) associe un filtre gaussien de lissage à un opérateur de détection zéro laplacien afin de réduire simultanément le bruit et de trouver des gouttes ou des régions d'image toujours plus claires ou plus sombres que leur environnement. Le détecteur de contour Canny trouve les blobs en recherchant les maxima de gradient locaux à l'aide du dérivé du filtre Gaussien (DOG). La figure 1 nous permet de comparer les détecteurs de contour basés sur des dérivés simples aux détecteurs LOG et Canny.

En plus des taches, des motifs plus complexes d'aspect et de forme peuvent être trouvés grâce à la détection des bords de texture. Des descripteurs de texture, tels que, par exemple, des coefficients spectraux de type Gabor, sont souvent utilisés pour représenter différentes

propriétés d'échelle et de direction d'une texture et peuvent ainsi guider différentes techniques de segmentation.

Les bords d'intensité ou de texture permettent de détecter des modèles de bords paramétriques tels que des cercles 2D, des lignes droites, des ellipses, des sphères 3D, des ellipsoïdes ou des formes plus complexes. Par exemple, un détecteur de bord suivi d'un seuillage est appliqué à une image. Ensuite, la transformation de Hough de la carte binaire des arêtes détectées détecte les courbes paramétriques requises même si elles sont pivotées ou mises à l'échelle. Pour les formes complexes, une telle représentation paramétrique peut être obtenue en apprenant un modèle précédent à partir d'un ensemble d'apprentissage, bien qu'un tel cadre puisse s'avérer trop lent et difficile à mettre en œuvre en raison d'un espace de paramètres de grande dimension. De plus, l'efficacité de la transformation de Hough dépend à la fois de la qualité de l'image et de la qualité de la détection des contours. Pour améliorer ce dernier point, seuls les bords connectés appropriés doivent être sélectionnés dans la carte des contours produite par le détecteur. Cela peut être fait, par exemple, en attribuant des poids à chaque contour candidat détecté et en utilisant des techniques de recherche de graphe ou une programmation dynamique pour trouver les contours connectés optimaux minimisant une certaine fonction de coût.

3 Techniques de segmentation

La segmentation de l'image est considérée comme l'un des processus les plus importants du traitement de l'image. La segmentation d'images est la technique de division ou de partitionnement d'une image en parties, appelées segments. Il est surtout utile pour des applications telles que la compression d'images ou la reconnaissance d'objets, car pour ces types d'applications, il est inefficace de traiter l'ensemble de l'image. Ainsi, la segmentation d'images est utilisée pour segmenter les parties de l'image pour un traitement ultérieur. Il existe plusieurs techniques de segmentation d'images, qui partitionnent l'image en plusieurs parties en fonction de certaines caractéristiques de l'image telles que la valeur d'intensité de pixel, la couleur, la texture, etc [19]. Toutes ces techniques sont classées par catégorie en fonction de la méthode de segmentation utilisée.

La grande complexité et la variabilité des apparences et des formes des structures anatomiques font de la segmentation d'images l'une des tâches les plus difficiles et essentielles de tout système de CAD. En raison de la diversité des objets d'intérêt, des modalités d'image et des problèmes de CAO, il n'existe pas de jeu de fonctions universel ni de technique de segmentation générale [22]. Certaines techniques populaires basées sur des règles, des statistiques, des atlas et des modèles déformables, ainsi que leurs principales forces et faiblesses sont décrites ci-dessous.

La recherche sur la segmentation des images suscite depuis de nombreuses années une grande attention. Des milliers de techniques de segmentation différentes sont présentes dans la littérature, mais il n'y a pas une seule méthode qui puisse être considérée comme bonne pour des images différentes, toutes les méthodes ne sont pas aussi bonnes pour un type d'image particulier.

Ainsi, le développement d'algorithmes pour une classe d'images peut ne pas toujours être appliqué à une autre classe d'images. Par conséquent, il existe de nombreuses difficultés, telles que le développement d'une approche unifiée de la segmentation d'images pouvant être appliquée à tous les types d'images. Même le choix d'une technique appropriée pour un type d'image spécifique est un problème difficile.

Ainsi, malgré plusieurs décennies de recherche, il n'existe pas de méthode universellement acceptée pour la segmentation d'images et reste donc un problème épineux en traitement d'images et en vision par ordinateur.

Basées sur différentes technologies, les approches de segmentation d'images sont actuellement divisées en catégories suivantes, basées sur deux propriétés d'image.

• Détecter les discontinuités

Cela implique de partitionner une image en fonction de changements brusques d'intensité, ce qui inclut des algorithmes de segmentation d'images tels que la détection de contour.

• Détecter les similitudes

Cela signifie partitionner une image en régions similaires selon un ensemble de critères prédéfinis ; cela inclut des algorithmes de segmentation d'image tels que le seuillage, la croissance de région, la division et la fusion de région.

3.1 Segmentation basée sur la détection de contour

Cette méthode tente de résoudre la segmentation d'images en détectant les contours ou les pixels entre les différentes régions présentant une transition rapide en intensité, puis liés pour former des limites d'objet fermées. Le résultat est une image binaire. Sur la base de la théorie, il existe deux méthodes de segmentation principales : l'histogramme des gris et la méthode des gradients.

3.1.1 Technique d'histogramme des gris

Le résultat de la technique de détection des contours dépend principalement de la sélection du seuil S, et il est vraiment difficile de rechercher une intensité maximale et minimale des niveaux de gris car l'histogramme des gris est inégal pour l'impact du bruit. Nous substituons donc approximativement les courbes d'objet et d'arrière-plan à deux courbes coniques gaussiennes, dont l'intersection est la vallée de l'histogramme. Le seuil S est la valeur grise du point d'intersection de cette vallée.

3.1.2 Méthode basée sur le gradient

Le dégradé est la première dérivée de l'image $f\ (x, y)$, lorsqu'il y a un changement brusque d'intensité près du bord et qu'il y a peu de bruit d'image. La méthode basée sur le dégradé fonctionne bien. Cette méthode implique la convolution des opérateurs de dégradé avec l'image. La valeur élevée de l'amplitude du gradient est un lieu possible de transition rapide entre deux régions différentes. Ce sont des pixels de bord, ils doivent être liés pour former des limites fermées des régions. Les opérateurs de détection de bord couramment utilisés dans la méthode par gradient sont : opérateur Sobel, opérateur avisé, opérateur Laplace, opérateur laplacien de gaussien (LOG), etc.

Les méthodes de détection des contours exigent un équilibre entre détection de la précision et immunité au bruit dans la pratique. Si le niveau de précision de détection est trop élevé, le bruit peut créer de faux contours rendant le contour des images déraisonnable et si le degré d'immunité au bruit est trop excessif, certaines parties du contour de l'image peuvent ne pas être détectées et la position des objets peut être confondue. Ainsi, les algorithmes de détection des contours conviennent aux images simples et sans bruit, qui produisent souvent des contours manquants ou des contours supplémentaires sur des images complexes et bruitées.

3.2 Méthode de seuillage

La segmentation d'images par seuillage est une approche simple mais puissante pour la segmentation d'images comportant des objets clairs sur un fond sombre. La technique de seuillage est basée sur les régions d'espace d'image, c'est à dire sur les caractéristiques de l'image. Une opération de seuillage convertit une image à plusieurs niveaux en une image binaire, c'est à dire qu'elle choisit un seuil approprié S pour diviser les pixels de l'image en plusieurs régions et séparer les objets de l'arrière-plan. Tout pixel (x, y) est considéré comme faisant partie de l'objet si son intensité est supérieure ou égale à la valeur de seuil, c'est-à-dire $F(x, y) \geq S$, sinon les pixels appartiennent à l'arrière-plan. Selon la sélection de la valeur de seuillage, il existe deux types de méthodes de seuillage, le seuillage global et le seuil local. Lorsque S est constant, l'approche est appelée seuil global, sinon elle est appelée seuil local. Les méthodes de seuillage globales peuvent échouer lorsque l'éclairage d'arrière-plan est inégal. Dans le seuillage local, plusieurs seuils sont utilisés pour compenser une illumination inégale. La sélection de seuil est généralement effectuée de manière interactive, cependant, il est possible de dériver des algorithmes de sélection automatique de seuil.

La méthode de seuillage est limitée car deux classes seulement sont générées et ne peut pas être appliquée aux images multicanaux. De plus, le seuillage ne prend pas en compte les caractéristiques spatiales d'une image car il est sensible au bruit, ces deux artefacts altérant l'histogramme de l'image, rendant la séparation plus difficile.

3.3 Méthodes de segmentation par région

Comparés à la méthode de détection des contours, les algorithmes de segmentation basés sur les régions sont relativement simples et plus immunisés contre le bruit. Les méthodes basées sur les contours partitionnent une image en fonction de changements rapides d'intensité près des contours, tandis que les méthodes basées sur les régions partitionnent une image en régions similaires en fonction d'un ensemble de critères prédéfinis. Les algorithmes de segmentation basés sur les régions incluent principalement les méthodes suivantes :

3.3.1 Région en croissance

L'expansion régionale est une procédure qui regroupe les pixels de l'ensemble de l'image en sous-régions ou régions plus grandes en fonction de critères prédéfinis. La croissance de la région peut être traitée en quatre étapes :

1. Sélectionner un groupe de pixels d'origine dans l'image d'origine.
2. Sélectionner un ensemble de critères de similarité tel que l'intensité du niveau de gris ou la couleur et configurer une règle d'arrêt.
3. Développer des régions en ajoutant à chaque graine les pixels voisins possédant des propriétés prédéfinies similaires aux pixels de graine.
4. Arrêter la croissance de la région lorsque plus aucun pixel ne remplit le critère d'inclusion dans cette région (c'est à dire la taille, la ressemblance entre un pixel candidat et le pixel développé jusqu'à présent, la forme de la région en cours de croissance).

3.3.2 Division et fusion de régions

Plutôt que de choisir des points de départ, l'utilisateur peut diviser une image en un ensemble de régions non connectées arbitraires, puis fusionner les régions afin de satisfaire les conditions d'une segmentation d'image raisonnable. La division et la fusion de régions sont généralement mises en œuvre avec une théorie basée sur des données quadri-arborescentes.

Laisser R représenter la totalité de la région d'image et sélectionner un prédicat Q.

1. On commence par l'image entière si $Q\ (R)\ =\ FAUX$, on divise l'image en quadrants, si Q est faux pour tout quadrant qui est, si $Q\ (Ri)\ =\ FAUX$, on subdivise les quadrants en sous quadrants et ainsi de suite jusqu'à ce qu'aucun fractionnement supplémentaire n'est possible.
2. Si seule la division est utilisée, la partition finale peut contenir des régions adjacentes ayant des propriétés identiques. Il est possible de remédier à cet inconvénient en autorisant la fusion et la scission, c'est à dire la fusion de toutes les régions adjacentes $Rj\&Rk$ pour lesquelles $Q\ (Rj\ U\ Rk)\ =\ VRAI$.
3. S'arrêter lorsque plus aucune fusion n'est possible.

4 Méthodes de segmentation basées sur la PDE (Partial Differential Equation)

En utilisant une méthode basée sur PDE et en résolvant l'équation PDE par un schéma numérique, on peut segmenter l'image. La segmentation des images basée sur les PDE est

principalement réalisée par un modèle de contour actif ou par des snakes. Cette méthode a été introduite pour la première fois par Kassetal en 1987. Kass a développé cette méthode pour trouver des objets familiers en présence de bruit et d'autres ambiguïtés. L'idée centrale de Snake est de transformer un problème de segmentation en un framework PDE. C'est à dire que l'évolution d'une courbe, d'une surface ou d'une image donnée est gérée par les PDE et que la solution de ces PDE correspond à ce que nous attendons avec différentes méthodes de segmentation d'images : snake, level set et modèle de Mumford-shah.

a. Snakes

Les contours ou snakes actifs sont des courbes générées par ordinateur qui se déplacent dans l'image pour trouver les limites des objets sous l'influence de forces internes et externes. Cette procédure est la suivante :

1. Snake est placé près du contour de la région d'intérêt (ROI).
2. Au cours d'un processus itératif dû à diverses forces internes et externes au sein de l'image, le snake est attiré vers la cible. Ces forces contrôlent la forme et l'emplacement du snake dans l'image.
3. Une fonction d'énergie composée de forces internes et externes est construite pour mesurer l'adéquation du contour de la région d'intérêt, minimiser la fonction d'énergie (intégrale), qui représente l'énergie totale du contour actif, les forces internes étant responsables du lissage, les forces externes guidant contours vers le contour du ROI.

L'inconvénient du snake traditionnel est qu'il nécessite l'interaction de l'utilisateur, qui consiste à déterminer la courbe entourant l'objet détecté ; la fonction d'énergie converge souvent vers une énergie locale minimale. Sensible au bruit. Plus sensible au choix de ses paramètres et ajuste de manière adaptative les paramètres dans un processus extrêmement complexe. La complexité de calcul de l'algorithme est élevée.

Pour résoudre ces problèmes, de nombreux chercheurs ont apporté diverses améliorations au modèle de base, mais les inconvénients du snake ne sont toujours pas résolus.

b. Modèle de niveau

Un grand nombre des PDE utilisées dans le traitement des images sont basées sur des courbes et des surfaces en mouvement ayant des vitesses basées sur la courbure. Dans ce

domaine, la méthode de jeu de niveaux mise au point par Osher et Sethian était très influente et utile. L'idée de base est de représenter les courbes ou les surfaces comme l'ensemble de niveaux zéro d'une surface hyper dimensionnelle supérieure. Cette technique permet non seulement d'obtenir des implémentations numériques plus précises, mais également de gérer très facilement les changements topologiques.

Il présente plusieurs avantages. Sa stabilité et son manque de pertinence avec la topologie, présente un grand avantage pour résoudre les problèmes de production de points d'angle, de cassures et de combinaisons de courbes, etc.

Étant donné que la fonction d'arrêt des contours dépend du dégradé de l'image, seuls les objets dont les contours sont définis par des dégradés peuvent être segmentés. Un autre inconvénient est que, dans la pratique, la fonction d'arrêt des arêtes n'est jamais exactement nulle aux arêtes et la courbe peut donc éventuellement passer à travers les limites des objets.

c. Modèle Mumford Shah

Le modèle Mumford-Shah utilise les informations globales de l'image comme critère d'arrêt pour segmenter l'image. Mumford-shah tire parti de l'ensemble des informations de l'image pour obtenir la meilleure segmentation possible.

d. Modèle C-V (Chan-Vese)

L'idée de base est de rechercher une partition particulière d'une image donnée dans deux régions, l'une représentant les objets à détecter et l'autre arrière-plan. Le modèle C-V n'est pas basé sur la fonction de bord, pour arrêter la courbe en évolution sur la limite souhaitée. (Il n'est pas nécessaire de lisser l'image initiale, même si elle est bruyante), l'emplacement des limites est très bien détecté. Il peut détecter des objets dont les limites ne sont pas nécessairement définies par des gradients ou des limites très lisses. À partir d'une seule courbe initiale, ce modèle peut détecter automatiquement les contours intérieurs et ne commence pas nécessairement autour des objets à détecter.

5 Segmentation Basée Sur Un Réseau De Neurones Artificiels

La segmentation basée sur le réseau de neurones est totalement différente des algorithmes de segmentation conventionnels. En cela, une image est d'abord mappée dans un réseau de neurones. Là où chaque neurone représente un pixel, le problème de segmentation d'image est

converti en problème de minimisation d'énergie. Le réseau de neurones a été formé avec un ensemble d'échantillons d'apprentissage afin de déterminer la connexion et les poids entre les nœuds. Ensuite, les nouvelles images ont été segmentées avec un réseau de neurones formés. Par exemple, nous pouvons extraire les contours d'image en utilisant des équations dynamiques qui dirigent l'état de chaque neurone vers l'énergie minimale définie par le réseau de neurones.

La segmentation du réseau neuronal comprend deux étapes importantes : l'extraction de caractéristiques et la segmentation d'images basée sur le réseau neuronal.

L'extraction des caractéristiques est très importante car elle détermine les données d'entrée du réseau neuronal. Certaines caractéristiques sont d'abord extraites des images, de sorte qu'elles deviennent adaptées à la segmentation, puis ce sont elles qui constituent l'entrée du réseau neuronal. Toutes les entités sélectionnées constituent un espace hautement non linéaire de limites de clusters.

La segmentation basée sur un réseau de neurones a quatre caractéristiques de base :

1. Capacité hautement parallèle et capacité de calcul rapide, ce qui le rend approprié pour une application en temps réel.
2. Améliorer les résultats de la segmentation lorsque les données s'écartent de la situation normale.
3. Robustesse le rendant insensible au bruit.
4. Exigence réduite d'intervention d'experts au cours du processus de segmentation d'image.

Cependant, la segmentation basée sur les réseaux de neurones présente certains inconvénients, tels que :

a) Une sorte d'information sur la segmentation doit être connue à l'avance.
b) L'initialisation peut influencer le résultat de la segmentation de l'image.
c) Le réseau de neurones doit être formé en utilisant le processus d'apprentissage à l'avance, la période de formation peut être très longue et nous devons éviter de surentraîner en même temps.

6 Segmentation basée sur le clustering

Le clustering est une tâche d'apprentissage non supervisée, dans laquelle il est nécessaire d'identifier un ensemble fini de catégories, appelées clusters, pour classifier les pixels. Le regroupement n'utilise pas d'étapes de formation, mais utilise plutôt les données disponibles. Le clustering est principalement utilisé lorsque les classes sont connues à l'avance. Un critère de similarité est défini entre les pixels, puis les pixels similaires sont regroupés pour former des clusters. Le regroupement des pixels en clusters repose sur le principe de maximisation de la similarité intra-classe et de maximisation de la similarité inter-classe. La qualité d'un résultat de classification dépend à la fois de la mesure de similarité utilisée par la méthode et de sa mise en œuvre.

Les algorithmes de clustering sont classés en clustering dur, en cluster k-means, en clustering flou, etc.

a. Clustering dur « Hard »

Le hard clustering suppose des limites nettes entre les clusters ; un pixel appartient à un et un seul cluster. Un algorithme de clustering dur populaire et bien connu est l'algorithme de clustering K-means.

L'algorithme K-means est une technique de clustering permettant de partitionner n pixels en k groupes, où $k < n$. L'algorithme K-means Développé par Mac Queen en 1965, puis perfectionné par Hartigan et Wong en 1979. L'algorithme K-means est une technique de clustering qui consiste à classer les pixels d'une image en un nombre K de clusters, où K est un entier positif, selon certaines caractéristiques de similarité sont telles que l'intensité du niveau de gris des pixels et la distance entre les intensités de pixels, de l'intensité des pixels centroïdes. Les principaux avantages de cet algorithme sont sa simplicité et son faible coût de calcul, ce qui lui permet de fonctionner efficacement sur de grands ensembles de données. Le principal inconvénient est que : K le nombre de clusters doit être déterminé, il ne donne pas le même résultat à chaque exécution de l'algorithme et les clusters résultants dépendent des affectations initiales des centroïdes. Le processus est comme suit :

1. Choisir au hasard le nombre de clusters K.
2. Calculer l'histogramme des intensités de pixels.
3. Choisir au hasard K pixels d'intensités différentes comme centroïdes.

4. Les centroïdes le découvrent en calculant la moyenne des valeurs de pixels dans une région et les placent le plus loin possible les uns des autres.
5. Comparer un pixel à chaque centroïde et l'attribuer au centroïde le plus proche pour former un cluster.

$$c^{(i)} := \arg\min \left\| x^{(t)} - \mu_j \right\| \qquad (2)$$

6. Lorsque tous les pixels ont été attribués, la mise en cluster initiale est terminée.
7. Recalculer la position des centroïdes dans Kclusters.

$$\mu_i := \frac{\sum_{i=1}^{m} 1\{c_{(i)}=j\} x^{(i)}}{\sum_{i=1}^{m} 1\{c_{(i)}=j\}} \qquad (3)$$

8. Répéter les étapes 5 et 6 jusqu'à ce que les centroïdes ne bougent plus.
9. Image segmentée en K clusters.

b. Clustering flou

Dans les applications en temps réel, l'une des tâches les plus difficiles en analyse d'image et en vision informatique consiste à classer correctement le pixel dans une image. Ainsi, lorsqu'il n'existe pas de limites nettes entre les objets d'une image, les techniques de clustering flou sont utilisées.

La technique de classification floue classe les valeurs de pixel avec une grande précision. Elle est essentiellement adaptée aux applications orientées décision telles que la classification des tissus et la détection de tumeurs, etc. La classification floue divise les pixels d'entrée en clusters ou groupes sur la base d'un critère de similarité, de telle sorte que les pixels similaires appartiennent au même cluster. Le critère de similarité utilisé peut être la distance, la connectivité, l'intensité. La partition résultante améliore la compréhension de l'être humain et aide à une prise de décision plus éclairée. L'avantage des systèmes flous est qu'ils sont faciles à comprendre, car la fonction d'appartenance partitionne correctement l'espace de données. Les algorithmes de classification floue comprennent les algorithmes FCM (fuzzy C means), GK (Gustafson-Kessel), GMD (Gaussian mixture decomposition), FCV (Fuzzy C varieties), AFC, FCS, FCSS, FCQS, FCRS, etc.

FCM est la méthode la plus acceptée car elle peut conserver beaucoup plus d'informations que les autres approches.

FCM attribue des pixels à chaque classe au moyen d'une fonction d'appartenance.

Supposons :

$X = (X_1, X_2, X_3, - - - - - - - - X_n)$

Indique une image de N pixels qui doit être divisée en clusters C, FCM suit un processus itératif qui minimise la fonction objective suivante :

$$J = \sum_{j=1}^{N} \sum_{i=1}^{C} uijm \left\| X_j - V_i \right\| \qquad (4)$$

Où, u_{ij}^{m} = fonction d'appartenance du pixel x_j dans le ième cluster.

V_i est le pixel central du ième cluster.

m est le fuzzifier qui contrôle le flou des clusters résultants et se situe entre $1<m \leq \infty$.

La fonction d'appartenance et les centres de clusters sont mis à jour. Les centres de clusters peuvent être initialisés de manière aléatoire ou par une méthode d'approximation.

L'inconvénient du FCM est que pour les images bruyantes, il ne prend pas en compte les informations spatiales, ce qui le rend sensible au bruit et à d'autres artefacts d'image. L'affectation des clusters FCM étant basée sur la distribution de l'intensité des pixels, elle la rend sensible aux variations d'intensité de l'éclairage. Pour surmonter ces inconvénients du FCM, plusieurs autres algorithmes sont introduits comme le FCM modifié, le GSFCM (Generalized spatial FCM), le FCM basé sur le décalage moyen, le FLICM (fuzzy logic information C-means clustering algorithm), le NFCM (novel FCM), l'ISFCM (improved spatial FCM).

c. Density Peak Clustering

L'algorithme de clustering basé sur les pics de densité est une approche de clustering récemment proposée. Il utilise la densité locale de chaque donnée et la distance jusqu'au voisin le plus proche avec une densité plus élevée pour isoler et identifier les centres de cluster. Une fois les centres de clustering identifiés, les autres données se voient attribuer des étiquettes égales à celles de leurs voisins les plus proches avec une densité plus élevée [23]. Cet algorithme est simple et efficace. À condition que les centres de cluster soient correctement identifiés, cela peut générer de très bons résultats de clustering. Cependant, les résultats de cet algorithme dépendent d'un paramètre dans le calcul de la densité locale.

C’est une nouvelle méthode de clustering basée sur la densité. Il passe la majeure partie de son temps d'exécution à calculer la densité locale et la distance de séparation pour chaque point de données d'un ensemble de données. Le but de cette étude est d'accélérer son calcul. En moyenne, l'algorithme DPC scanne la moitié de l'ensemble de données pour calculer la distance de séparation de chaque point de données [24].

Étant donné un ensemble de données X, la densité locale $\rho\ (x_i)$ d'un point de données $x_i \in X$ est le nombre de points de données au voisinage de x_i. C’est :

$$\boldsymbol{\rho(x_i) = |B\ (x_i)|} \qquad \textbf{(5)}$$

Où $B(x_i)$ désigne le voisinage de x_i et est défini comme l'ensemble des points de données dans X dont la distance à x_i est inférieure à un paramètre spécifié par l’utilisateur d_c.
C’est :

$$\boldsymbol{B(x_i) = \{x_j \in X | d(x_i, x_j) < d_c\}} \qquad \textbf{(6)}$$

Où $d(x_i, x_j)$ représente la distance entre x_i et x_j. Notamment, les deux équations ci-dessus utilisent le paramètre d_c comme seuil dur pour dériver le voisinage et la densité locale d'un point de données, respectivement.

La valeur de d_c peut être choisie de sorte que le nombre moyen de voisins d'un point de données soit d'environ $p\%$ du nombre de points de données dans X, et la valeur suggérée pour p soit comprise entre 1 et 2. Pour les petits ensembles de données, Rodriguez et Laio [25] ont suggéré d'utiliser un noyau exponentiel pour calculer la densité locale, comme le montre l'équation suivante :

$$\boldsymbol{\rho(x_i) = \sum_{x_j \in X} exp\left(-\frac{d(x_i,x_j)^2}{{d_c}^2}\right)} \qquad \textbf{(7)}$$

La distance de séparation $\delta(x_i)$ de x_i est la distance minimale de x_i à tout autre point de données avec une densité locale $>\rho(x_i)$, ou la distance maximale de x_i à tout autre point de données dans X s'il n'existe aucun point de données avec une densité locale $>\rho(x_i)$, comme le montre l'équation suivante :

$$\boldsymbol{\delta(x_i) = \begin{cases} \min\limits_{j:\rho(x_j)>\rho(x_i)} d(x_i, x_j), si\ \rho(x_i) < \max\limits_{x_j \in X} \rho(x_j) \\ \max\limits_{x_j \in X} d(x_i, x_j), sinon. \end{cases}} \qquad \textbf{(8)}$$

Pour faciliter l'exposition, nous utilisons $\sigma(x_i)$ pour désigner l'indice j du point de données x_j qui est le plus proche de x_i et $\rho(x_j)>\rho(x_i)$, et si aucun tel point de données n'existe, $\sigma(x_i)$ est défini sur i, comme illustré dans l'équation suivante :

$$\boldsymbol{\sigma(x_i) = \begin{cases} \underset{j:\rho(x_j)>\rho(x_i)}{\mathbf{argmin}}\ \boldsymbol{d(x_i, x_j)}, \boldsymbol{si}\ \boldsymbol{\rho(x_i) < \max_{x_j \in X} \rho(x_j)} \\ \boldsymbol{i, sinon.} \end{cases}} \quad \textbf{(9)}$$

Notamment, il peut y avoir plus d'un point de données le plus proche de x_i et ayant une densité locale>$\rho(x_i)$. Selon l'implémentation Matlab de Laio de l'algorithme DPC, si cette situation se produit, alors $\sigma(x_i)$ est choisi au hasard parmi les index de ces points de données avec la densité locale la plus élevée parmi tous les points de données les plus proches de x_i et ont une densité locale>$\rho(x_i)$ [24].

7 Segmentation d'images multiobjectives

Un problème de segmentation d'images antérieur a été traité comme mono-objectif. Les images mono-objectives ne considèrent qu'un seul objectif, car une seule image de segmentation. Ce type d'images segmentées est de bonne qualité mais peut ne pas permettre à un processus de niveau supérieur (comme la segmentation d'images considérée comme un processus de bas niveau et une reconnaissance de modèle, un suivi d'objet et une analyse de scène comme un processus de haut niveau) d'extraire toutes les informations incluses dans l'image. Ainsi, différents résultats de segmentation sont calculés.

La segmentation d'images est un problème d'optimisation à objectifs multiples. La prise en compte de plusieurs critères (objectifs) commence par la compréhension du modèle d'image jusqu'au processus de segmentation sélectionné impliqué (sélection / extraction des caractéristiques, mesure de similarité / dissimilarité) et enfin l'évaluation de sa sortie (évaluation de la validité). Comme il existe des possibilités de sources multiples d'informations pour un problème de segmentation, de multiples représentations doivent donc être prises en compte. Par exemple, la sélection des caractéristiques est le processus d'identification des critères de similitude utilisés dans le processus de segmentation, maintenant soit un seul critère est utilisé, c'est à dire l'intensité de pixels, ou pour en faire un problème multi-objectif, considérer plusieurs critères de similitude pour segmenter la même

image, qui peut être l'intensité, la couleur, la texture, la forme, l'information spatiale. Par exemple, lors de la segmentation d'une image médicale basée sur la tomodensitométrie, plusieurs caractéristiques telles que l'intensité, la forme et la relation spatiale peuvent être prises en compte. De même, les critères de similitude inter-motifs qui se regroupent peuvent être multiples, cohérence spatiale vs homogénéité des caractéristiques, connectivité vs compacité, diversité vs précision. Pour la segmentation d'images, plusieurs méthodes peuvent être utilisées pour obtenir une sortie appropriée, et il peut y avoir une tendance à plusieurs optimisations et processus de prise de décision où une évaluation de validité multiple doit être utilisée.

Il existe deux approches générales pour le problème d'optimisations multi-objectives, la première approche consiste à combiner plusieurs fonctions objectives en une seule fonction composite, et la seconde consiste à déterminer un ensemble de solutions non dominées par rapport à chaque objectif.

a. L'approche de la formule pondérée conventionnelle (WFA)

Dans cette approche, un problème multi-objectif est transformé en un problème à objectif unique, qui consiste généralement à attribuer une pondération numérique à chaque objectif, puis à combiner les valeurs de critères pondérés en une valeur unique en ajoutant ou en multipliant des critères pondérés. La qualité d'un modèle de candidat donné est donnée par l'un des deux types de formule :

$$\boldsymbol{Q = w_1 c_1 + w_2 c_2 - - - - - - - - - - - - - + w_n c_n}$$

$$\boldsymbol{Q = w_1 c_1 \times w_2 c_2 - - - - - - - - - - - - - \times w_n c_n} \qquad \textbf{(10)}$$

$w_i, i = 1, 2, - - - - - - - - - - n$, désigne le poids attribué aux critères $c_i$et n est le nombre de critères d'évaluation.

b. Approche de Pareto (PTA)

L'idée de base est qu'au lieu de transformer un problème multi-objectif en une fonction objectif unique, puis de le résoudre en utilisant une méthode de recherche d'objectif unique, on utilise un algorithme multi-objectif pour résoudre le problème. La formulation commence par l'optimisation simultanée de plusieurs objectifs. Une solution raisonnable consiste à rechercher un ensemble de solutions satisfaisant chacun des objectifs à un niveau acceptable

sans être dominés par d'autres solutions. Ces solutions sont appelées solutions non dominées et la région de ces solutions est appelée front de Pareto.

8 Applications de la segmentation d'images

Avec les progrès technologiques en cours, la capacité d'analyser et d'extraire des informations utiles à partir d'images a ouvert une multitude d'opportunités dans divers secteurs. Les applications de segmentation d'image couvrent divers domaines, de l'imagerie médicale et des véhicules autonomes à la mode et à la vente au détail. Jetons un œil à certaines des plus courantes.

8.1 Imagerie médicale

La segmentation d'images est bénéfique dans ce domaine car elle analyse les images médicales (IRM, tomodensitométrie et rayons X) pour détecter des structures ou des irrégularités spécifiques pour le diagnostic et la planification du traitement. La segmentation d'images est également largement utilisée dans la recherche biomédicale pour le comptage des cellules, l'analyse des tissus et les études de structure anatomique.

8.2 Vidéosurveillance

Pour détecter et suivre les objets d'intérêt en temps réel, la segmentation d'images devient extrêmement bénéfique dans la vidéosurveillance, y compris les individus et les véhicules. En appliquant des techniques de segmentation d'images, les systèmes de vidéosurveillance peuvent facilement identifier et isoler les objets pertinents, offrant une surveillance plus précise. Elle permet également au personnel de sécurité d'agir en temps opportun sur les menaces potentielles ou les activités suspectes, améliorant ainsi l'efficacité globale de la surveillance et créant un environnement plus sûr.

8.3 Véhicules autonomes

La segmentation d'images permet aux véhicules autonomes de percevoir et de comprendre leur environnement, y compris les piétons, les autres véhicules et même les panneaux de signalisation. Elle est également utile pour prendre en charge des tâches telles que l'analyse du comportement, la reconnaissance d'objets et la détection d'anomalies.

8.4 Agriculture

L'intégration de la segmentation d'images dans l'agriculture implique la classification des terres et permet aux chercheurs et aux planificateurs de catégoriser et de comprendre différents types d'utilisation des terres. Ces informations sont essentielles à la planification urbaine, car elles fournissent des informations sur la répartition spatiale des infrastructures, des zones résidentielles et des paysages naturels.

8.5 Imagerie satellite

Elle aide à analyser les images satellite à diverses fins, notamment la classification de la couverture terrestre, la planification urbaine et la surveillance de l'environnement.

8.6 Robotique

La segmentation d'images joue un rôle essentiel en dotant les robots de la capacité de percevoir et d'interagir activement avec leur environnement. En utilisant cette technique, les robots parviennent à identifier et à distinguer avec précision les objets dans leur champ de vision. De plus, ils peuvent prendre des décisions plus éclairées et effectuer des tâches qui nécessitent la reconnaissance et la manipulation d'objets avec une compétence remarquable.

8.7 Art et design

Dans le domaine de l'art et du design, la segmentation d'images trouve son application comme moyen d'extraire et de manipuler des régions particulières d'une image. Ce processus offre aux artistes et aux concepteurs la possibilité de procéder à des modifications créatives et d'améliorer les effets visuels.

8.8 Jeux vidéo

L'intégration de la segmentation d'images dans les applications de jeu permet de détecter et de séparer les objets du jeu, de permettre aux personnages virtuels d'interagir avec l'environnement de jeu et d'enrichir l'expérience de jeu globale.

8.9 Mode et vente au détail

La segmentation d'images est un outil précieux pour des tâches telles que la reconnaissance d'objets, permettant l'identification et la catégorisation de divers produits de mode. De plus, la segmentation d'images contribue à la catégorisation efficace des produits, aidant les clients à

naviguer dans des inventaires étendus. Elle joue également un rôle crucial dans les expériences d'essayage virtuel, permettant aux clients de visualiser comment différents articles de vêtements apparaîtraient sur eux sans avoir besoin d'essayages physiques.

9 Conclusion

Grâce à la segmentation d'images, nous sommes désormais en mesure de détecter l'emplacement, la classe et les limites précises de tout objet donné dans les images et les vidéos. Deux catégories de segmentation d'images, la segmentation sémantique et la segmentation d'instance sont particulièrement utilisées. La dernière catégorie nous donne la possibilité d'explorer de nouvelles voies de segmentation précise et détaillée des variations au sein des images. Cela est particulièrement précieux pour de nombreux cas d'utilisation quotidiens allant de l'imagerie satellite à la vision artificielle dans l'automatisation agricole, etc.

La segmentation d'images, qui est devenue un point névralgique de la recherche dans le domaine du traitement d'images et de la vision par ordinateur, fait référence au processus de division d'une image en régions significatives et non superposées, et constitue une étape essentielle dans la compréhension de la scène naturelle. Malgré des décennies d'efforts et de nombreuses réalisations, l'extraction de caractéristiques et la conception de modèles posent encore des défis.

Chapitre II. Méthodes Classiques de Segmentation

1 Introduction

La segmentation est le processus consistant à diviser une image en un certain nombre de régions distinctes (qui ne se chevauchent pas) et à combiner ces régions pour former l'image globale [26]. C'est retrouver ses zones et ses contours uniformes. Ces régions et contours sont supposés être liés, c'est à dire que ces régions doivent correspondre aux parties significatives des objets réels et les contours à leurs contours apparents [27]. Une région est définie comme un ensemble de pixels connectés partageant une propriété commune qui les distingue des pixels des régions adjacentes [28]. Tous les pixels d'une région sont similaires en terme de caractéristiques ou de propriétés calculées, telles que la couleur, l'intensité ou la texture. Les régions adjacentes sont très différentes avec les mêmes caractéristiques [29].

La segmentation d'images est une tâche essentielle de la vision par ordinateur qui s'appuie sur le principe de la détection d'objets. Nous développerons plus en détail les similitudes et les différences importantes entre la segmentation d'images et d'autres tâches de vision par ordinateur, telles que la détection d'objets et bien d'autres, dans un instant. Avant d'en arriver là, il est nécessaire de déterminer en quoi consiste exactement la tâche de segmentation d'images. Comme son nom l'indique, la segmentation d'images consiste essentiellement à segmenter une image en fragments et à attribuer une étiquette à chacun d'eux. Cela se produit au niveau du pixel pour définir le contour précis d'un objet dans son cadre et sa classe. Ces contours, autrement appelés sorties, sont mis en évidence avec une ou plusieurs couleurs, selon le type de segmentation.

Récemment, les domaines d'application des images numériques se sont multipliés [28]. Le but de la segmentation est de simplifier la représentation d'une image ou de la transformer en une entité plus significative et plus facile à analyser [29].

La segmentation d'images est l'une des étapes clés de l'analyse d'image qui détermine la qualité des caractéristiques calculées ultérieurement dans le processus de compréhension de l'image. Cela vous permet de séparer les objets de l'image ou doit porter l'analyse et de séparer la zone d'intérêt de l'arrière-plan [26].

Autrement dit, si $\boldsymbol{I}$ est une image composée de $\boldsymbol{N}$ sous-ensembles $(\boldsymbol{I_1}, \boldsymbol{I_2}, \dots, \boldsymbol{I_N})$ formant une partition et $\boldsymbol{P}$ un prédicat d'uniformité, alors :

$\cup_{\boldsymbol{I}}^{\boldsymbol{N}} = \boldsymbol{I}$;

$\forall (i,j), i \neq j, I_i \cap I_j = \emptyset$;

$\forall I_i, P(I_i) = vrai$;

$\forall (i,j), I_i$ ***spatialement*** **adjacent à** $I_j, P(I_i \cup I_j) = faux$

Le choix d'une technique de segmentation dépend de plusieurs facteurs, notamment : Type d'image, conditions d'acquisition (bruit), éléments de base à extraire (contour, texture, etc.).

2 Méthodes de segmentation

Il existe plusieurs techniques pour réaliser la segmentation d'images, certaines étant classiques et d'autres moins traditionnelles, chacune proposant une approche unique pour atteindre le résultat final d'une image ou d'une vidéo. Parmi les techniques de segmentation d'images les plus courantes, on trouve notamment :

- Segmentation par région
- Segmentation par détection de contour
- Seuillage [Le seuillage est rarement utilisé comme solution de bout en bout pour la segmentation d'images], il s'agit généralement d'une étape de prétraitement. Le seuillage peut également être basé sur la région.
- Clustering
- Segmentation utilisant les outils de l'intelligence artificielle tels que le deep learning.

2.1 Segmentation par région

Cette approche consiste à décomposer l'image en différentes régions. Contrairement à l'approche contour, la méthode de cette approche s'intéresse au contenu de la région. Le diagramme ci-dessous [26] présente les techniques de segmentation par région les plus courantes.

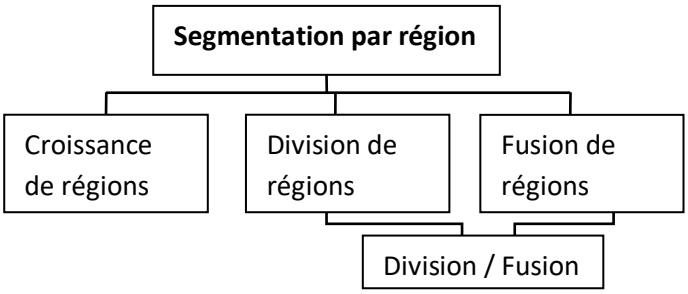


Figure 2 : Méthodes de segmentation par approche de région

2.1.1 Segmentation par croissance de régions (Region growing)

La méthode de croissance de régions a été initialement développée en 1968. Le principe de l'algorithme de type croissance de région est le suivant [30] :

- Définir un point (ou des points) de départ dans l'image. Ces points sont appelés les germes de la région de recherche.
- Définir des critères d'uniformité pour la zone cherchée (par exemple, valeur de gris ou critères de texture).
- Dans une procédure récursive (c'est à dire étape par étape), tous les points de connexion qui répondent aux critères sont inclus dans la région.

En conséquence, la région s'agrandit tant que les critères sont respectés.

A. Points de départ (germe)

La sélection du point de départ est une partie importante de l'algorithme. En fait, l'étape de croissance utilise une mesure de similarité pour sélectionner les pixels à regrouper. Si le point de départ se situe dans une région non uniforme, il y aura de grandes variations dans les mesures de similarité et la croissance s'arrêtera très rapidement [30].

B. Croissance de région (growing)

Cette étape vise à élargir la région en rassemblant les pixels voisins. Les pixels sont sélectionnés pour maintenir l'homogénéité de la région. Pour ce faire, nous devons définir un indice d'homogénéité. Si l'indicateur d'homogénéité reste vrai, les pixels adjacents sont ajoutés à la région. La croissance s'arrête lorsqu'aucun pixel ne peut être ajouté sans affecter l'homogénéité. L'algorithme de croissance de région est montré dans la figure 3 [30] suivante :

Créer une liste S de points de départ (triés de manière à ce que le centre du plus grand bloc apparaisse en premier).

Pour chaque pixel P de la liste S

Si un pixel P est déjà attribué à la région, alors obtenir le pixel P suivant dans la liste S

- Créer une nouvelle région R
- Ajouter le pixel P à la région R
- Calculer la valeur/couleur moyenne pour R
- Créer une liste m des pixels voisins du pixel P

Pour chaque pixel P_n de la liste m

Si (P_n n'est associé à aucune région et que $R + P_n$ est homogène), alors :

Ajouter le pixel P_n à la région R

Ajouter les pixels voisins P_n dans la liste m

Recalculer la valeur/couleur moyenne pour R

Fin Si

FinPour

Fin Pour

Figure 3 : Algorithme croissance de région

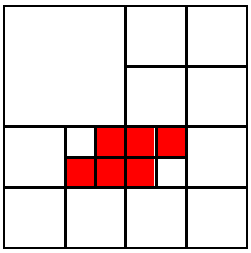
2.1.2 Segmentation par division de régions (Split)

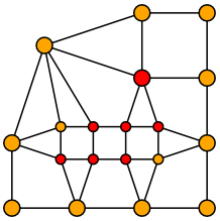
Selon un certain critère, la division consiste à diviser une image en régions homogènes. Le principe de cette technique est de considérer l'image elle-même comme une région de départ puis de la diviser en régions. Le processus de division est répété pour chaque nouvelle région jusqu'à ce que des classes homogènes soient obtenues comme le montre la figure 4.

Figure 4 : Segmentation par division de régions

2.1.3 Segmentation par fusion de régions (Merge)

Tous les nœuds du Region Adjacency Graph sont examinés. Si l'un des voisins de ce nœud est séparé de moins que le seuil de regroupement, les deux nœuds sont fusionnés au sein du RAG. Lorsqu'un nœud ne peut plus fusionner avec ses voisins, STOP.





La distance d'homogénéité d'une région est portée par l'arête valuée qui les relie au sein du RAG.

2.1.4 Segmentation par division-fusion (Split and Merge)

Il s'agit d'une combinaison de scissions et de fusions utilisant l'avantage des deux méthodes. Cette méthode est basée sur une représentation arborescente à quadrants de données dans laquelle le segment d'image est divisé en quatre quadrants à condition que les propriétés du segment d'origine ne soient pas uniformes. Ensuite, les quatre carrés voisins sont fusionnés en fonction de l'uniformité de la région (segments). Ce processus de division et de fusion se poursuit jusqu'à ce qu'aucune autre division ou fusion ne soit possible [31].

L'algorithme de division et de fusion suit les étapes suivantes :

1. Définir le critère d'homogénéité. Diviser l'image en quatre quadrants carrés.
2. Si un carré résultant n'est pas homogène, le diviser en quatre quadrants.
3. À chaque niveau, fusionner les deux ou plusieurs régions voisines satisfaisant la condition d'homogénéité.
4. Continuer la division et la fusion jusqu'à ce qu'aucune autre division ni fusion de région ne soit possible.

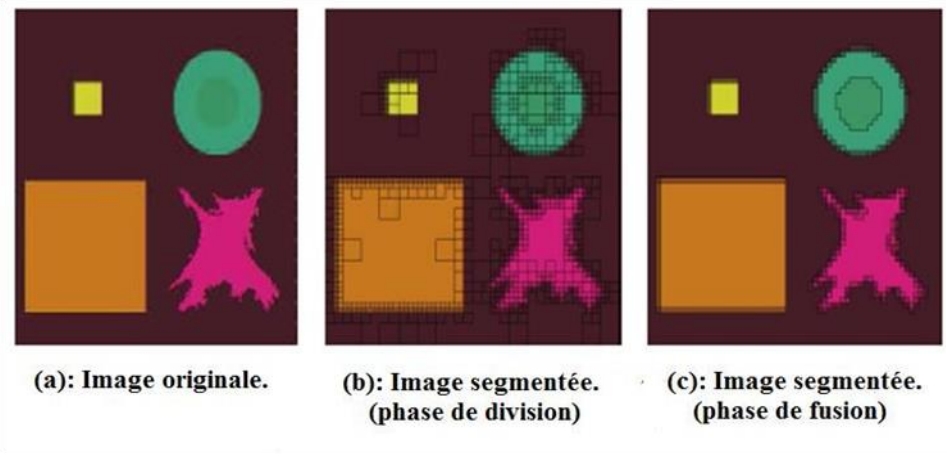


Figure 5 : Segmentation par (division/fusion) de régions

La figure 5 [26] illustre les deux étapes fondamentales de l'algorithme de segmentation par division et fusion (souvent appelé Split and Merge).

- ***Limites de segmentation par région***

La segmentation par région présente certains inconvénients [26], répertoriés ci-dessous :

- La limitation de la segmentation basée sur les régions est qu'il existe des risques de sous-segmentation et de sur-segmentation des régions de l'image. Cependant, ce problème peut être résolu de deux manières.
- En sélectionnant de manière optimale le critère de segmentation, plusieurs algorithmes utilisant des techniques d'intelligence artificielle ont été développés.
- En combinant une approche basée sur la région avec une approche basée sur les contours.
- Les régions résultantes ne correspondent pas toujours aux objets représentés dans l'image.
- Les limites des régions résultantes sont souvent inexactes et ne correspondent pas complètement aux limites des objets de l'image.
- Difficulté à définir des critères d'agrégation de pixels ou de fusion et division de régions.

2.2 Segmentation par clustering

Cette méthode fonctionne sur la base des pixels appartenant à une image, qui sont distingués et classés selon certaines exigences et règles, à l'aide d'un algorithme mathématique. Les pixels seront regroupés en un certain nombre de groupes en fonction de leur similitude de caractéristiques. L'image sera divisée en un certain nombre de groupes sur la base de la similitude des caractéristiques de ses pixels. Le regroupement est effectué à plusieurs reprises pour obtenir la similarité des pixels dans un cluster ou la différence maximale entre les clusters.

Il existe deux groupes de techniques de clustering, à savoir le clustering dur et le clustering soft. Le clustering dur met l'accent sur les différences entre les clusters, de sorte qu'un pixel ne sera joint que dans un cluster. En soft clustering, le regroupement se fait en fonction de plusieurs critères de similarité, comme la distance, la connectivité ou l'intensité, il est donc très possible de joindre plusieurs clusters.

Pour effectuer le regroupement d'images, nous devons d'abord choisir un espace de représentation, puis utiliser une mesure de séparation appropriée (mesure de proximité), pour coordonner les images et les points de groupe dans l'espace de représentation choisi. Le regroupement d'images est ensuite effectué selon un processus géré, en utilisant une intervention humaine ou selon un processus non supervisé, en fonction de la similitude entre les images et les différents points de groupe [32].

Les algorithmes de clustering les plus fréquemment employés incluent : Le clustering K-Means est une méthode de clustering populaire et largement utilisée. L'algorithme est simple, le processus est rapide et fonctionne en fonction de la proximité de la distance entre les éléments par rapport au centre du cluster, pour produire jusqu'à K données du cluster. Les calculs de distance sont effectués sur diverses propriétés qui constituent la base du regroupement. Avec un nombre prédéterminé de clusters allant jusqu'à K et un centre de cluster initial attribué aléatoirement, les pixels sont regroupés en fonction de leur proximité. Chaque cluster obtenu est ensuite calculé sur la moyenne, et cette valeur est utilisée comme centre du nouveau cluster. Ce processus est ensuite répété jusqu'à ce qu'il n'y ait plus de changements significatifs ou un certain nombre de répétitions [33].

Il existe également l'algorithme le plus admiré dans le domaine du clustering flou qui est l'algorithme Fuzzy C-Means (FCM). Le Fuzzy C-Means (FCM) est une technique de clustering qui permet à une donnée d'appartenir à deux ou plusieurs clusters [32].

DBSCAN (Density Based Spatial Clustering of Applications with Noise) est également un algorithme de clustering très largement utilisé qui construit des clusters à partir d'un seul point et de ses voisins. L'idée clé est de regrouper les régions de haute densité en un seul cluster. Cette approche permet à la méthode de fournir des résultats avec des formes de clusters arbitraires, de détecter le bruit des clusters réels et d'être efficace pour les grandes bases de données spatiales [34].

L'algorithme de clustering basé sur la densité joue un rôle essentiel dans la recherche de structures de formes non linéaires basées sur la densité. Dans le clustering basé sur la densité, les clusters sont définis comme des zones de densité supérieure à celle du reste de l'ensemble de données. Le concept derrière l'algorithme basé sur la densité est l'accessibilité de la densité et la connectivité de la densité [35].

La principale caractéristique de cet algorithme est la découverte de clusters de forme arbitraire et la capacité de gérer les données de bruit en une seule analyse. Les nombreuses études intéressantes sur l'algorithme basé sur la densité sont DBSCAN, GDBSCAN, OPTICS, DENCLUE et CLIQUE. Les deux paramètres globaux de la densité sont Eps : rayon maximal du voisinage et $MinPts$: nombre minimal de points dans un voisinage Eps de ce point.

L'idée du clustering hiérarchique est de développer une hiérarchie de requêtes s'adressant à l'ensemble établi d'exemples (pour l'image, appelés pixels) et aux niveaux de comparabilité auxquels les regroupements changent. Nous pouvons appliquer la collecte d'informations bidimensionnelle pour traduire le fonctionnement du calcul de regroupement progressif [32].

2.3 Segmentation par techniques d'intelligence artificielle

L'intelligence artificielle (IA) consiste à rendre les machines aussi intelligentes que le cerveau humain. En informatique, l'IA fait référence à l'étude des « agents intelligents » : C'est un appareil qui reconnaît l'environnement et prend des mesures qui maximisent les chances de succès dans la réalisation des objectifs. De manière informelle, le terme «intelligence artificielle» est utilisé lorsque les machines peuvent exécuter des fonctions que les humains associent à l'esprit humain, telles que «l'apprentissage» et la «résolution de

problèmes». L'apprentissage est un élément essentiel des machines. L'apprentissage automatique est donc un sous-domaine de l'IA (Figure 6) [36]. Les informaticiens travaillent dans le domaine de l'apprentissage automatique depuis les années 1950. Au cours des dernières décennies, des efforts importants ont été déployés pour faire progresser l'apprentissage automatique. Cela conduit à des attentes plus élevées à l'égard des machines. L'apprentissage profond est une tentative dans ce sens. L'apprentissage profond est un sous-domaine de l'apprentissage automatique (Figure 6) qui structure les algorithmes en couches pour créer des réseaux neuronaux artificiels capables d'un apprentissage autonome et d'une prise de décision intelligente. L'apprentissage progresse dans de nombreux nouveaux domaines, et l'applicabilité de ces nouveaux domaines demeure un défi permanent dans la communauté de la recherche.

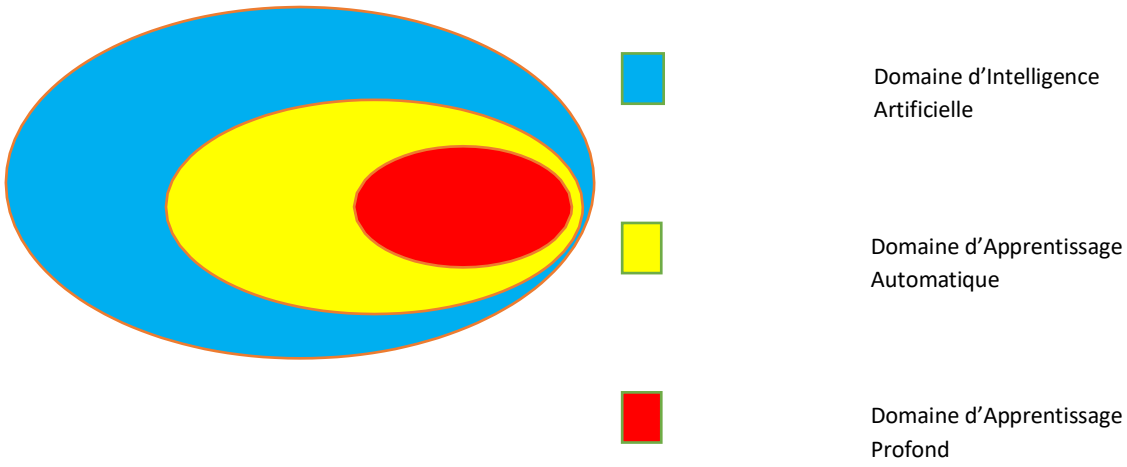


Figure 6 : Présentation sur la relation entre l'apprentissage automatique et l'apprentissage profond

Les évolutions technologiques, notamment dans les domaines de l'intelligence artificielle et des algorithmes d'apprentissage automatique (machine learning et deep learning), ont apporté de nombreux bénéfices. Récemment, plusieurs algorithmes tels que les machines à vecteurs de support, les réseaux de neurones et les forêts aléatoires ont été utilisés avec succès pour des applications de télédétection. Ce processus traduit les connaissances humaines en intelligence artificielle de manière plus sophistiquée [36].

Bien que de bons résultats aient été obtenus en traitant les données de télédétection à l'aide de diverses techniques d'intelligence artificielle et d'apprentissage automatique, la sélection et l'évaluation de l'efficacité de chaque technique restent une tâche difficile et laborieuse.

En 2017, Litjens et al. [37] ont publié une revue très complète d'articles décrivant l'utilisation d'outils d'apprentissage profond (IA) pour analyser des images, résumant plus de 300 références [38]. Ils ont examiné l'utilisation de l'apprentissage profond pour la classification d'images, la détection d'objets, la segmentation, l'enregistrement et d'autres tâches. Ils ont mis l'accent sur l'utilisation des algorithmes d'apprentissage profond, en particulier les réseaux convolutifs qui sont rapidement devenus une méthodologie de choix pour l'analyse des images médicales.

Initialement, la segmentation d'images a été développée à l'aide de modèles mathématiques (tels que les techniques de Fuzzy C-mean clustering ou de K-means clustering) ou de processus d'analyse de pixels de bas niveau (tels que la croissance de régions et la détection de contours) [38]. Depuis le début du 21e siècle, les outils d'IA sont utilisés pour effectuer la segmentation. Par conséquent, des réseaux de neurones ont été développés qui nécessitent une phase d'apprentissage visant à convertir les images d'entrée en étiquettes (c'est à dire en régions segmentées).

L'apprentissage profond (Deep Learning) et le traitement d'images sont deux domaines qui intéressent beaucoup les universitaires et les professionnels de l'industrie. Les domaines d'application de ces deux domaines sont très différents et incluent des domaines tels que la médecine, la robotique, la sécurité et la surveillance.

Le Deep Learning n'a aucun impact négatif sur le traitement des images. L'apprentissage profond nécessite d'énormes ensembles de données et de grandes quantités de ressources informatiques. Il existe de nombreuses applications pour lesquelles il est souhaitable de réduire la charge de calcul et les besoins en mémoire, et de pouvoir traiter des images sans accéder à de grandes bases de données. Les exemples incluent les téléphones mobiles, les tablettes, les appareils photo mobiles et les voitures. L'apprentissage profond connaît actuellement du succès et peut donner de très bons résultats de classification.

La segmentation d'images est l'un des nombreux problèmes de traitement d'images traités par le Deep Learning.

L'intelligence artificielle (IA) a révolutionné de nombreux secteurs, mais l'un des domaines dans lesquels l'IA a fait des progrès significatifs est celui de la segmentation d'images. La segmentation d'images est le processus de division d'une image en plusieurs segments ou régions pour simplifier sa représentation et faciliter l'analyse. Cette technologie a

de nombreuses applications dans divers domaines, notamment la médecine, les voitures autonomes et la robotique. Cependant, cela comporte également des défis.

Les techniques de segmentation d'images IA ont évolué au fil des années, les algorithmes d'apprentissage en profondeur jouant un rôle clé dans la production de résultats précis. Les réseaux de neurones convolutifs (CNN) sont largement utilisés dans les tâches de segmentation d'images car ils peuvent apprendre des représentations hiérarchiques. CNN analyse les images à différents niveaux d'abstraction, permettant une identification et une classification précises des objets présents dans une image.

Une technique courante de segmentation d'images d'IA est la segmentation sémantique. Cette dernière attribue une étiquette à chaque pixel d'une image, lui permettant d'identifier différents objets et régions. Cette technologie est particulièrement utile dans les voitures autonomes, car elle les aide à identifier les piétons, les véhicules et autres obstacles sur la route.

Une autre technique est la segmentation d'instance. Cela identifie non seulement les objets, mais distingue également les différentes instances du même objet. Par exemple, dans une image encombrée, la segmentation des instances peut distinguer plusieurs individus même s'ils appartiennent à la même classe. Cette technologie est utilisée dans les systèmes de surveillance et de suivi d'objets.

La segmentation d'images IA trouve des applications importantes dans l'industrie médicale. La segmentation des images médicales joue un rôle important dans le diagnostic et le traitement de diverses maladies. Par exemple, la segmentation des tumeurs et des lésions à partir d'images médicales en radiologie permet de déterminer avec précision leur taille, leur forme et leur emplacement. Ces informations facilitent la planification du traitement et le suivi de la progression de la maladie.

Dans le domaine de la robotique, la segmentation d'images par l'IA permet aux robots de percevoir et d'interagir efficacement avec leur environnement. La segmentation d'objets en temps réel permet aux robots d'identifier et de manipuler une variété d'objets, les rendant plus polyvalents et capables d'effectuer des tâches complexes. Cette technologie a le potentiel de révolutionner des secteurs tels que la fabrication et la logistique.

Malgré l'immense potentiel de la segmentation d'images par l'IA, elle est également confrontée à plusieurs défis. L'un des plus grands défis est qu'il nécessite une grande quantité de données d'entraînement étiquetées. La formation de modèles d'apprentissage profond pour la segmentation d'images nécessite de grandes quantités d'images annotées, dont l'acquisition peut prendre du temps et être coûteuse. De plus, la qualité et la diversité des données d'entraînement affectent directement la précision et la capacité de généralisation du modèle.

Un autre défi réside dans le compromis entre précision et efficacité informatique. Les modèles d'apprentissage profond pour la segmentation d'images peuvent nécessiter beaucoup de calculs, nécessitant un matériel puissant et un temps de traitement important. Il est important d'équilibrer précision et efficacité, en particulier dans les applications en temps réel telles que les voitures autonomes, où une prise de décision rapide est essentielle.

De plus, la segmentation des images IA peut être difficile à gérer avec des scènes complexes, des occultations et des conditions d'éclairage variables. Ces défis peuvent entraîner des erreurs dans la segmentation précise des objets et affecter les performances globales du système. Pour surmonter ces limites, il faut poursuivre la recherche et le développement dans le domaine de la segmentation des images par l'IA.

3 Conclusion

La segmentation a bénéficié des résultats de recherche dans le domaine de clustering des données tiré des avancées réalisées en mathématique et en physique ainsi que des avancées récentes en IA. En effet, la technologie de segmentation d'images par IA a considérablement progressé ces dernières années, permettant une analyse d'images précise et efficace. Des soins de santé à la robotique, les applications de la segmentation d'images par l'IA sont diverses et prometteuses. Cependant, pour exploiter tout le potentiel de cette technologie, des défis tels que le besoin de données d'entraînement étiquetées, l'efficacité informatique et la gestion de scènes complexes doivent être relevés. Grâce à la recherche et à l'innovation en cours, la segmentation d'images par IA est sur le point de révolutionner une variété d'industries et d'ouvrir la voie à un avenir où les machines pourront percevoir et comprendre les informations visuelles tout comme les humains.

Chapitre III. Techniques de Clustering

1 Introduction

La tâche la plus illustrative dans le processus d'exploration de données est le clustering. Il joue un rôle extrêmement important dans l'ensemble du processus de data mining, car la catégorisation des données est l'une des étapes les plus rudimentaires de la découverte de connaissances. Il s'agit d'une tâche d'apprentissage non supervisée utilisée pour l'analyse exploratoire des données afin de trouver des modèles non révélés qui sont présents dans les données mais ne peuvent pas être catégorisés clairement. Les ensembles de données peuvent être désignés ou regroupés en fonction de certaines caractéristiques communes et appelés clusters. Le mécanisme impliqué dans l'analyse de cluster dépend essentiellement de la tâche principale consistant à garder les objets d'un cluster plus proches que les objets appartenant à d'autres groupes ou clusters.

En fonction des données et des caractéristiques de cluster attendues, il existe différents types de paradigmes de clustering. Ces derniers temps, de nombreux nouveaux algorithmes ont émergé qui visent à relier les différentes approches du clustering et à fusionner différents algorithmes de clustering compte tenu de l'exigence de gérer des données séquentielles et étendues avec de multiples relations dans de nombreuses applications à travers un large spectre. Divers algorithmes de clustering ont été développés sous différents paradigmes pour regrouper des points de données dispersés et former des formes de cluster efficaces avec un minimum de valeurs aberrantes.

Le clustering signifie la division des données en groupes (appelés clusters) de telle manière que les objets appartenant à un groupe sont similaires les uns aux autres, mais ils sont différents des objets appartenant à d'autres groupes [2] [39]. Il s'agit d'un processus important dans la reconnaissance de formes et l'apprentissage automatique. De plus, le clustering de données est un processus central en intelligence artificielle (IA). Les algorithmes de clustering sont utilisés dans de nombreuses applications, telles que la segmentation d'images, quantification d'images vectorielles et couleur, fouille de données, compression, apprentissage automatique, etc. Un cluster est généralement identifié par un centre de cluster (ou centroïde). Le clustering de données est un problème difficile dans la reconnaissance de formes non supervisée, car les clusters peuvent avoir des formes et des tailles différentes [40] [41]. Les algorithmes de clustering basés sur la densité regroupent les objets de données en fonction de la densité entre les objets [39].

Le clustering regroupe un ensemble de techniques visant à regrouper les enregistrements d'une base de données en groupes en fonction de leur proximité les uns par rapport aux autres, sans s'appuyer sur des informations préalables. C'est un apprentissage non supervisé.

2 Techniques de clustering

Parmi les techniques de clustering les plus connues, on retrouve :

2.1 L'algorithme k-means

Le clustering K-means est un algorithme de data mining / d'apprentissage automatique utilisé pour regrouper les observations en groupes d'observations connexes sans aucune connaissance préalable de ces relations. L'algorithme k-means est l'une des techniques de clustering les plus simples et il est couramment utilisé en imagerie médicale, en biométrie et dans des domaines connexes [42].

L'algorithme k-means est un algorithme évolutif qui tire son nom de sa méthode de fonctionnement. L'algorithme regroupe les observations en k groupes, où k est fourni comme paramètre d'entrée. Il attribue ensuite chaque observation à des clusters en fonction de la proximité de l'observation avec la moyenne du cluster. La moyenne du cluster est ensuite recalculée et le processus recommence. Voici comment fonctionne l'algorithme :

1. L'algorithme sélectionne arbitrairement k points comme centres de cluster initial «means».
2. Chaque point de l'ensemble de données est affecté au cluster fermé, en fonction de la distance euclidienne entre chaque point et chaque centre de cluster.
3. Chaque centre de cluster est recalculé comme la moyenne des points de ce cluster.
4. Les étapes 2 et 3 se répètent jusqu'à ce que les clusters convergent. La convergence peut être définie différemment selon l'implémentation, mais cela signifie normalement qu'aucune observation ne change de cluster lorsque les étapes 2 et 3 sont répétées ou que les changements ne font pas de différence significative dans la définition des clusters.

2.2 L'algorithme Fuzzy C-Means

Fuzzy C-Mean (FCM) est un algorithme de clustering non supervisé qui a été appliqué à un large éventail de problèmes impliquant l'analyse des fonctionnalités, le clustering et la conception du classificateur.

FCM a un large domaine d'applications telles que le génie agricole, l'astronomie, la chimie, la géologie, l'analyse d'images, le diagnostic médical, l'analyse de forme et la reconnaissance de cibles [43]. Avec le développement de la théorie du flou, l'algorithme de clustering fuzzy c-means basé sur la théorie du clustering flou de Ruspini a été proposé dans les années 1980. Cet algorithme est examiné pour analyser en fonction de la distance entre les différents points de données d'entrée. Les clusters sont formés en fonction de la distance entre les points de données et les centres de cluster sont formés pour chaque cluster. La structure de base de l'algorithme FCM est discutée ci-dessous. L'Algorithme Fuzzy C-Means (FCM) est une méthode de clustering qui permet à un élément de données d'appartenir à deux clusters ou plus. Cette méthode est fréquemment utilisée dans la reconnaissance de formes. Il est basé sur la minimisation de la fonction objectif suivante :

$$\boldsymbol{J_m = \sum_{i=1}^{N}\sum_{j=1}^{C} u_{ij}^{m}\left\|x_i - c_j\right\|^2 \quad , \quad 1 \le m < \infty} \qquad \textbf{(11)}$$

Où m est un nombre réel supérieur à 1, u_{ij} est le degré d'appartenance à x_i dans le cluster j, x_i est le $i^{ème}$ de données mesurées en dimension d, c_j est le centre de dimension d du cluster, et $\|x_i - c_j\|$ est toute norme exprimant la similitude entre les données mesurées et le centre. Le partitionnement flou est réalisé par une optimisation itérative de la fonction objectif présentée ci-dessus, avec la mise à jour de l'adhésion u_{ij} et les centres de cluster c_j par :

$$\boldsymbol{u_{ij} = \frac{1}{\sum_{i=1}^{C}\left(\frac{\left\|x_i - c_j\right\|}{\left\|x_i - c_k\right\|}\right)^{\frac{2}{m-1}}} \quad c_j = \frac{\sum_{i=1}^{N} u_{ij}^{m} x_i}{\sum_{i=1}^{N} u_{ij}^{m}}} \qquad \textbf{(12)}$$

Cette itération va s'arrêter lorsque $\max_{ij}\left\{\left|u_{ij}^{(k+1)} - u_{ij}^{(k)}\right|\right\} < \delta$, où ξ est un critère de terminaison compris entre 0 et 1, tandis que k est les étapes de l'itération. Cette procédure converge vers un minimum local ou un point de selle de J_m . L'algorithme est composé des étapes suivantes :

Étape 1 : Initialiser la matrice $U = [u_{ij}]$, $U^{(0)}$.

Étape 2 : A k-étape : calculer les vecteurs centres $C^{(k)} = [c_j]$ avec $U^{(k)}$.

Étape 3 : Mettre à jour $U^{(k)}$, $U^{(k+1)}$.

Étape 4 : Si $\|U^{(k+1)} - U^{(k)}\| < \xi$ alors ARRÊT ;

Sinon, revenir à l'étape 2.

Dans cet algorithme, les données sont liées à chaque cluster au moyen d'une fonction d'appartenance, qui représente le comportement flou de l'algorithme. Pour ce faire, l'algorithme doit construire une matrice appropriée nommée U dont les facteurs sont des nombres compris entre 0 et 1, et représentant le degré d'appartenance entre les données et les centres de clusters. Les techniques de clustering FCM sont basées sur un comportement flou et fournissent une technique naturelle pour produire un clustering où les poids d'appartenance ont une interprétation naturelle (mais non probabiliste). Cet algorithme a une structure similaire à l'algorithme K-Means et se comporte également de manière similaire.

2.3 Techniques de clustering basées sur la densité

Le clustering basé sur la densité fait référence à des méthodes d'apprentissage non supervisées qui identifient des groupes/clusters distinctifs dans les données, sur la base de l'idée qu'un cluster dans un espace de données est une région contiguë de densité de points élevée, séparée des autres clusters par des régions contiguës de faible densité de points. Les points de données dans les régions de séparation à faible densité de points sont généralement considérés comme du bruit/des valeurs aberrantes.

L'algorithme de clustering basé sur la densité joue un rôle essentiel dans la recherche de structures de formes non linéaires basées sur la densité. Dans le clustering basé sur la densité, les clusters sont définis comme des zones de densité supérieure à celle du reste de l'ensemble de données [35].

a. L'algorithme DBSCAN (Density Based Spatial Clustering of Applications with Noise)

En tant que l'un des algorithmes de clustering basés sur la densité les plus cités, DBSCAN est probablement l'algorithme de clustering basé sur la densité le plus connu dans la communauté scientifique aujourd'hui. L'idée centrale derrière DBSCAN et ses extensions et révisions est la notion selon laquelle les points sont attribués au même cluster s'ils sont accessibles en densité les uns des autres [13]. Pour comprendre ce concept, nous passerons en revue les définitions les plus importantes utilisées dans DBSCAN et les algorithmes associés.

Le clustering commence avec un ensemble de données D contenant un ensemble de points p ∈ D. Les algorithmes basés sur la densité doivent obtenir une estimation de la densité sur l'espace de données. DBSCAN estime la densité autour d'un point en utilisant le concept de ϵ -voisinage.

Définition 1. ϵ-voisinage. Le ϵ -voisinage, $N_\epsilon(p)$, d'un point de données p est l'ensemble des points dans un rayon spécifié ϵ autour de p.

$$\boldsymbol{N_\epsilon(p) = \{q | d(p,q) \leq \epsilon\}} \qquad \textbf{(13)}$$

où d est une mesure de distance et $\epsilon \in \mathbb{R}^+$. Noter que le point p est toujours dans son propre ϵ-voisinage, c'est à dire $p \in N_\epsilon(p)$ tient toujours.

Suivant cette définition, la taille du voisinage $|N_\epsilon(p)|$ peut être considérée comme une simple estimation non normalisée de la densité du noyau autour de p en utilisant un noyau uniforme et une bande passante de ϵ.

DBSCAN utilise $N_\epsilon(p)$ et un seuil appelé $MinPts$ pour détecter les régions denses et pour classer les points d'un ensemble de données en points de noyau, de bordure ou de bruit.

Définition 2. Classes de points. Un point $p \in D$ est classé comme

- un point central si $N_\epsilon(p)$ a une densité élevée, c'est à dire $|N\_\epsilon\ (p)| \geq MinPts$ où $MinPts \in \mathbb{Z}^+$ est un seuil de densité spécifié par l'utilisateur,
- un point frontière si p n'est pas un point central, mais il est au voisinage d'un point central $q \in D$ c'est à dire $p \in N_\epsilon(q)$, ou
- un point de bruit, sinon.

Pour former des régions denses contiguës à partir de points individuels, DBSCAN définit les notions d'accessibilité et de connectivité.

Définition 3. Directement accessible en densité. Un point $q \in D$ est directement accessible en densité à partir d'un point $p \in D$ par rapport à ϵ et $MinPts$ si, et seulement si,

1. $|N_\epsilon(p)| \geq MinPts$, et
2. $q \in N_\epsilon(p)$.

Autrement dit, p est un point central et q est dans son voisinage ϵ.

Définition 4. Densité-accessible. Un point p est accessible en densité à partir de q s'il existe dans D une suite ordonnée de points $(p1, p2, \ldots, pn)$ avec $q = p_1$ et $p = p_n$ tel que $p_i + 1$ directement accessible à partir de la densité $p_i \forall i \in \{1,2, \ldots, n-1\}$.

Définition 5. Connecté Densité. Un point $p \in D$ est la densité connectée à un point $q \in D$ s'il y a un point $o \in D$ de telle sorte que p et q soient tous deux accessibles en densité à partir de o.

La notion de connexion de densité peut être utilisée pour former des groupes en tant que régions denses contiguës.

Définition 6. Cluster. Un cluster C est un sous-ensemble non vide de D satisfaisant les conditions suivantes :

1. Maximalité : Si $p \in C$ et q est la densité accessible à partir de p, alors $q \in C$; et
2. Connectivité: $\forall p, \in C$, p est lié à la densité de q.

L'algorithme DBSCAN identifie tous ces clusters en trouvant tous les points centraux et en étendant chacun à tous les points de densité atteignables. L'algorithme commence par un point arbitraire p et récupère son ϵ-voisinage. S'il s'agit d'un point central, il démarrera un nouveau cluster qui sera développé en attribuant tous les points de son voisinage au cluster. Si un point central supplémentaire est trouvé dans le voisinage, la recherche est étendue pour inclure également tous les points de son voisinage. Si aucun autre point central n'est trouvé dans le voisinage étendu, le cluster est terminé et les points restants sont recherchés pour voir si un autre point central peut être trouvé pour démarrer un nouveau cluster. Après le traitement de tous les points, les points qui n'ont pas été affectés à un cluster sont considérés comme du bruit.

Dans l'algorithme DBSCAN, les points principaux font toujours partie du même cluster, indépendamment de l'ordre dans lequel les points de l'ensemble de données sont traités. Ceci est différent pour les points frontaliers. Les points frontières peuvent être accessibles en densité à partir des points centraux de plusieurs clusters et l'algorithme les affecte au premier de ces clusters traités, ce qui dépend de l'ordre des points de données et de l'implémentation particulière de l'algorithme. Pour atténuer ce comportement, Campello et al. (2015) suggèrent une modification appelée DBSCAN* qui considère à la place tous les points de frontière comme du bruit et les laisse non attribués.

b. L'algorithme DPC

L'algorithme de clustering par pic de densité (DPC) est un nouvel algorithme de clustering proposé en 2014 [25]. La clé de l'algorithme DPC est de dessiner le graphe de décision en fonction des caractéristiques du centre du cluster des données et d'identifier rapidement le centre du cluster par le graphe de décision. Le centre du cluster défini dans le clustering par pic de densité repose sur deux hypothèses et constitue également une méthode permettant d'identifier le centre du cluster. Tout d'abord, le centre du cluster est entouré de points de données voisins dont la valeur de densité ne le dépasse pas, c'est à dire que la densité locale du centre du cluster est relativement plus grande. Deuxièmement, l'emplacement du centre du cluster est plus éloigné des autres points de données ayant une densité relativement élevée. Étant donné que la base de l'algorithme est de calculer la distance entre les points de données, pour chaque point de données, nous devons calculer deux valeurs de paramètres : la densité locale ρ_i du point et de la distance δ_i à partir du point de densité la plus élevée. En calculant la similarité d_{ij} entre chaque point de données, la matrice de similarité est construite. Supposons qu'un ensemble de données soit $X = \{x_1, x_2, \dots, x_n\}$ et la taille de l'ensemble de données est n, et nous utilisons la distance euclidienne pour trouver la similitude entre deux points :

$$\boldsymbol{d_{ij} = \left\| x_i - x_j \right\|_2} \qquad \textbf{(14)}$$

En fonction de la similarité obtenue, la matrice de similarité est construite.

$$D = [d_1, d_2, \dots, d_n]^T, D \in R_{n \times n}$$

Dans l'équation ci-dessus, d_i est la similitude entre un point et un autre point, ce qui signifie $d_i = [d_{i1}, d_{i2}, \dots, d_{in}]$.

D'après la définition de l'hypothèse, les deux premiers paramètres dépendent de d_{ij} , afin de satisfaire l'inégalité triangulaire suivante et de calculer la valeur de densité locale ρ_i du point de données :

$$\boldsymbol{\rho_i = \sum_j X(d_{ij} - d_c)} \qquad \textbf{(15)}$$

Où :

$$X(x) = \begin{cases} 1 & x < 0 \\ 0 & x \geq 0 \end{cases}$$

Où d_c est la distance limite, la valeur de d_c devrait être défini empiriquement dans des expériences sur différents ensembles de données. Il est généralement défini sur la valeur à laquelle la similarité globale de l'ensemble de données est de 2 % comme distance limite. En

général, la valeur de ρ_i est fondamentalement égal à la somme du nombre de points de données inférieurs à d_c loin du point cible. Par conséquent, pour les ensembles de données à volume important, les résultats de l'analyse sont robustes au paramètre de d_c dans une certaine mesure. Pour les ensembles de données plus petits, la fonction de noyau gaussien suivante peut être utilisée :

$$\boldsymbol{\rho_i = \sum_j exp\left(-\frac{d_{ij}^2}{d_c^2}\right)} \qquad \textbf{(16)}$$

Après avoir obtenu la densité locale, l'autre paramètre clé est d'obtenir la distance relative δ_i. L'équation pour obtenir la distance relative ici est la suivante :

$$\boldsymbol{\delta_i = \begin{cases} min(d_{ij})_j & j: \rho_j > \rho_i \\ max(d_{ij})_j & otherwise \end{cases}} \qquad \textbf{(17)}$$

L'équation ci-dessus peut calculer la distance relative entre la densité locale supérieure au point de données et le point le plus proche, qui est le processus de clustering. Dans le cas contraire, la distance maximale est obtenue.

2.4 Clustering hiérarchique

L'idée du clustering hiérarchique est de développer une demande hiérarchique s'adressant à l'ensemble établi d'exemples (pour l'image, appelés pixels) et aux niveaux de comparabilité auxquels les regroupements changent. Nous pouvons appliquer la collecte informationnelle bidimensionnelle pour traduire le fonctionnement du calcul de groupement progressif [32]. Étant donné un ensemble de N objets à regrouper et une matrice de distance (ou de similarité) $N * N$, la procédure essentielle du clustering hiérarchique est la suivante :

1. Commencer par tout distribuer à un groupe, de sorte que dans le cas où vous avez N choses, vous avez maintenant N groupes, chacun contenant une seule chose. Laisser les séparations (similitudes) entre les groupes être les mêmes que les séparations (ressemblances) entre les choses qu'ils contiennent.
2. Trouver la combinaison de groupes la plus proche (la plus comparative) et les réunir en un seul groupe, de sorte qu'avoir maintenant un groupe de moins.
3. Calculer les séparations (similitudes) entre le nouveau groupe et chacun des anciens groupes.

4. Répéter les étapes 2 et 3 jusqu'à ce que tous les éléments soient regroupés dans un seul groupe de taille N.

2.5 Réseaux de Neurones Artificiels (ANN)

En tant que l'une des méthodes d'apprentissage automatique, l'ANN a montré son succès dans la résolution des problèmes de prédiction, de classification et d'association. Le type d'ANN le plus courant est le perceptron multicouche (MLP) suggéré par Werbos en 1974 et amélioré par Rumelhart et al en 1995 [44].

La période de 1943 à 1949 est celle des premières tentatives de réseaux neuronaux. L'émergence de ce domaine a été marquée par le développement des premiers modèles de réseaux neuronaux en 1943. L'année entre 1969 et 1974 est une période où le développement des réseaux neuronaux a connu le plus de difficultés et de revers lors de l'exploitation de ce nouveau domaine. Ce n'est qu'à partir des années 1980 que les réseaux neuronaux ont connu une innovation. Nous vivons aujourd'hui une ère de transformation pour les réseaux neuronaux et ce domaine a fait des progrès significatifs. Par exemple, l'équipe de recherche Schmidhuber a inventé les réseaux neuronaux à propagation directe profonde et les réseaux neuronaux récurrents [45].

Un réseau neuronal est un ensemble de nœuds, d'unités et d'éléments de traitement simples interconnectés dont la fonction est approximativement basée sur les neurones d'un animal. En d'autres termes, il s'agit d'un algorithme d'apprentissage automatique qui imite la façon dont les neurones biologiques transmettent des signaux entre eux.

Dans les domaines de la classification, du clustering, de la reconnaissance de modèles et de la prédiction, les réseaux neuronaux ont été largement utilisés, ce qui permet aux réseaux neuronaux de fournir un soutien efficace pour traiter des problèmes complexes et non complexes dans de nombreux domaines, ce qui contribue grandement à l'efficacité du travail des industries et facilite également la vie des gens.

L'approche du clustering par réseau neuronal artificiel (ANN) présente des liens théoriques étroits avec le traitement cérébral réel. Il est nécessaire de le rendre plus puissant et plus polyvalent dans les bases de données volumineuses en raison des longues circonstances de préparation et de la complexité des informations complexes [32].

Les réseaux de neurones artificiels constituent la base de l'apprentissage automatique et de l'intelligence artificielle. Ils se déclinent en plusieurs types, chacun étant adapté à des tâches spécifiques. Voici un aperçu des principaux types :

a. Feedforward Neural Networks (FNN)

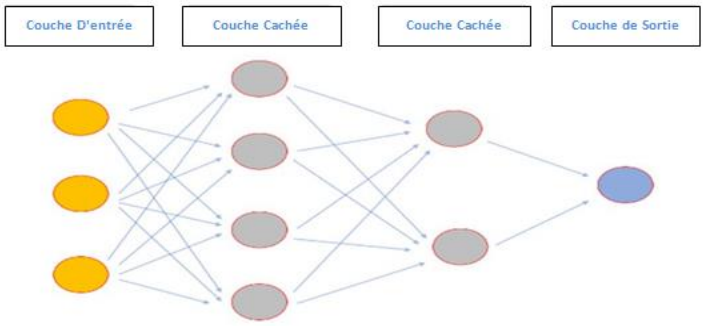


Figure 7 : Feedforward : des réseaux neuronaux à propagation directe

Comme le montre la figure 7 [45], l'un des modèles de réseaux neuronaux les plus basiques est un réseau neuronal à propagation directe (FNN), également connu sous le nom de perceptron multicouche (MLP). Il est formé avec des algorithmes d'apprentissage par rétropropagation et fait partie des réseaux neuronaux les plus populaires disponibles aujourd'hui. Il se compose de plusieurs neurones, chacun recevant la sortie des neurones de la couche précédente, et est pondéré et traité par certains poids et biais pour finalement obtenir la sortie des neurones de la couche actuelle, qui sert à son tour d'entrée aux neurones de la couche suivante. Le flux d'informations du réseau est unidirectionnel et ne peut se déplacer que de la couche d'entrée vers la couche de sortie, c'est pourquoi on l'appelle un réseau neuronal à rétroaction.

Le réseau neuronal à propagation directe est généralement composé d'une couche d'entrée, de plusieurs couches cachées et d'une couche de sortie, comme illustré dans la figure 7, lorsque la couche d'entrée reçoit l'entrée de données d'origine, la couche cachée la transforme et extrait la caractéristique dans une certaine mesure, et les résultats de la couche cachée sont utilisés pour prédire la valeur du modèle dans la couche de sortie. Chaque couche se compose de plusieurs neurones, et les connexions entre les neurones sont pondérées, ce qui peut être appris à optimiser grâce à des algorithmes de rétropropagation. Les réseaux neuronaux à propagation directe sont des modèles puissants capables de gérer des tâches de classification et de régression non linéaires.

FNN est un réseau neuronal artificiel dans lequel les nœuds ne forment pas de boucles. Le FNN qui a des couches cachées peut être formé pour effectuer et résoudre tous les problèmes de classification et de régression. Les cas d'application incluent la classification d'images et la notation de crédit.

Classification d'image : le nœud de la couche d'entrée est déterminé par la valeur du pixel de l'image dans la tâche de classification d'image, puis les fonctions d'activation sont utilisées pour transmettre le signal à la couche cachée et les pondérations de connexion. Il peut y avoir plusieurs couches cachées, et chaque couche cachée peut apprendre différentes représentations de caractéristiques. Enfin, la couche de sortie utilise des classificateurs, tels que Softmax, pour mapper les caractéristiques sur des étiquettes de catégorie prédéfinies, permettant ainsi la classification des images.

Notation de crédit : Dans la tâche de notation de crédit, FNN peut faire des prédictions de risque en fonction de divers attributs du client, tels que le revenu, l'historique de crédit, etc. Ces attributs sont utilisés comme nœuds de couche d'entrée, l'extraction de caractéristiques et l'apprentissage de la représentation sont effectués via plusieurs couches cachées, et enfin, la probabilité de défaut prédite est donnée par la couche de sortie. En fonction de cette probabilité, l'institution financière peut attribuer au client une note de crédit et décider d'approuver ou non le prêt.

b. *Réseaux de neurones convolutionnels (CNN)*

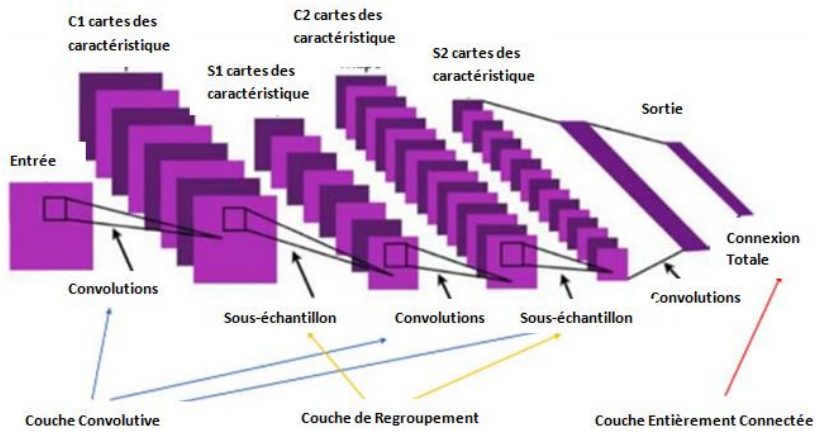


Figure 8 : Structure des réseaux neuronaux convolutionnels

Comme le montre la figure 8 [45], les réseaux neuronaux à propagation directe qui ont une structure profonde et incluent le calcul convolutionnel sont connus sous le nom de réseaux

neuronaux convolutionnels (CNN). Les réseaux de neurones convolutionnels ont été proposés par le mécanisme du domaine de réception en biologie. Les réseaux neuronaux convolutionnels sont conçus pour traiter des données qui ont une structure en forme de grille. Par exemple, une grille unidimensionnelle formée par un échantillonnage régulier sur la chronologie est la forme de données de séries chronologiques et les données d'image sont considérées comme une grille de pixels en deux dimensions.

Dans la figure 8, la couche convolutive dictera la sortie du neurone et la connectera au local en calculant le produit scalaire entre son poids et les zones qui sont connectées au volume d'entrée.

Dans la figure 8, la couche de regroupement est principalement responsable de la réduction de la taille de la carte des caractéristiques, ce qui réduit à son tour la quantité de calculs et de paramètres, et améliore l'efficacité de la formation du modèle. Les fonctions de la couche de regroupement comprennent : 1. Réduction de la dimension : le regroupement réduit la hauteur et la largeur du graphique des caractéristiques et réduit la complexité de calcul. Cela permet d'atténuer le phénomène de surajustement et d'améliorer la capacité de généralisation du modèle. 2. Invariance de translation : les opérations de regroupement peuvent fournir une invariance de translation dans une certaine mesure. Lorsque l'objet de l'image est légèrement décalé, la carte des caractéristiques regroupées peut toujours conserver les informations clés. 3. Sélection des caractéristiques : l'opération de regroupement peut filtrer les caractéristiques pour conserver les caractéristiques importantes. Par exemple, le regroupement maximal préserve la valeur maximale dans chaque zone locale pour mettre en évidence les caractéristiques fortes ; le regroupement moyen calcule la moyenne dans une zone locale, aidant ainsi à préserver la fonction de lissage. Dans la figure 8, dans les dernières couches du CNN, la couche entièrement connectée est souvent utilisée pour intégrer les caractéristiques extraites par les couches de convolution et de regroupement et réaliser la tâche finale de classification ou de régression. Les fonctions de la couche entièrement connectée comprennent : 1. Intégration des caractéristiques : la couche entièrement connectée intègre les caractéristiques locales des couches de convolution et de regroupement dans un vecteur de caractéristiques global pour capturer les relations d'ordre supérieur entre les caractéristiques. 2. Classification ou régression : la couche entièrement connectée peut être utilisée pour mettre en œuvre la tâche finale de classification ou de régression. La tâche de classification utilise généralement la fonction d'activation Softmax dans la couche de sortie pour mapper le vecteur de caractéristiques à la distribution de probabilité de chaque classe. Dans une tâche de

régression, une fonction d'activation linéaire ou une autre fonction d'activation appropriée peut être utilisée pour prédire des valeurs continues. 3. Mappage non linéaire : les fonctions d'activation dans la couche entièrement connectée (telles que ReLU, Sigmoid, etc.) introduisent un mappage non linéaire dans le réseau pour améliorer l'expressivité du modèle.

CNN est un réseau neuronal à propagation directe unique doté d'une connexion locale, d'un partage de poids et d'un regroupement, qui convient au traitement de données avec une structure de grille (telles que des images et des signaux vocaux). Les cas d'application incluent la vision par ordinateur et l'analyse d'imagerie médicale.

Vision par ordinateur : dans les tâches de vision par ordinateur, CNN extrait les caractéristiques de l'image et les classe via des couches de convolution, des couches de regroupement et des couches entièrement connectées. La couche de convolution exploite un noyau de convolution pour glisser sur l'image d'entrée afin de capturer les caractéristiques locales. La couche de regroupement est utilisée pour réduire la taille de la carte des caractéristiques, réduire le calcul et améliorer la robustesse du modèle. La couche entièrement connectée mappe la convolution et les caractéristiques regroupées aux étiquettes de classe. Grâce à cette approche en couches, les CNN peuvent automatiquement apprendre des caractéristiques avancées dans les images et obtenir une reconnaissance d'image précise.

Analyse d'images médicales : CNN peut être utilisé pour le diagnostic automatique des images médicales. Tout d'abord, les images médicales (telles que les rayons X, la tomodensitométrie, l'IRM, etc) sont introduites dans le CNN. Ensuite, les caractéristiques de l'image sont extraites et classées via plusieurs convolutions, regroupements et couches entièrement connectées. Enfin, la couche de sortie donne des résultats de diagnostic, tels que la présence de la tumeur et la localisation de la lésion.

c. Réseaux neuronaux récurrents (RNN)

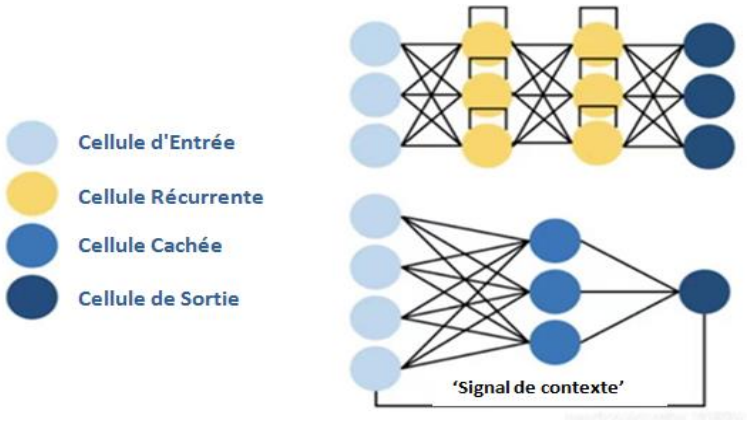


Figure 9 : Structure des réseaux neuronaux récurrents

Un réseau qui s'envoie des signaux de rétroaction est appelé réseau neuronal récurrent (RNN) comme le montre la figure 9 [45]. Le traitement des données de séries chronologiques nécessite un type spécifique de réseau neuronal. La principale caractéristique du RNN est l'existence de connexions cycliques dans le réseau, ce qui permet au réseau de mémoriser les informations précédentes. Voici les principaux composants du RNN et ce qu'ils font :

Cellule d'entrée : Dans la figure 9, la cellule d'entrée est responsable de la réception des données de synchronisation. Lors du traitement de données séquentielles (telles que du texte, des séries chronologiques, etc.), le réseau reçoit des données de la cellule d'entrée pour chaque étape. La cellule d'entrée peut être un vecteur simple ou une structure plus complexe, telle que la couche d'incorporation, qui mappe des données discrètes (telles que des mots) sur un espace vectoriel continu.

Cellule cachée : Dans la figure 9, la cellule cachée est la partie centrale du RNN et contient des unités récurrentes. La couche masquée est informée des données d'entrée de l'étape temporelle actuelle et de l'état masqué de l'étape temporelle précédente à chaque étape temporelle. L'état caché contient des informations sur les étapes temporelles précédentes, donnant au RNN une puissance de mémoire. Les unités cycliques de la cellule cachée peuvent adopter différentes structures, telles que des unités RNN simples, des unités de mémoire à long terme (LSTM) ou des unités récurrentes à grille (GRU). Les principales fonctions de la cellule cachée sont : 1. Apprendre et préserver les dépendances à long terme et à court terme dans les données de séries chronologiques. 2. Les informations de l'étape temporelle précédente sont intégrées à l'entrée de l'étape temporelle actuelle pour générer un nouvel état caché.

Cellule récurrente : Dans la figure 9, la cellule récurrente est le composant central du réseau qui traite les données de synchronisation et conserve les informations historiques. Les principales fonctions de la cellule récurrente sont les suivantes : 1. Traitement des données de séries temporelles : L'unité de boucle permet aux RNN de traiter des données avec une séquence temporelle, telles que du texte, des données de séries temporelles, etc. À chaque étape temporelle, l'unité cyclique reçoit les données d'entrée de l'étape temporelle actuelle et, combinées à l'état caché de l'étape temporelle précédente, génère un nouvel état caché. Cette structure permet au réseau de partager des paramètres entre différentes étapes temporelles pour s'adapter à des séquences de différentes longueurs. 2. Conservation des informations historiques : L'unité de boucle implémente la fonction de mémoire en stockant des informations sur les étapes temporelles précédentes dans un état caché. À chaque étape temporelle, l'unité de boucle met à jour l'état caché, en intégrant les informations de l'étape temporelle précédente à l'entrée de l'étape temporelle actuelle. Cela permet aux RNN de capturer les dépendances à long et à court terme dans les données de séries temporelles. 3. Apprentissage des caractéristiques temporelles : les unités cycliques peuvent apprendre des caractéristiques et des modèles complexes dans les données temporelles. En fonction des exigences de la tâche, l'unité cyclique peut adopter différentes structures, telles qu'une unité RNN simple, une unité de mémoire à long terme (LSTM) ou une unité récurrente à porte (GRU). Ces différentes structures d'unités cycliques permettent un apprentissage et une rétention plus efficaces des caractéristiques et des dépendances dans les données de séries temporelles.

Cellule de sortie : dans la figure 9, l'état caché de la cellule cachée est utilisé par la cellule de sortie pour générer la sortie finale. La cellule de sortie peut être une simple transformation linéaire, une fonction d'activation ou une structure plus complexe telle que le classificateur Softmax. Les principales fonctions de la cellule de sortie comprennent : 1. Mapper les caractéristiques temporelles apprises par le RNN à l'espace cible (telles que les étiquettes de classification, les valeurs de régression, etc.). 2. Dans les tâches de génération de séquences (telles que la traduction automatique, le résumé de texte, etc.), la cellule de sortie peut renvoyer les résultats générés à la cellule d'entrée, formant ainsi une boucle fermée.

Un RNN est un réseau neuronal doté d'un état interne capable de traiter des données de séries chronologiques. Dans les RNN, certains résultats sont renvoyés dans le réseau, ce qui lui confère une capacité de mémoire. Les cas d'application incluent :

Traitement du langage naturel : dans les tâches de traduction automatique, le RNN peut être utilisé pour traiter des séquences de texte de longueur variable. Une structure RNN typique est un codeur et un décodeur. Des unités cycliques (comme LSTM ou GRU) sont utilisées par le codeur pour encoder la séquence de texte de la langue source en une représentation vectorielle de longueur fixe ; le décodeur décode ensuite la représentation en une séquence de texte dans la langue cible. Comme le RNN a des capacités de mémoire temporelle, il peut capturer les dépendances à longue distance dans le texte, permettant une traduction automatique précise.

Reconnaissance vocale : le RNN peut être utilisé pour convertir des signaux vocaux en texte. Dans la tâche de reconnaissance vocale, le signal vocal est d'abord divisé en trames, et des caractéristiques (comme le coefficient de cepstre de fréquence Meir) sont obtenues. Ces caractéristiques sont ensuite introduites dans un RNN (généralement à l'aide d'un LSTM ou d'un GRU bidirectionnel) pour capturer des informations de synchronisation via l'unité de boucle. Enfin, la couche de sortie est décodée à l'aide de la fonction de perte de classification temporelle connexionniste (CTC) pour mapper les étiquettes de trame successives sur la séquence de texte finale.

2.6 Clustering par Intelligence Artificielle

L'intelligence artificielle (IA) est le domaine scientifique et technique concerné par la théorie et la pratique du développement de systèmes qui présentent les caractéristiques que nous associons à l'intelligence dans le comportement humain, telles que la perception, le traitement du langage naturel, la résolution et la planification de problèmes, l'apprentissage et l'adaptation, et agissant sur l'environnement. Son principal objectif scientifique est de comprendre les principes qui permettent un comportement intelligent chez les humains, les animaux et les agents artificiels. Cet objectif scientifique soutient directement plusieurs objectifs d'ingénierie, tels que le développement d'agents intelligents, la formalisation des connaissances et la mécanisation du raisonnement dans tous les domaines de l'activité humaine, rendre le travail avec des ordinateurs aussi simple que le travail avec des personnes et le développement de systèmes homme-machine qui exploitent la complémentarité des raisonnements humain et automatisé [46].

Les algorithmes d'apprentissage automatique sont généralement classés en fonction de la nature de la variable de sortie et du problème spécifique qu'ils visent à résoudre. Ces

algorithmes sont globalement divisés en trois types, à savoir la régression, le clustering et la classification. La régression et la classification sont des types d'algorithmes d'apprentissage supervisé tandis que le clustering est un type d'algorithme non supervisé.

Lorsque la variable de sortie est continue, il s'agit alors d'un problème de régression tandis que lorsqu'elle contient des valeurs discrètes, il s'agit d'un problème de classification. Les algorithmes de clustering sont généralement utilisés lorsque nous devons créer des clusters en fonction des caractéristiques des points de données.

Le clustering est divisé en deux groupes : le clustering hard et le clustering soft. Dans le clustering dur, le point de données est attribué à l'un des clusters uniquement, tandis que dans le clustering souple, il fournit une probabilité qu'un point de données se trouve dans chacun des clusters [3].

Le clustering est l'un des algorithmes les plus connus dans le domaine de l'apprentissage automatique, appelé algorithme d'apprentissage non supervisé. L'importance de l'algorithme de clustering est de diviser le grand volume de données en groupes de données plus petits lorsqu'aucune étiquette de classe n'est disponible pour traiter les ensembles de données. Chaque cluster contient un ensemble de points de données où l'algorithme de clustering est principalement utilisé pour classer et regrouper chaque point de données dans un cluster particulier. En outre, les points de données d'un même cluster doivent avoir des propriétés similaires, tandis que les points de données des différents clusters doivent avoir des propriétés et/ou des fonctionnalités très différentes [47].

3 Conclusion

La technologie de clustering a des applications importantes dans l'exploration de données, la reconnaissance de formes, l'apprentissage automatique et d'autres domaines. Cependant, avec la croissance explosive des données, l'algorithme de clustering traditionnel a de plus en plus de mal à répondre aux besoins de l'analyse du big data. Comment améliorer l'algorithme de clustering traditionnel et garantir la qualité et l'efficacité du clustering dans le contexte du big data est devenu un sujet de recherche important de l'intelligence artificielle et du traitement du big data. L'algorithme de clustering basé sur la densité peut regrouper des ensembles de données de forme arbitraire dans le cas d'une distribution de données inconnue.

Chapitre IV.Clustering Basée sur la Densité des Données

1 Introduction

Le clustering basé sur la densité fait référence à des méthodes d'apprentissage automatique non supervisées qui identifient des clusters distinctifs dans les données, en se basant sur l'idée qu'un cluster/groupe dans un espace de données est une région contiguë de forte densité de points, séparée des autres clusters par des régions clairsemées. Les points de données dans les régions clairsemées et séparées sont généralement considérés comme du bruit/des valeurs aberrantes [2] [48].

L'analyse de cluster est un problème important dans l'analyse de données. Les data scientists utilisent le clustering pour identifier les serveurs défectueux, regrouper les gènes avec des modèles d'expression similaires, identifier les anomalies dans les images biomédicales et effectuer diverses autres applications. Les algorithmes de clustering basés sur la densité regroupent les objets de données en fonction de la densité entre les objets. L'algorithme de clustering basé sur la densité est l'une des principales méthodes de clustering dans le data mining. Les clusters formées en fonction de la densité sont faciles à comprendre et ne se limitent pas aux formes de clusters [49].

Dans le clustering basé sur la densité, les clusters sont définis comme des zones de densité plus élevée séparées par des zones de densité plus faible. Les principaux atouts de l'algorithme de clustering basé sur la densité par rapport aux autres types de clustering sont qu'il a une notion de bruit, peut détecter des clusters de forme arbitraire et n'a pas besoin de connaître le nombre de clusters au préalable [3]. Un cluster est une région dense d'objets entourée d'une région de faible densité. Les objets dans ces zones clairsemées - qui sont nécessaires pour séparer les clusters - sont généralement considérés comme du bruit et des points frontières. L'idée générale des approches de clustering basées sur la densité est de rechercher des régions de haute densité dans l'espace de données. Ces régions peuvent avoir une forme arbitraire et les points à l'intérieur d'une région peuvent être distribués arbitrairement [49].

Le clustering basé sur la densité est un type de méthodes de clustering utilisées dans le data mining pour extraire des modèles inconnus à partir d'ensembles de données [39].

2 Techniques de clustering basées sur la densité

Il convient de mentionner que, à partir d'un même ensemble de données, différents clusterings peuvent être obtenus par différentes méthodes de clustering. Le clustering est généralement effectué par des algorithmes de clustering utilisant des ordinateurs sur de grands ensembles de données ; il n'est essentiellement pas possible d'effectuer cela manuellement. Les clusters peuvent être considérés comme des régions denses dans l'espace de données ; où les clusters sont séparées par une région clairsemée contenant «relativement peu» de données. Compte tenu de cette hypothèse, un cluster peut être de forme «régulière» ou «arbitraire». L'idée derrière le clustering basé sur la densité est de détecter des clusters de formes non sphériques ou arbitraires. Certaines des techniques de clustering basées sur la densité courantes sont DBSCAN, OPTICS, VDBSCAN, DVBSCAN, DBCLASD, ST-DBSCAN et DENCLUE [39].

2.1 DBSCAN "Density-Based Spatial Clustering of Applications with Noise"

DBSCAN [48] est l'une des premières méthodes de clustering basées sur la densité. Il découvre les régions à haute densité dans les bases de données spatiales avec du bruit et en crée des clusters [39].

Les étapes impliquées dans cet algorithme sont les suivantes :

• Initialement, tous les objets sont marqués non visités.

• Ensuite, sélectionner au hasard un objet non visité m et le marquer comme visité.

• Vérifier si son voisinage n'a pas d'objets $MinPts$, puis marquer m comme point de bruit.

• Sinon

• Former un nouveau cluster C et ajouter m à C.

• Ajouter les objets restants du voisinage de m pour définir N.

• Répéter la même procédure pour chaque objet non visité de N jusqu'à ce que tous les objets soient visités.

Avantages :

- ❖ Il peut détecter des valeurs aberrantes.
- ❖ Il peut trouver des clusters de forme non sphérique.
- ❖ Son traitement est rapide.
- ❖ Il n'est pas nécessaire de définir à l'avance le nombre de clusters.

Inconvénients :

- ❖ Il ne peut pas trouver efficacement des clusters si les données ont une densité variable.
- ❖ Il ne convient pas aux données de grande dimension.
- ❖ Il nécessite un rayon et un nombre minimum de points dans le voisinage à spécifier par l'utilisateur à l'avance.

2.2 VDBSCAN "Varied Density Based Spatial Clustering of Applications with Noise"

L'algorithme VDBSCAN peut détecter des clusters de densité variable. De plus, la méthode sélectionne automatiquement plusieurs valeurs du paramètre d'entrée Eps pour différentes densités. Même, le paramètre k est généré automatiquement en fonction des caractéristiques des jeux de données.

Cet algorithme consiste en deux étapes, en choisissant les paramètres Eps_i et cluster avec des densités variées.

Les étapes impliquées dans cet algorithme sont les suivantes :

- Il calcule et stocke $k - dist$ pour chaque projet et partitionne les tracés $k - dist$.
- Le nombre de densités est donné intuitivement par le tracé $k - dist$.
- Le paramètre Eps est sélectionné automatiquement pour chaque densité.
- Il analyse l'ensemble de données et regroupe différentes densités à l'aide d'Eps correspondant
- Affiche le cluster valide en fonction de la densité variée.

Avantages :

- ❖ Des clusters de densités variées peuvent être détectés.
- ❖ Sélection automatique de plusieurs valeurs du paramètre d'entrée Eps pour différentes densités.

Inconvénients :

- ❖ La complexité temporelle est élevée, c'est à dire O (n^2).
- ❖ Pour n objets de données. Si l'indexation spatiale est utilisée, la complexité devient O (n log n). Mais le maintien d'un index spatial prend du temps.

2.3 DVBSCAN "A Density Based Algorithm for discovering Density Varied Clusters in Large Spatial Databases"

L'algorithme DVBSCAN gère la variation de densité locale au sein du cluster. Les paramètres d'entrée utilisés dans cet algorithme sont les objets minimum (μ), le rayon, les valeurs de seuil (α, λ).

Ses étapes sont :

- Un cluster est formé en sélectionnant l'objet principal.
- Ensuite, il calcule la moyenne de densité de cluster (CDM) pour le cluster en croissance avant d'autoriser l'expansion d'un objet principal non traité.
- Le calcul de la variance de densité du cluster (CDV) inclut le voisinage E de l'objet central non traité par rapport au CDM.
- Si le CDV du cluster en croissance par rapport au CDM est inférieur à une valeur seuil spécifiée α et que la différence entre l'objet minimum et maximum se trouvant dans le voisinage E de l'objet est inférieure à une valeur seuil spécifiée λ, alors seul un objet principal non traité est autorisé pour l'expansion.
- Sinon, l'objet est simplement ajouté dans le cluster.

Avantages :

- ❖ Il gère la variation de densité locale au sein du cluster.
- ❖ Il surpasse DBSCAN spécialement en cas de densité locale.

Inconvénients :

- ❖ La complexité temporelle est élevée, c'est à dire O (n^2) pour un ensemble de n objets de données.

2.4 DBCLASD "A Distribution- Based clustering Algorithm for Mining Large Spatial Databases"

Ce nouvel algorithme de clustering DBCLASD détecte les clusters de forme arbitraire et ne nécessite aucun paramètre d'entrée. L'efficacité de DBCLASD sur de grandes bases de données spatiales est également très intéressante.

L'algorithme est décrit comme suit :

- Il s'agit d'un algorithme incrémentiel, c'est à dire qu'un point est ajouté à un cluster basé uniquement sur les points traités jusqu'à présent et sans tenir compte de l'ensemble de la base de données.
- Il augmente progressivement un cluster initial par ses points voisins tant que la distance du voisin le plus proche du cluster résultant correspond à la distribution de distance attendue. Un ensemble de candidats d'un cluster est construit à l'aide de requêtes de région prises en charge par les méthodes d'accès spatial (SAM). Le calcul de m est basé sur le modèle de points uniformément répartis à l'intérieur de du cluster C. Soit A le domaine de C et N le nombre de ses éléments. Une condition nécessaire pour m est :

$$\boldsymbol{N \times P(NNdist_C(P) > m) < 1}$$

Lors de l'insertion d'un nouveau point p dans le cluster C, une requête circulaire avec le centre P et le rayon m est effectuée et les points résultants sont considérés comme de nouveaux candidats.

- L'approche incrémentale implique une dépendance inhérente des clusters de découverte par rapport à l'ordre de génération et de test des candidats.

L'ordre de test des candidats est crucial.

Les candidats qui ne sont pas acceptés par le test pour la première fois sont appelés candidats non retenus. Pour minimiser la dépendance sur l'ordre des tests, les deux fonctionnalités suivantes sont prises en compte :

1. Les candidats non retenus ne sont pas écartés mais ils sont rejugés plus tard.
2. Les points déjà attribués à un cluster peuvent basculer vers un autre cluster ultérieurement.

Les tests des candidats sont effectués en deux étapes :

1. Le cluster actuel est augmenté par le candidat.
2. Le test Chi-square est utilisé pour vérifier l'hypothèse que l'ensemble de distances du voisin le plus proche du groupe augmenté correspond toujours à la distribution de distance attendue.

Avantages :

- ❖ Il est très efficace sur les grandes bases de données spatiales.
- ❖ Il ne nécessite aucun paramètre d'entrée.

Inconvénients :

- ❖ La complexité est $3n^2$.
- ❖ Il est plus lent que DBSCAN et DENCLUE.

2.5 ST-DBSCAN "Spatial- Temporal Density Based Clustering"

L'algorithme ST-DBSCAN est construit en modifiant l'algorithme DBSCAN. Contrairement à l'algorithme de clustering basé sur la densité existant, l'algorithme ST-DBSCAN a la capacité de découvrir des clusters par rapport aux valeurs non spatiales, spatiales et temporelles des objets. Il convient aux applications impliquant des données spatio-temporelles telles que les systèmes d'information géographique, l'imagerie médicale et les prévisions météorologiques.

La technique commence par le premier point p de la base de données D.

- Ce point p est traité selon l'algorithme DBSCAN et le point suivant est pris.
- La fonction Retrieve_Neighbors (object, $Ep1$, $Ep2$) récupère tous les objets accessibles en densité à partir de l'objet sélectionné par rapport à $Eps1$, $Eps2$ et $MinPts$. Si les points renvoyés dans Eps-neighborhood sont plus petits que l'entrée $MinPts$, l'objet est affecté en tant que bruit.
- Les points marqués comme bruit peuvent être modifiés ultérieurement, c'est à dire que les points ne sont pas directement accessibles en densité, mais ils seront accessibles en densité.

- Si le point sélectionné est un objet principal, un nouveau cluster est construit. Ensuite, tous les voisins directement accessibles en densité de ces objets principaux sont également inclus.
- Ensuite, l'algorithme collecte de manière itérative les objets accessibles en densité à partir de l'objet principal à l'aide d'une pile.
- Si l'objet n'est pas marqué comme bruit ou s'il n'est pas dans un cluster et que la différence entre la valeur moyenne du cluster et la nouvelle valeur est plus petite que E, il est placé dans le cluster actuel.
- Si deux clusters $C1$ et $C2$ sont très proches l'un de l'autre, un point p peut appartenir à la fois à $C1$ et $C2$. Ensuite, le point p est affecté au cluster qui est découvert en premier.

Avantages :

- ❖ Le clustering des données spatio-temporelles peut être effectué sur la base d'attributs non spatiaux, spatiaux et temporels.
- ❖ Il peut détecter des points de bruit même lorsqu'il est de densité variée en attribuant un facteur de densité à chaque cluster.

Inconvénients :

- ❖ Le paramètre d'entrée n'est pas généré automatiquement.
- ❖ La performance de l'algorithme doit également être améliorée.

2.6 OPTICS "Ordering Points To Identify the Clustering Structure"

Pour surmonter la difficulté d'utiliser un ensemble de paramètres globaux dans l'analyse de clustering, une méthode d'analyse de cluster appelée OPTICS a été proposée.

Il s'agit d'une méthode indirecte, c'est à dire qu'OPTICS ne produit pas explicitement un clustering d'ensembles de données ; au lieu de cela, il génère un ordre de cluster. Il s'agit d'une liste linéaire de tous les objets en cours d'analyse et représente la structure de clustering basée sur la densité des données. Les objets d'un cluster plus dense sont répertoriés plus près les uns des autres dans l'ordre des clusters. Cet ordre équivaut à un clustering basé sur la densité obtenu à partir d'un large éventail de réglages de paramètres.

Avantages :

- ❖ il n'est pas nécessaire que l'utilisateur fournisse un seuil de densité.
- ❖ L'ordre de clustering est utile pour extraire les informations de clustering de base.

Inconvénients :

- ❖ Il ne produit qu'un ordre de cluster.
- ❖ Il ne peut pas gérer des données de grande dimension.

2.7 DENCLUE « DENsity-based CLUstEring »

Il s'agit d'un algorithme de clustering qui dépend des fonctions de densité. Il s'agit d'une amélioration par rapport aux méthodes de clustering de base basées sur la densité DBSCAN et OPTICS en termes d'estimation de la densité.

La technique est brièvement décrite ci-dessous :

- Une méthode d'estimation statistique de la densité est utilisée pour estimer la densité du noyau. Il en résulte la valeur maximale locale de densité.
- Des clusters peuvent être formés à partir de cette valeur maximale locale de densité.
- Si la valeur de densité locale est très petite, les objets des clusters sont rejetés sous forme de bruit

Dans cette méthode, les objets considérés sont ajoutés à un cluster par le biais d'attracteurs de densité en utilisant une procédure d'escalade par étapes.

Avantages :

- ❖ Plus rapide que les algorithmes existants.
- ❖ Bon pour la base de données qui contient une énorme quantité de bruit.
- ❖ Produire un résultat précis.

Inconvénients :

- ❖ Nécessite un grand nombre de paramètres.
- ❖ Le paramètre de densité doit être sélectionné avec soin.
- ❖ Il a besoin de solides bases mathématiques.

2.8 DPC « Density Peak Clustering »

L'algorithme DPC doit calculer la densité locale et la distance de séparation pour chaque point de données [24]. Étant donné un ensemble de données X, la densité locale $\rho(x_i)$ d'un point de données $x_i \in X$ est le nombre de points de données au voisinage de x_i. C'est à dire :

$$\boldsymbol{\rho(x_i) = |B(x_i)|} \qquad \textbf{(18)}$$

Où $B(x_i)$ désigne le voisinage de x_i et est défini comme l'ensemble des points de données dans X dont la distance à x_i est inférieur à un paramètre spécifié par l'utilisateur d_c. C'est à dire :

$$\mathbf{B(x_i) = \{x_j \in X | d(x_i, x_j) < d_c\}} \qquad \textbf{(19)}$$

Où $d(x_i, x_j)$ représente la distance entre x_i et x_j. Notamment, Les équations (18) et (19) utilisent le paramètre d_c comme seuil dur pour dériver respectivement le voisinage et la densité locale d'un point de données.

La valeur de d_c peut être choisie de sorte que le nombre moyen de voisins d'un point de données soit d'environ p% du nombre de points de données dans X, et la valeur suggérée pour p est comprise entre 1 et 2. Pour les petits ensembles de données, Rodriguez et Laio [25] ont suggéré d'utiliser un noyau exponentiel pour calculer la densité locale, comme indiqué dans l'équation (20) :

$$\boldsymbol{\rho(x_i) = \sum_{x_j \in X} \exp\left(-\frac{d(x_i,x_j)^2}{{d_c}^2}\right)} \qquad \textbf{(20)}$$

La distance de séparation $\delta(x_i)$ de x_i est la distance minimale de x_i à tout autre point de données avec une densité locale>$\rho(x_i)$, ou la distance maximale de x_i à tout autre point de données dans X s'il n'existe aucun point de données avec une densité locale>$\rho(x_i)$, comme indiqué dans l'équation (21) :

$$\boldsymbol{\delta(x_i) = \begin{cases} \min\limits_{j:\rho(x_j)>\rho(x_i)} d(x_i, x_j), si\ \rho(x_i) < \max\limits_{x_j \in X} \rho(x_j) \\ \max\limits_{x_j} d(x_i, x_j), sinon \end{cases}} \qquad \textbf{(21)}$$

Pour faciliter l'exposition, nous utilisons $\sigma(x_i)$ pour désigner l'indice *j* du point de données x_j qui est le plus proche de x_i et $\rho(x_j)>\rho(x_i)$, et si un tel point de données n'existe pas, $\sigma(x_i)$ est réglé sur *i*, comme indiqué dans l'équation (22):

$$\boldsymbol{\sigma(x_i)} = \begin{cases} \underset{j:\rho(x_j)>\rho(x_i)}{\mathbf{argmin}} \; \boldsymbol{d(x_i, x_j)}, \boldsymbol{si\rho(x_i)} < \underset{x_j \in X}{\max} \boldsymbol{\rho(x_j)} \\ \boldsymbol{i, sinon} \end{cases} \quad \textbf{(22)}$$

Notamment, il peut y avoir plus d'un point de données qui est le plus proche de x_i et a une densité locale$>\rho(x_i)$.

Une fois que $\rho(x_i)$ et $\delta(x_i)$ de chaque point de données ont été déterminés, l'algorithme DPC utilise l'hypothèse suivante pour sélectionner les pics de densité : si un point de données $x_i \in X$ est un pic de densité, alors x_i doit être entouré de nombreux points de données (c'est-à-dire, $\rho(x_i)$ est large) et doit être à une distance relativement élevée des autres points de données avec une densité locale supérieure à $\rho(x_i)$ (c'est à dire, $\delta(x_i)$ est large). Pour aider le choix des pics de densité, l'algorithme DPC trace chaque point de données dans un graphe de décision, qui est un graphe bidimensionnel avec la densité locale et la distance de séparation comme axes horizontal et vertical, respectivement. Les points de données présentant à la fois une densité locale élevée et une distance de séparation élevée sont sélectionnés manuellement comme pics de densité. Alternativement, on peut définir un seuil sur $\gamma(x_i) = \rho(x_i)\delta(x_i)$ et sélectionner des points de données avec $\gamma(x_i)$ supérieur au seuil en tant que pics de densité.

Une fois que tous les pics de densité ont été déterminés, chaque pic de densité sert de point de départ d'un cluster, et donc le nombre de pics de densité est égal au nombre de clusters. Chaque point de données sans pic est affecté au même cluster que son point de données le plus proche de densité plus élevée, c'est à dire des points de données x_i est affecté au cluster qui contient $x_{\sigma(x_i)}$. Soit y_i le nom de cluster du point de données x_i, alors $y_i = y_{\sigma(x_i)}$.

La figure 10 montre l'algorithme DPC [24]. Notamment, il est important de trier les points de données par leur densité locale de manière décroissante à l'étape 2 afin que le calcul de $\delta_{(x_i)}$ et $\sigma_{(x_i)}$ à l'étape 3 et l'affectation de cluster à l'étape 6 puissent être effectués efficacement. Sans l'étape 2, pour chaque point de données x_i, l'étape 3 nécessiterait de balayer tous les points de données dans *X* pour trouver les points de données avec une densité locale$>\rho_{(x_i)}$. Avec l'étape 2, l'étape 3 n'a besoin que d'analyser les points de données situés avant x_i dans *X*, et, par conséquent, réduit la durée d'exécution de l'étape 3 de moitié en

moyenne. De plus, à l'étape 2, les points de données avec une densité locale plus élevée sont traités plus tôt à l'étape 6. Puisque $\left(\mathrm{x}_{\sigma(\mathrm{x_i})}\right) > \rho(\mathrm{x_i})$, $\mathrm{y}_{\sigma(\mathrm{x_i})}$ sera déterminé avant $\mathrm{y_i}$ à l'étape 6, et, ainsi, L'étape 6 peut terminer l'attribution de cluster en temps O (N).

Algorithme DPC.

Entrée : l'ensemble des points de données $X \in \mathbb{R}_{N\times M}$et les paramètres d_c pour définir le voisinage, et d_r pour sélectionner les pics de densité.

Sortie : le vecteur d'étiquette de l'indice de cluster $y \in \mathbb{R}_{N\times 1}$.

Algorithme :

1. Calculer $\rho(x_i)$ pour chaque $x_i \in X$ en utilisant soit (18) ou (20).
2. Trier tous les points de données dans X par leurs densités locales de manière décroissante.
3. Calculer $\delta(x_i)$ et $\sigma(x_i)$ pour chaque $x_i \in X$ en utilisant respectivement (21) et (22).
4. Sélectionner les points de données avec $\rho(x_i)\ \delta(x_i) > d_r$ comme pics de densité.
5. Pour chaque pic de densité x_i, définir $y_i = i$ point de départ de chaque cluster.
6. Pour chaque point de données sans pic x_i, définir $y_i = y_{\sigma(x_i)}$ l'affectation de cluster.
7. Retourner y.

Figure 10 : Algorithme DPC

3 Conclusion

Le clustering basé sur la densité constitue une contribution majeure à l'analyse exploratoire des données, permettant une découverte intuitive et robuste des structures inhérentes à divers types de jeux de données. Il constitue une famille d'algorithmes particulièrement adaptée aux données complexes, irrégulières et bruitées. Les différentes techniques abordées dans ce chapitre offrent une diversité de solutions pour répondre aux besoins variés des applications

modernes. Ce cadre conceptuel et méthodologique pose ainsi les bases pour les développements ultérieurs et les contributions futures dans ce domaine.

Chapitre V.Synthèse des Travaux Existants

1 Introduction

La segmentation d'images est l'un des domaines de recherche les plus populaires en vision par ordinateur et constitue la base de la reconnaissance de formes et de la compréhension d'images. Le développement des techniques de segmentation d'images est étroitement lié à de nombreuses disciplines et domaines, par exemple, les véhicules autonomes, la technologie médicale intelligente, les moteurs de recherche d'images, l'inspection industrielle et la réalité augmentée.

La segmentation d'images divise les images en régions ayant des caractéristiques différentes et extrait les régions d'intérêt (ROI). Ces régions, selon la perception visuelle humaine, sont significatives et ne se chevauchent pas. La segmentation d'images présente deux difficultés : (1) comment définir les « régions significatives », car l'incertitude de la perception visuelle et la diversité de la compréhension humaine conduisent à un manque de définition claire des objets, ce qui fait de la segmentation d'images un problème mal posé ; et (2) comment représenter efficacement les objets dans une image.

Les images numériques sont constituées de pixels, qui peuvent être regroupés pour constituer des ensembles plus grands en fonction de leur couleur, de leur texture et d'autres informations. On les appelle « ensembles de pixels » ou « superpixels ». Ces caractéristiques de bas niveau reflètent les attributs locaux de l'image, mais il est difficile d'obtenir des informations globales (par exemple, la forme et la position) à travers ces attributs locaux.

2 Contexte général et état de l'art

Depuis les années 1970, la segmentation d'images a reçu une attention continue de la part des chercheurs en vision par ordinateur. Les méthodes de segmentation classiques se concentrent principalement sur la mise en évidence et l'obtention des informations contenues dans une seule image, ce qui nécessite souvent des connaissances professionnelles et une intervention humaine. Cependant, il est difficile d'obtenir des informations sémantiques de haut niveau à partir d'images. Les méthodes de co-segmentation impliquent l'identification d'objets communs à partir d'un ensemble d'images, ce qui nécessite l'acquisition de certaines

connaissances préalables. Étant donné que l'annotation d'image de ces méthodes est supervisée, elles sont classées comme des méthodes semi-supervisées ou faiblement supervisées. Avec l'enrichissement des ensembles de données d'images d'annotation fine à grande échelle, les méthodes de segmentation d'images basées sur des réseaux neuronaux profonds sont progressivement devenues un sujet populaire.

Bien que de nombreuses réalisations aient été réalisées dans la recherche sur la segmentation d'images, de nombreux défis subsistent, par exemple la représentation des caractéristiques, la conception de modèles et l'optimisation.

La littérature présente plusieurs techniques de clustering de données, telles que des algorithmes hiérarchiques, algorithmes basés sur des modèles, et algorithmes basés sur la densité. Les algorithmes de clustering basés sur le partitionnement tels que k-means et autres [40] spécifient un nombre initial de clusters au début et représentent chaque cluster par un centroïde, et les objets proches du même centroïde sont considérés comme similaires. Ces algorithmes déterminent tous les clusters en même temps en visant une forte similarité intra-groupe. Ils nécessitent un certain nombre de clusters définis par l'utilisateur qui sont souvent inconnus en pratique. De plus, ils construisent des clusters sphériques, qui ne conviennent pas aux clusters de forme arbitraire.

Les algorithmes de clustering hiérarchique [50] sont des approches ascendantes telles que chaque point de données commence dans un cluster séparé, et les paires de clusters en bas sont fusionnées à mesure que nous remontons dans la hiérarchie. Dans les algorithmes de clustering hiérarchiques, le processus de clustering gère bien les clusters de différentes formes, mais ils ne conviennent pas aux clusters de densité variable.

Les méthodes de clustering basées sur un modèle [51] estiment un modèle pour les données en incorporant une mesure de probabilité ou d'incertitude dans les affectations de cluster. Le clustering basé sur un modèle tente de résoudre ce problème et de fournir une allocation flexible lorsque les observations sont susceptibles d'appartenir à chaque cluster. En outre, le clustering basé sur un modèle offre l'avantage supplémentaire d'identifier automatiquement le nombre optimal de clusters.

Dans les méthodes de clustering basées sur la densité, Rodriguez et Laio [25] ont proposé un nouvel algorithme de clustering en trouvant des pics de densité qui sont les centres potentiels du cluster. Dans cet algorithme, chaque point de données peut obtenir sa propre

densité locale et sa distance au plus proche voisin qui a une densité plus élevée. Selon la densité et la distance des points, le résultat du regroupement peut être obtenu en une seule étape sans itération supplémentaire. Comme les méthodes de clustering basées sur la recherche de pics de densité produisent des clusters de formes arbitraires, la notion de "centre" est quelque peu trompeuse. Ces algorithmes de clustering basés sur des pics de densité présentent une déficience dans le processus d'allocation, ce qui est susceptible de déclencher un effet domino. Ils nécessitent un paramètre défini par l'utilisateur (distance de coupure) et un jugement visuel pour estimer les centres de cluster sur la base du diagramme de décision. De plus, ils ne peuvent pas traiter certains ensembles de données non sphériques tels que Spiral.

DBSCAN [48] est probablement l'algorithme de clustering basé sur la densité le plus populaire pour construire des clusters de forme arbitraire. Un cluster est créé s'il y a un $MinPts$ d'objets dans une région dense avec un Eps (seuil de distance); $MinPts$ (taille minimale du cluster), qui sont définis par l'utilisateur. DBSCAN démarre avec un point de départ arbitraire. Le voisinage Eps de ce point est récupéré et un cluster est créé, s'il contient un nombre suffisant de points. Sinon, le point est considéré comme du bruit. Le point sélectionné et ses points accessibles sont considérés comme un cluster. Le cluster augmentera plusieurs fois en ajoutant d'autres points qui sont à la distance Eps des points accessibles comme nouveaux points accessibles.

L'algorithme continue jusqu'à ce que tous les points aient soit une étiquette de cluster ou qu'il y ait moins de points $MinPts$ autour d'eux et ils sont donc considérés comme des bruits. Le tableau 1 résume la comparaison entre les travaux les plus notables sur le clustering basé sur la densité.

Méthode	Principales caractéristiques	Limites
DBSCAN	Les utilisateurs doivent définir deux paramètres : La distance Eps qui détermine le rayon de voisinage autour de chaque point, et le nombre minimum de points $MinPts$ requis pour qu'une région soit considérée comme un cluster.	DBSCAN ne parvient pas à découvrir des clusters de densités variables et nécessite deux paramètres d'entrée définis par l'utilisateur Eps et $MinPts$.
DPC	L'utilisateur doit sélectionner les centres de cluster à partir d'un graphique de décision pour identifier les clusters	Sensible à la saisie de l'utilisateur et les résultats dépendent du centre sélectionné par l'utilisateur
GMDBSCAN	Utilise une technique de division spatiale et considère chaque grille comme une partie distincte, puis il estime les $MinPts$ indépendants pour chaque grille en fonction de sa densité	Applique plusieurs DBSCAN sur chaque grille et enfin, il utilise une méthode basée sur la distance pour améliorer les limites
DPCSA	L'utilisateur doit sélectionner les centres de clusters à partir d'un graphique de décision pour identifier les clusters en deux étapes	Sensible à l'entrée de l'utilisateur

Tableau 1 : Comparaison à l'aide des principales caractéristiques et limites

DBSCAN et ses variantes [48][51][52][53] mentionnées ci-dessus présentent des avantages significatifs : i) découvrir des clusters avec différentes formes, tailles et bruits, ii) robuste aux valeurs aberrantes, iii) inutile de préciser le nombre de clusters dans les données, iv) insensible à l'ordre des points. Cependant, ces algorithmes ne parviennent pas à découvrir des clusters de densités variables et nécessitent deux paramètres d'entrée définis par l'utilisateur Eps et $MinPts$.

La méthode GMDBSCAN [54] utilise une technique de division spatiale et considère chaque grille comme une partie distincte, puis il estime les $MinPts$ indépendants pour chaque grille en fonction de sa densité, après quoi il applique plusieurs DBSCAN sur chaque grille et enfin, il utilise une méthode basée sur la distance pour améliorer les limites.

Hou et al. ont proposé un algorithme de clustering sans paramètres basé sur la combinaison des algorithmes DSets (ensembles dominants) et DBSCAN [48]. Dans l'algorithme AGED [53], la valeur de l'Eps définie dans DBSCAN est déterminée en fonction des densités locales. Wang et al. ont proposé un algorithme DBSCAN modifié pour déterminer automatiquement les valeurs Eps pour différentes distributions de données.

Dans [55], l'auteur a introduit une méthode qui estime la densité locale - pour chaque point de l'ensemble de données - comme la somme des distances aux k plus proches voisins. En organisant les points de données par ordre croissant selon la densité locale, l'algorithme proposé détermine des clusters de différentes densités. Cependant, le seuil de densité associé à l'échantillon initiateur de cluster manque de définition précise, ce qui conduit à des échecs de clustering dans certaines situations.

3 Conclusion

Ce chapitre a permis de présenter une synthèse des travaux existants dans le domaine de la segmentation d'images, avec un accent particulier sur les algorithmes de clustering et, plus précisément, ceux basés sur la densité. Nous avons examiné les principales approches, telles que l'algorithme DBSCAN, et leurs variantes, en mettant en évidence leur efficacité dans l'identification de clusters de formes complexes et leur capacité à gérer les données bruitées.

Toutefois, notre analyse a également révélé plusieurs limitations, notamment la difficulté des algorithmes basés sur la densité à gérer les ensembles de données à densité variable et la sensibilité aux paramètres, tels que le rayon de voisinage Eps et le nombre minimum de points $MinPts$. Ces défis soulignent la nécessité de développer des méthodes adaptées pour des données plus complexes et des applications spécifiques, comme celles rencontrées dans la segmentation d'images.

Ces constats positionnent notre recherche dans une perspective où l'objectif est de surmonter ces limitations en proposant « Clustering d'Ensembles de Données Multi-densités Utilisant les K voisins les Plus Proches et l'Inégalité de Chebyshev ». Le chapitre suivant détaillera la méthodologie adoptée pour répondre à ces problématiques et introduira les concepts et techniques utilisés dans notre démarche.

Chapitre VI. Clustering d'Ensembles de Données Multi-densités Utilisant les K-voisins les Plus Proches et l'Inégalité de Chebyshev

1 Introduction

Le clustering est un puissant outil d'apprentissage automatique pour détecter des structures dans des ensembles de données. La détection, l'analyse et la description des clusters naturels au sein d'un ensemble de données sont d'une importance fondamentale pour un certain nombre de domaines différents [21] [56]. De tels domaines concernent la bioinformatique pour identifier des gènes similaires, le marketing pour segmenter les clients afin d'établir des études de marché, les sciences sociales, la psychologie, la biologie, la sécurité, la vision par ordinateur et le traitement d'images dans divers domaines tels que le domaine médical ou industriel, etc.

Les techniques de clustering sont généralement divisées en quatre classes : 1) partitionnement [4] [12] 2) hiérarchique [50], 3) basé sur un modèle [9], et 4) basé sur la densité [48]. En raison de leur capacité à supporter des clusters de formes arbitraires, les algorithmes de clustering basés sur la densité sont naturellement largement utilisés. Ils considèrent les clusters comme des régions denses séparées par des zones moins denses ou dispersées. Ils déterminent les clusters ainsi que leur nombre, tout en identifiant le bruit.

Parmi les algorithmes de clustering basés sur la densité les plus populaires, DBSCAN [48] présente plusieurs avantages essentiels. Premièrement, il regroupe les données en clusters de formes arbitraires. Deuxièmement, il n'est pas nécessaire que le nombre de clusters soit donné en entrée. Le nombre de clusters est déterminé par la nature des données et les valeurs de Eps et $MinPts$. Troisièmement, il est insensible à l'ordre d'entrée des points dans l'ensemble de données. Tous ces points forts sont très importants pour tout algorithme de clustering.

Cependant, DBSCAN nécessite que l'utilisateur fournisse un seuil de séparation global Eps et le nombre minimum de points $MinPts$ pour tout cluster. DPC (Density Peak Clustering) demande également à l'utilisateur de saisir le seuil de distance dc. Pour ces deux types d'algorithmes et leurs variantes, les résultats de clustering dépendent de ces paramètres d'entrée qui sont difficiles à déterminer par l'utilisateur, en l'absence de règles précises pour leur calcul. Bien qu'ils aient apporté des avancées importantes dans les techniques de clustering et de plus en plus largement utilisées, ils présentent toujours des limites pour traiter efficacement des ensembles de données complexes ayant des clusters de formes, de tailles et de densités variables [3].

Dans ce chapitre, nous proposons une nouvelle approche de clustering basée sur la densité qui surmonte ces limites. Elle est basée sur l'idée principale de caractériser les points de données dans les ensembles de données avec leurs densités locales et d'utiliser l'inégalité de Chebyshev pour estimer les Eps locaux pour chaque cluster. Ce faisant, non seulement nous déterminons automatiquement Eps, mais nous gérons également efficacement des clusters de différentes densités et formes.

Nous définissons le concept de voisins locaux d'un point de données donné x, à partir duquel il détermine le seuil local Eps pour le cluster courant. En effet, pour chaque point de données x dans l'ensemble de données, nous calculons la densité locale de manière à ce qu'elle soit la plus petite dans une région dense et la plus grande dans une région clairsemée. Ensuite, nous trions l'ensemble de données sur la clé de densité, par ordre décroissant.

Ce faisant, le premier point de données avec la densité la plus élevée initiera la construction du premier cluster en calculant son seuil de séparation local Eps à l'aide de l'inégalité de Chebyshev [57] et de ses distances, appliquées aux k plus proches voisins. Suite à quoi, notre algorithme étend le cluster initialement construit à partir de ces k-voisins de la même manière que DBSCAN. Lorsqu'aucun nouveau Eps-voisin n'est trouvé, l'algorithme ferme le cluster actuel et répète la même procédure pour les points de données non classés. Il s'arrête lorsque l'ensemble de données est vide et ainsi, tous les clusters sont déterminés automatiquement, sans nécessiter de seuil de distance ni de nombre de clusters. Ce faisant, l'Eps local peut être adapté à une densité de cluster spécifique, ce qui est un avantage de la méthode proposée par rapport à DBSCAN et ses améliorations.

Le reste du chapitre est organisé comme suit : La section 2 Présente les algorithmes DBSCAN et DPC. La section 3 décrit notre approche pour regrouper des ensembles de données avec différentes densités, formes et tailles. La section 4 présente l'évaluation expérimentale. La section 5 présente une discussion. Enfin, la section 6 conclut le chapitre et décrit les travaux futurs.

2 Clustering basé sur la densité

Un processus de clustering partitionne un ensemble de données en groupes de points de données en fonction de leur proximité les uns par rapport aux autres. Il prend en entrée un ensemble de points de données et une mesure de similarité entre eux, et produit en sortie un ensemble de partitions décrivant la structure générale de l'ensemble de données. Les techniques de clustering basées sur la densité sont l'une des techniques d'apprentissage non supervisé les plus populaires utilisées dans les algorithmes d'apprentissage automatique. Ils procèdent en détectant les zones où les points de données sont concentrés et où ils sont séparés par des zones clairsemées ou vides. Les points qui n'appartiennent à aucun cluster sont étiquetés comme bruit.

L'analyse de clustering basé sur la densité est bien connue et largement appréciée pour deux caractéristiques souhaitables. Premièrement, il ne nécessite pas un nombre prédéfini de clusters comme paramètre d'entrée et est robuste au bruit. Deuxièmement, il peut découvrir des clusters avec des formes arbitraires, et pas seulement des clusters en forme de boule. L'algorithme de clustering basé sur la densité a joué un rôle essentiel dans la recherche d'une structure de formes non linéaires basée sur la densité.

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) est l'algorithme basé sur la densité le plus largement utilisé. Il utilise le concept d'accessibilité de densité et de connectivité de densité [12] [52]. DBSCAN est un algorithme simple qui définit les clusters à l'aide d'une estimation de densité locale. Il peut être divisé en 4 étapes [54]:

Étape 1 : Pour chaque observation, on regarde le nombre de points au plus à une distance Eps de celle-ci. Cette zone est appelée Eps-voisinage de l'observation.

Étape 2 : Si une observation a au moins un certain nombre de voisins, y compris elle-même, elle est considérée comme une observation centrale. Une observation de haute densité a alors été détectée.

Étape 3 : Toutes les observations à proximité d'une observation centrale appartiennent au même cluster. Il peut y avoir des observations centrales proches les unes des autres. Par conséquent, étape par étape, nous obtenons une longue séquence d'observations centrales qui constitue un seul cluster.

Étape 4 : Toute observation qui n'est pas une observation centrale et qui n'inclut pas d'observation centrale dans son voisinage est considérée comme une anomalie.

Par conséquent, pour appliquer DBSCAN, les utilisateurs doivent définir deux paramètres : Quelle distance Eps déterminer pour chaque observation ? Quel est le nombre minimum de points $MinPts$ nécessaire pour considérer qu'une observation est une observation centrale ? Ces deux paramètres sont fournis librement par l'utilisateur.

Contrairement à l'algorithme k-means ou à la classification ascendante hiérarchique, il n'est pas nécessaire de définir le nombre de clusters, ce qui rend l'algorithme moins rigide. Un autre avantage de DBSCAN est qu'il permet également de gérer les valeurs aberrantes ou les anomalies.

En revanche, la notion de distance et le choix des Eps et $MinPts$ sont critiques pour la qualité du clustering. Dans cet algorithme, deux points sont fondamentaux :

Quelle est la métrique utilisée pour évaluer la distance entre une observation et ses voisins ? Quel est l'Eps idéal ?

Dans le DBSCAN, nous utilisons généralement la distance euclidienne, soit $p = (p1, \dots., pn)$ et $q = (q1, \dots., qn)$:

$$d(\mathbf{p},\mathbf{q}) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_i - q_i)^2 + \cdots + (p_n - q_n)^2} = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}$$

A chaque observation, pour compter le nombre de voisins à une distance Eps au plus, on calcule la distance euclidienne entre le voisin et l'observation et on vérifie si elle est inférieure à Eps.

3 Méthode proposée

Dans cette section, nous présentons notre méthode pour résoudre les problèmes mentionnés ci-dessus dans le clustering d'ensembles de données avec différentes densités. L'idée principale est d'utiliser le concept de densité locale et l'inégalité statistique de Chebyshev [57] pour déterminer automatiquement un seuil de séparation pour chaque classe de densité dans un ensemble de données, et nous construisons chaque cluster avec son seuil de distance associé, conduisant à des clusters de densités et de formes variées. Notre algorithme que nous

appelons MDC (Multi-Density Clustering) utilise les informations des k plus proches voisins pour définir une densité locale de chaque point de données comme suit :

$$\boldsymbol{\rho_i = \sum_{j \in KNN(i)} \exp(-d_{ij})} \qquad \textbf{(23)}$$

Où ρ_i est la densité locale d'un point de données i, et d_{ij} est la distance entre le point de données i et un point de données voisin j dans le voisinage KNN(i). Ce faisant, la valeur de densité est réduite de la densité globale à la densité locale des k voisins les plus proches du point de données, et plus la distance entre un point de données et son voisinage k est grande, plus la densité locale est petite. Cette constatation reflétera mieux les informations locales dans l'ensemble de données.

3.1 L'inégalité de Chebyshev

L'inégalité de Chebyshev est adéquate pour des distributions complètement arbitraires [57]. Elle constitue une technique d'estimation efficace facilitant le calcul de la probabilité uniquement sur la base de l'espérance mathématique et de la variance, sans avoir besoin d'informations sur la distribution. L'inégalité de Chebyshev est très utile dans la mesure où elle peut être appliquée à des distributions complètement arbitraires.

L'inégalité de Chebyshev est valable dans le cas où x (variable aléatoire) avec :

une moyenne ($M(x) = \mu < \infty$) et un écart ($V(x) = \sigma < \infty$).

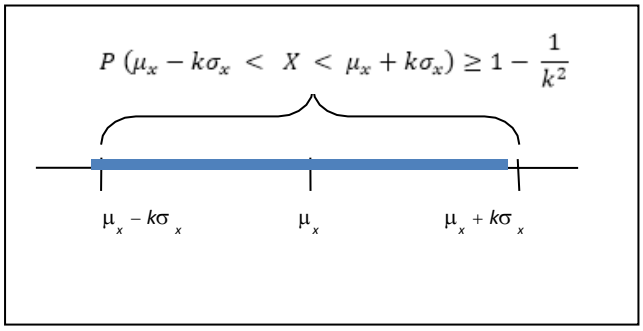


Figure 11 : Intervalle de µx

Cette inégalité s'exprime comme suit :

$$\mathbf{P\{|x - \mu| \geq k\sigma\} \leq \frac{1}{k^2}} \qquad \textbf{(24)}$$

et elle est valable pour tout $k > 1$.

Nous pouvons exprimer une variante de l'inégalité de Chebyshev comme l'expression suivante :

$$\mathbf{P\{|x-\mu|<k\sigma\}>1-\frac{1}{k^2}} \qquad \textbf{(25)}$$

Et on reformule dans l'inégalité suivante (Figure 11) :

$$\boldsymbol{P(\mu_x - k\sigma_x < X < \mu_x + k\sigma_x) \geq 1 - \frac{1}{k^2}} \qquad \textbf{(26)}$$

Cela signifie que plus que $(1-\frac{1}{k^2})$ de la population tombe dans $k(k>1)$ écarts types (c'est à dire kσ) de la moyenne de la population μ. En d'autres termes, il indique que pour une distribution, le pourcentage d'observations qui se situent à l'intérieur de k écarts types est au moins : $1-\frac{1}{k^2}$ (Figure 11).

3.2 Algorithme proposé

Dans l'étape de clustering, l'algorithme de clustering proposé est utilisé pour extraire les clusters spatiaux de l'ensemble de données. Enfin, les sorties sont évaluées et visualisées. L'algorithme MDC prend en entrée un ensemble de données et produit des clusters de différentes formes et de différentes densités.

Il prend le point de la plus haute densité comme initiateur de cluster CI qui absorbe ses k plus proches voisins $\{v1, v2, \dots, vk\}$ pour construire un nouveau cluster. Ainsi, le cluster actuellement initié est composé de points de données $C = \{CI, v1, v2, \dots, vk\}$. À cette étape, nous utilisons l'inégalité de Chebyshev pour estimer l'Eps local du cluster. Nous calculons les distances par paires de C comme distances entre deux points de données dans C pour avoir $D = \{d1, d2, \dots, dm\}$. Eps est estimé comme la borne de D. Pour estimer l'extension maximale du cluster actuel, on considère l'ensemble des distances D et on applique la formule de Chebyshev suivante :

$$\boldsymbol{Eps = mean + 3.5 \times std} \qquad \textbf{(27)}$$

Où $mean$ est la moyenne de D et std est l'écart type de D. Ensuite, nous supprimons les points de données classifiés de l'ensemble de données. On initie un nouveau cluster en retirant un nouvel initiateur et on procède de la même manière, tout en supprimant les points de données classifiés de l'ensemble de données, tout en utilisant $MinPts$ pour contrôler la densité minimale autorisée dans un cluster, et ainsi de suite jusqu'à obtenir un ensemble de données

vide. Ce faisant, une valeur Eps est estimée pour chaque nouveau cluster, conduisant à un ensemble de clusters de différentes densités.

Algorithme MDC : Multi-Density datasets Clustering (MDC) **Entrée :** ensemble de données X, k (pour les k-plus proches voisins) et $MinPts$ (peut être 4 ou 5) **Sortie :** Un ensemble de clusters multi-densité
Étape 1 : Calculer la densité de chaque point de données dans l'ensemble de données X en utilisant l'équation (23), stocker le résultat dans le vecteur RO et créer une liste vide 'Clusters'. **Étape 2 :** Trier RO par ordre décroissant de densités. **Étape 3 :** Extraire le premier point de données $p1$ dans RO en tant qu'initiateur de cluster. **Étape 4 :** Construire un nouveau cluster *C = {p1, v1, ..., vk}*, où *v1, …, vk* sont les k-voisins de $p1$, et supprimer ces points de données de RO. **Étape 5 :** À partir de C, construire *D = {d1,d2,…,}*, distances par paires entre tous les deux points de données dans D. **Étape 6 :** À partir de D, calculer l'Eps *local*, en utilisant la formule (27). **Étape 7 :** En utilisant RO, étendre C en utilisant l'Eps local et k, tout en contrôlant la taille du cluster à l'aide de $MinPts$. **Étape 8 :** Ajouter C à 'Clusters', et supprimer de RO tous les points de données dans C. **Étape 9 :** Si RO n'est pas vide, répéter de l'étape 3 à l'étape 9. **Étape 10 :** Clusters de sortie.

Figure 12 : Étapes de l'algorithme MDC proposé

Dans l'étape 1, pour chaque point de données dans l'ensemble de données, il calcule ses distances à partir des k points de données voisins, calcule la densité locale selon l'équation (23) pour chaque point de données, stocke le résultat dans RO et crée une liste vide 'Clusters'. Dans l'étape 2, il trie RO par ordre décroissant de leurs densités locales. Dans l'étape 3, il prend le premier point de données $p1$ (il a la densité la plus élevée) en tant que nouvel

initiateur de cluster. Dans l'étape 4, l'algorithme MDC construit un nouveau cluster $C = \{p1, v1, .., vk\}$, où $p1, v1, .., vk$ sont les k voisins de $p1$, et supprime ces points de RO.

Pour pouvoir rechercher les membres restants du cluster à partir de RO, nous avons besoin d'estimer un Eps local pour ce cluster. Pour cela, à l'étape 5, l'algorithme MDC construit un ensemble initial de distances par paires, (entre tous les deux points de données dans le cluster actuel) et stocke le résultat dans D. Dans l'étape 6, il applique l'inégalité de Chebyshev sur D pour déterminer l'Eps local du cluster actuel à l'aide de l'équation (27).

Dans l'étape 7, il étend le cluster actuel C en recherchant des points de données accessibles à l'aide de l'Eps calculé de la même manière que dans DBSCAN, en supprimant de l'ensemble de données tous les points de données qui composent le cluster actuel. De plus, $MinPts$ est utilisé pour contrôler la taille minimale du cluster. La valeur $MinPts$ peut être 4 ou 5.

Dans l'étape 8, il ajoute C à 'Clusters' et supprime tous les nouveaux points de données classifiés de RO. Dans l'étape 9, il vérifie RO, s'il est vide, il arrête le clustering et exécute l'étape 10 pour générer des clusters, sinon, il initie un nouveau cluster de la même manière et ainsi de suite.

4 Résultats expérimentaux

Afin de valider l'efficacité de l'algorithme MDC proposé, nous avons mené des expériences de clustering d'ensembles de données non seulement sur des ensembles de données synthétiques, mais aussi sur des données réelles, avec des clusters de différentes densités, formes et tailles, en comparant les résultats avec ceux obtenus par DBSCAN et DPC. Les algorithmes sont implémentés dans l'environnement de développement intégré de Python (Python IDE version 3.7.5). Pour DBSCAN, nous avons utilisé la bibliothèque scikit-learn en Python.

Dans DPC, le dc requis est calculé en utilisant la valeur en pourcentage 2% comme conseillé dans [25]. Nous avons recherché la valeur optimale du paramètre de 0,7 à 3,0 avec un pas de 0,3. Afin de simplifier la mise en œuvre de DPC, on sélectionne M points avec les

premières valeurs de $\rho \times \delta$, directement comme centres de cluster. La valeur Eps varie dans la plage de 0,5 à 3,0 avec un pas de 0,3. Les valeurs $MinPts$ varient de 4 à 20.

Ensemble de données	**DBSCAN**		**DPC**		**MDC (proposé)**	
	ARI	F1	ARI	F1	ARI	F1
Flame	0.968	0.987	0.99	1.00	1.00	1.00
Jain	1.00	1.00	0.722	0.923	1.00	1.00
Compound	0.932	0.939	0.630	0.777	1.00	1.00
Aggregation	0.908	0.938	1.00	1.00	0.991	0.996

Tableau 2 : Comparaison à l'aide des métriques ARI et F-mesure

Le tableau 2 montre les mesures de qualité des algorithmes, lorsqu'ils sont appliqués aux ensembles de données de la figure 13. En utilisant deux indices de validité de cluster, F-mesure avec $\alpha = 1$ (F1) et l'indice rand ajusté (ARI), nous évaluons et discutons les performances de ces trois méthodes. La mesure F est la moyenne pondérée (ou moyenne harmonique) de la précision et du rappel. En utilisant le score F1 comme métrique, nous sommes sûrs que si le score F1 est élevé, la précision et le rappel du classificateur indiquent de bons résultats. ARI a la valeur maximale 1, et sa valeur attendue est 0 dans le cas de clusters aléatoires. Un indice rand ajusté plus grand signifie un accord plus élevé entre deux partitions. ARI est recommandé pour mesurer la concordance même lorsque les partitions comparées ont des nombres de clusters différents.

Ci-dessous, nous montrons à travers la figure 13, les résultats de clustering produits par différents algorithmes. L'ensemble de données 'Flame' a deux composants avec des densités similaires mais des formes différentes. Et ils sont très proches l'un de l'autre. MDC et DPC l'ont regroupé correctement, mais DBSCAN a identifié sept valeurs aberrantes comme bruit (Figure 13). 'Jain' est divisé en deux composants avec des densités différentes. DBSCAN, DPC, et MDC ont correctement identifié les deux composants.

Il est difficile de distinguer correctement les composants dans 'Compound', composé de six composants de densités et de formes variables, avec des relations spatiales complexes. On voit dans le coin supérieur gauche, deux composants contigus. Un petit composant en forme de disque est entouré d'un composant en forme d'anneau. Ils sont situés en bas à gauche. Sur la droite, un composant de forme irrégulière est entouré d'un ensemble de points dispersés qui se chevauchent dans l'espace.

Comme le montre la figure 13, MDC a correctement identifié les six clusters de l'ensemble de données composé. Contrairement à DBSCAN et DPC où le premier n'a détecté que cinq clusters, et un certain nombre de points considérés comme du bruit, tandis que le second divisait le composant annulaire en deux parties et fusionnait le composant dispersé avec le composant dense. Dans l'ensemble de données d'agrégation, il y a sept composants avec différentes tailles et formes. À l'exception des deux paires de composants légèrement connectées, les autres composants sont indépendants les uns des autres. DBSCAN regroupe chacune de ces deux paires dans un cluster. MDC et DPC ont produit des résultats corrects.

5 Discussions

Le clustering basé sur la densité procède en calculant la densité de chaque point d'échantillonnage. Les algorithmes DBSCAN et DPC [25] sont parmi les algorithmes de clustering basés sur la densité les plus émergents. Cependant, ces algorithmes de clustering présentent certains défauts.

Dans DBSCAN, les paramètres Eps et $MinPts$ requis sont difficiles à déterminer et seules les valeurs globales de ces paramètres sont utilisées, conduisant à un résultat de clustering inexact d'ensembles de données multi-densités.

Dans l'algorithme DPC, la sélection subjective des centres de cluster à l'aide d'un graphe de décision, et le paramètre dc avec seulement une valeur unique ne gère pas les données multi-densité. Visant le problème de la sélection du paramètre dc de l'algorithme DPC, dans [58] les auteurs ont proposé l'utilisation des paramètres optimisés en fonction de la distance locale des points de données. Cependant, avec des données multi-densité, cela ne peut pas obtenir un effet de clustering stable.

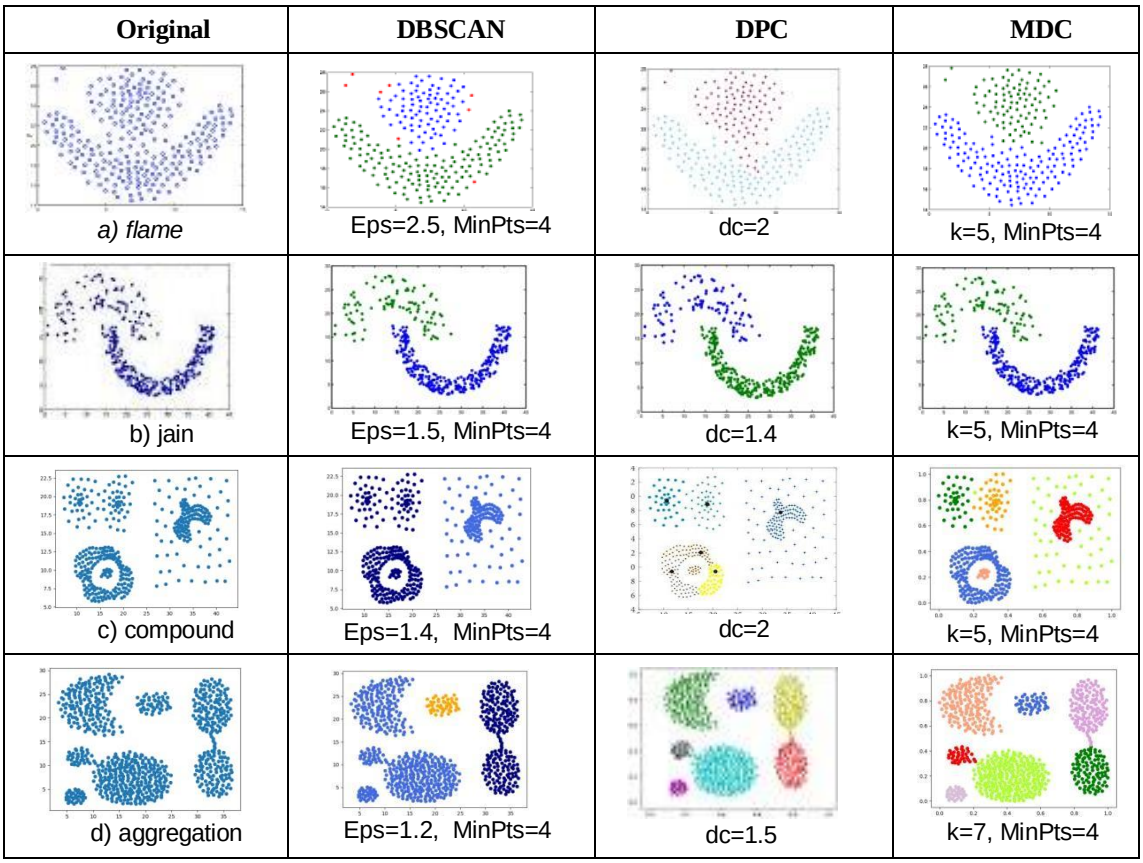


Figure 13. Résultats de clustering de DBSCAN, DPC et MDC à travers a) flame, b) jain, c) compound, et d) agrégation

Pour remédier à ces limitations, notre méthode résout les problèmes mentionnés ci-dessus dans le clustering d'ensembles de données avec différentes densités. Grâce à l'inégalité de Chebyshev, notre approche permet de déterminer automatiquement un seuil de séparation pour chaque classe de densité dans un ensemble de données, et nous construisons chaque cluster avec son seuil de distance associé.

La segmentation d'images par clustering est une méthode de segmentation d'images par regroupement de pixels en fonction de leur similarité ou de leur proximité. Elle constitue une approche puissante pour automatiser les tâches de segmentation d'images et extraire des informations précieuses des données. Grâce à ses avantages en termes d'automatisation, d'identification d'objets, de flexibilité et d'analyse quantitative, la segmentation par clustering trouve diverses applications dans des domaines tels que l'analyse d'images médicales et la gestion de contenu.

La segmentation d'images par clustering s'appuie sur des algorithmes de clustering, tels que le clustering K-means, FC-means, Mean Shift, le clustering par densité, etc. Le but est de

partitionner l'image en régions distinctes aux attributs similaires. En attribuant des pixels à différents clusters, la segmentation par clustering permet d'identifier et d'isoler des objets ou des zones d'intérêt au sein d'une image.

La figure 14 illustre des exemples de segmentation d'images par l'utilisation de notre approche de clustering par densité, présentant une bonne qualité de segmentation en matière de précision et de séparabilité des objets, ce qui facilite l'analyse de l'image.

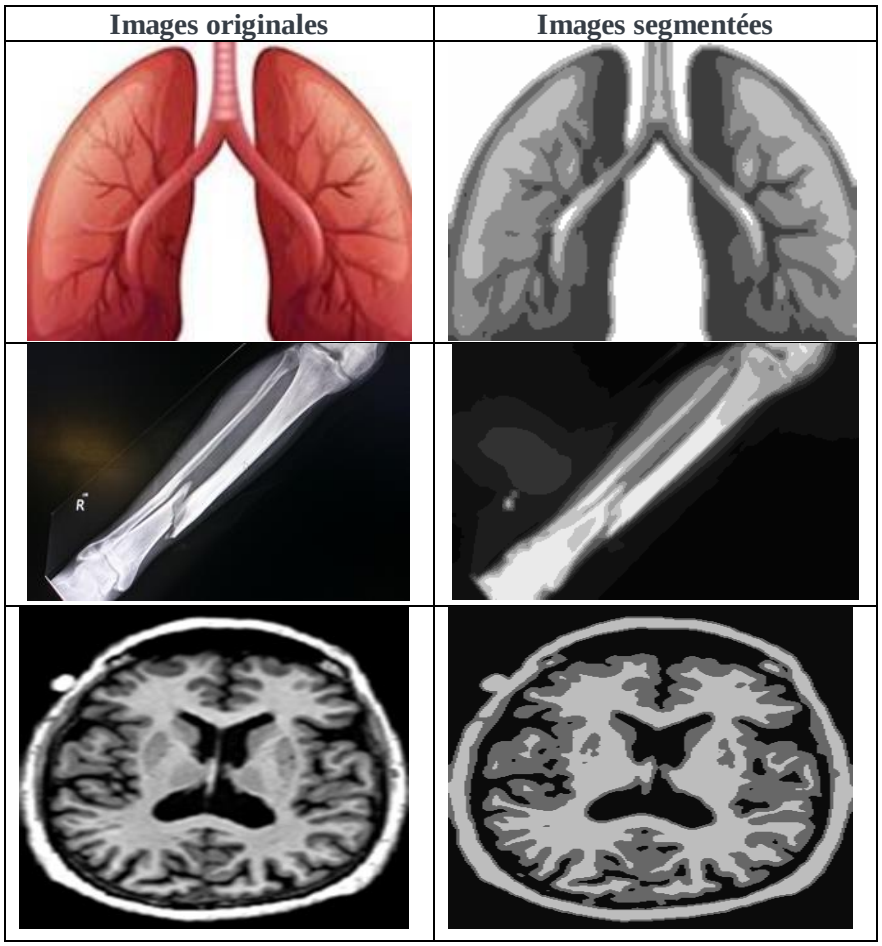


Figure 14. Segmentation d'images utilisant la méthode proposée du clustering par densité

Dans ce travail, nous avons proposé une approche de segmentation d'images utilisant une technique de clustering par densité, robuste au bruit et aux variations des propriétés statistiques des données. Elle permet d'identifier des régions arbitraires, pas nécessairement convexes, de points de données densément peuplés. Le nombre de clusters n'a pas besoin d'être spécifié au préalable ; un cluster est défini comme une région connectée qui dépasse un seuil de densité donné.

6 Conclusion

Ce chapitre montre comment regrouper automatiquement des données en groupes avec plusieurs densités et formes en déterminant plusieurs ensembles de paramètres Eps et $MinPts$. Pour exécuter DBSCAN, les utilisateurs doivent introduire des Eps et des $MinPts$ qui sont difficiles à déterminer. De plus, DBSCAN utilise simplement les Eps et $MinPts$ globaux, de sorte que le résultat du clustering des données multi-densité est inexact.

Pour remédier à ces lacunes, une nouvelle technique de clustering d'ensembles de données permettant de gérer efficacement des clusters de différentes densités et formes est présentée dans ce chapitre. Une modification simple sur DBSCAN a été introduite. L'idée principale repose sur la définition de la densité locale des points de données dans l'ensemble de données et sur l'utilisation de l'inégalité de Chebyshev pour définir un seuil local de séparation à l'étape d'initiation de chaque cluster, lors de l'utilisation de $MinPts$ pour contrôler la densité minimale autorisée dans un cluster.

Les expérimentations menées sur des ensembles de données synthétiques et du monde réel ont montré que l'algorithme proposé apporte des améliorations importantes aux techniques de clustering sur des ensembles de données avec différents niveaux de densité et formes. Dans le cadre de nos travaux futurs, nous prévoyons de nous concentrer sur l'application du MDC à la résolution de problèmes pratiques dans le traitement d'images numériques tels que la segmentation et la détection des contours.

La segmentation d'images est une tâche cruciale en vision par ordinateur qui consiste à diviser une image en segments significatifs pour simplifier ou modifier sa représentation, la rendant plus utile pour l'analyse. Les applications de la segmentation d'images incluent la conduite autonome, l'inspection industrielle, la robotique, l'imagerie médicale, l'analyse d'images satellites, etc.

Dans cette thèse, nous avons abordé le problème de segmentation d'images en utilisant les techniques d'exploration de données, en particulier les outils de clustering. Le clustering est un outil d'apprentissage puissant qui joue un rôle essentiel dans l'exploration de données et est utilisé dans les applications médicales, le climat, les systèmes d'information géographiques, la sécurité et la reconnaissance d'objet, le tracking d'objets, etc.

Basée sur une technique de clustering appropriée, nous avons proposé une méthode de segmentation pouvant déterminer le nombre de clusters et les centres directement en fonction de la densité locale déterminée par la technique des k plus proches voisins (KNN), combinée avec l'inégalité de Chebyshev. Appliquée à la segmentation d'images, l'approche proposée a permis d'obtenir des performances supérieures aux algorithmes de pointe existants.

En effet, l'algorithme proposé pour la segmentation d'images présente les trois principaux avantages suivants : (i) traiter des ensembles de données de densité irrégulière, (ii) regrouper efficacement des ensembles de données non convexes, (iii) une segmentation plus précise des images.

En conclusion, les travaux menés dans cette thèse ont permis de proposer une approche efficace de segmentation d'images basée sur une technique de clustering exploitant la densité locale via les k plus proches voisins (KNN), combinée à l'inégalité de Chebyshev. Les résultats expérimentaux obtenus ont démontré la pertinence de l'approche proposée, notamment en matière de précision de segmentation, de traitement des données de densité irrégulière et de gestion des ensembles de données non convexes.

Malgré ces résultats encourageants, plusieurs perspectives de recherche peuvent être envisagées afin d'améliorer davantage les performances de la méthode proposée. Dans un premier temps, il serait intéressant d'explorer la façon de sélectionner automatiquement la

distribution des écarts entre points de données en fonction de l'image d'entrée, ce qui nous permettra d'obtenir plus de précision des résultats.

Par ailleurs, l'intégration de techniques d'apprentissage profond, notamment les réseaux de neurones convolutifs (CNN), avec les approches traditionnelles de segmentation non supervisée basées sur le clustering constitue une perspective prometteuse. Une telle hybridation pourrait permettre de combiner la puissance de représentation des modèles profonds avec la flexibilité des méthodes de regroupement.

Enfin, l'extension de l'approche proposée à des bases de données plus complexes, notamment dans le domaine de l'imagerie médicale, pourrait permettre d'évaluer davantage sa robustesse et son efficacité dans des contextes réels plus exigeants.

RÉFÉRENCES

[1] Payel Roy, Srijan Goswami, Sayan Chakraborty, Ahmad Taher Azar, Nilanjan Dey. «Image Segmentation Using Rough Set Theory: A Review.» International Journal of Rough Sets and Data Analysis, 2014: 62-74.

[2] Qixiang Ye, Wen Gao, Wei Zeng. «Color Image Segmentation Using Density-Based Clustering.» 2003 International Conference on Multimedia and Expo (ICME '03), 2003: 401-404.

[3] Hong Zhu, Hanzhi He, Jinhui Xu, Qianhao Fang, Wei Wang. «Medical Image Segmentation Using Fruit Fly Optimization and Density Peaks Clustering.» Computational and Mathematical Methods in Medicine, 2018.

[4] Pinaki Pratim Acharjya, Soumya Mukherjee, Dibyendu Ghoshal. «Digital Image Segmentation Using Median Filtering and Morphological Approach.» International Journal of Advanced Research in Computer Science and Software Engineering, 2014.

[5] Nadaradjane Sri Madhava Raja, Steven Lawrence Fernandes, Nilanjan Dey, Suresh Chandra Satapathy, V. Rajinikanth. «Contrast Enhanced Medical MRI Evaluation Using Tsallis Entropy and Region Growing Segmentation.» Journal of Ambient Intelligence and Humanized Computing, 2018: 1603-1614.

[6] M.E. Celebi, Y.A. Aslandogan, P.R. Bergstresser. «Mining Biomedical Images with Density-Based Clustering.» International Conference on Information Technology: Coding and Computing (ITCC), 2005: 163-168.

[7] Jyotsna Rani, Ram Kumar, Fazal A. Talukdar, Nilanjan Dey. «The Brain Tumor Segmentation Using Fuzzy C-Means Technique: A Study.» Recent Advances in Applied Thermal Engineering, 2017: 1-10.

[8] T. Kamali, D. Stashuk. «Automated Segmentation of White Matter Fiber Bundles Using Diffusion Tensor Imaging Data and a New Density Based Clustering Algorithm.» IEEE Transactions on Biomedical Engineering, 2016: 1-10.

[9] Anitha Vishnuvarthanan, M. Pallikonda Rajasekaran, Vishnuvarthanan Govindaraj, Yudong Zhang, Arunprasath Thiyagarajan. « An Automated Hybrid Approach Using Clustering and Nature Inspired Optimization Technique for Improved Tumor and Tissue Segmentation in Magnetic Resonance Brain Images.» Applied Soft Computing, 2017: 1-15.

[10] R. GeethaRamani, Lakshmi Balasubramanian. «Retinal Blood Vessel Segmentation Employing Image Processing and Data Mining Techniques for Computerized Retinal Image Analysis.» Biocybernetics and Biomedical Engineering, 2016: 102-118.

[11] Fionn Murtagh, Pedro Contreras. «Algorithms for Hierarchical Clustering: An Overview.» Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2012: 86-97.

[12] Fahim, Ahmed. «An Extended DBSCAN Clustering Algorithm.» International Journal of Advanced Computer Science and Applications (IJACSA), 2022: 245-258.

[13] Michael Hahsler, Matthew Piekenbrock, Derek Doran. «dbscan: Fast Density-Based Clustering with R.» Journal of Statiscal Software, 2019: 1-30.

[14] Mohammad Mahmudur Rahman Khan, Md. Abu Bakr Siddique, Rezoana Bente Arif, Mahjabin Rahman Oishe. «ADBSCAN: Adaptive Density-Based Spatial Clustering of Applications with Noise for Identifying Clusters with Varying Densities.» 2018.

[15] Atrayee Dhua, Debjani Nath Sarma, Sneha Singh, Bijoyeta Roy. «Segmentation of Images using Density-Based Algorithms.» International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), 2015: 1-5.

[16] Ching-Hsiang Wang, Chia-Hsiu Chen. «Number Plate Recognition from Enhanced Super-Resolution Using Generative Adversarial Network.» Multimedia Tools and Applications, 2022.

[17] Bo Jin, Leandro Cruz, Nuno Gonçalves. «Deep Facial Diagnosis: Deep Transfer Learning from Face Recognition to Facial Diagnosis.» IEEE Access, 2020: 145181-145191.

[18] Tianyuan Yao, Chang Qu, Quan Liu, Ruining Deng, Yuanhan Tian, Jiachen Xu, Aadarsh Jha, Shunxing Bao, Mengyang Zhao, Agnes B. Fogo, Bennett A. Landman, Catie Chang, Haichun Yang, Yuankai Huo. «Compound Figure Separation of Biomedical Images with Side Loss.» 2021.

[19] D. Kaur, Y. Kaur. «Various Image Segmentation Techniques: A Review.» International Journal of Computer Science and Mobile Computing, 2014: 499-504.

[20] Robert M. Haralick, Linda G. Shapiro. «Image Segmentation Techniques: Survey.» Computer Vision, Graphics, and Image Processing, 1985: 100-132.

[21] Keh-Shih Chuang, Hong-Long Tzeng, Sharon Chen, Jay Wu, Tzong-Jer Chen. «Fuzzy C-means Clustering with Spatial Information for Image Segmentation.» Computerized Medical Imaging and Graphics, 2006: 9-15.

[22] Ahmed Elnakib, Georgy L. Gimel'farb, Jasjit S. Suri, Ayman El-Baz. «Medical Image Segmentation: A Brief Survey.» Dans Multi Modality State-of-the-Art Medical Image Segmentation and Registration Methodologies, 1-22. Springer, 2011.

[23] Wei-Xue Liu, Jian Hou. «Study on a Density Peak Based Clustering Algorithm.» 7th International Conference on Intelligent Control and Information Processing (ICICIP). 2016. 292-296.

[24] Lin, Jun-Lin. «Accelerating Density Peak Clustering Algorithm.» Symmetry, 2019: 859.

[25] Alex rodriguez, Alessandro laio. «Clustering by Fast Search and Find of Density Peaks.» Science, 2014: 1492-1496.

[26] Boudraa, Omar. Segmentation d'Images par Coopération Régions-Contours en Utilisant la Théorie des Jeux. Tlemcen: Université de Tlemcen, 2015.

[27] Jérémy Lecoeur, Christian Barillot. Segmentation d'images cérébrales : État de l'art. Inria, 2008.

[28] Chebbi, Imen. «Segmentation d'images par la Technique Régression Linéaire.» 2021.

[29] Nacereddine, Nafâa. Segmentation d'images par Approches Statistiques et Recherche d'images par le Contenu: Application aux images radiographiques de soudures. Alger: Ecole Nationale Polytechnique d'Alger, 2011.

[30] Ahmad El Allaoui, M. Merzougui, M. Nasri, M. El Hitmy, H. Ouariachi. «Segmentation Évolutionniste d'image par Croissance de Région : Application aux Images Médicales.» Internationl Journal of Computer Science and Network Security (IJCSNS), 2010: 325-332.

[31] Neeraj Sharma, Lalit Mohan Aggarwal. «Automated Medical Image Segmentation Techniques.» Journal of Medical Physics, 2010: 3-14.

[32] Sandeep Kumar Dubey, Dr.Sandeep Vijay, Pratibha. «A Review of Image Segmentation using Clustering Methods.» International Journal of Applied Engineering Research, 2018: 2484-2489.

[33] I Gede Made Karma, I Ketut Gede Darma Putra, Made Sudarma, Linawati Linawati. «Image Segmentation Based On Color Dissimilarity.» Informatica, 2022: 645-652.

[34] Lamine Benrais, Nadia Baha Touzene. «Image Segmentation Using Clustering Methods.» Proceedings of the SAI Intelligent Systems Conference (IntelliSys) 2016. 2016. 129-141.

[35] K. Kameshwaran, K. Malarvizhi. «Survey on Clustering Techniques in Data Mining.» International Journal of Computer Science and Information Technologies, 2014: 296-299.

[36] Lam, Chi-Nguyen. Méthodes de Machine Learning pour le suivi de l'occupation du sol des deltas du Viêt-Nam. Brest: Université de Bretagne occidentale, 2021.

[37] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen A. W. M. van der Laak, Bram van Ginneken, Clara I. Sanchez. «A survey on Deep Learning in Medical Image Analysis.» Medical Image Analysis, 2017: 60-88.

[38] Chaussy, Yann. Utilisation d'outils d'intelligence artificielle et d'ontologies pour la segmentation automatique d'images médicales : application au traitement du néphroblastome chez l'enfant. Université Bourgogne Franche-Comté, 2021.

[39] Rupanka Bhuyan, Samarjeet Borah. «A Survey of Some Density Based Clustering Techniques.» 2023.

[40] Ashish Mohan Yadav, Yogiraj Singh Kushawah. «A Survey on Unsupervised Clustering Algorithm based on K-Means Clustering.» International Journal of Computer Applications, 2016: 1-5.

[41] Mahamed G. H. Omran, Andries P. Engelbrecht, Ayed A. Salman. «An Overview of Clustering Methods.» Intelligent Data Analysis, 2007: 583-605.

[42] Hemlata Sahu, Shalini Shrma, Seema Gondhalakar. «A Brief Overview on Data Mining Survey.» International Journal of Computer Technology and Electronics Engineering (IJCTEE), 2011: 114-121.

[43] T. Velmurugan, T. Santhanam. «Implementation Of Fuzzy C-Means Clustering Algorithm for Arbitrary Data Points.» International Journal of Data Mining & Knowledge Management Process (IJDKP), 2011: 1-10.

[44] Marijana Zekić-Sušac, Rudolf Scitovski, Adela Has. «Cluster analysis and artificial neural networks in predicting energy efficiency of public buildings as a cost-saving approach.» Croatian Review of Economic, Business and Social Statistics, 2018: 57-66.

[45] Pan, Yumin. «Different Types of Neural Networks and Applications: Evidence from Feedforward, Convolutional and Recurrent Neural Networks.» Highlights in Science, Engineering and Technology, 2024: 247-255.

[46] Tecuci, Gheorghe. «Artificial Intelligence.» WIREs Computational Statistics, 2012: 168-180.

[47] Ambigavathi Munusamy, D. Sridharan. «Analysis of Clustering Algorithms in Machine Learning for Healthcare Data.» Proceedings of the International Conference on Computational Intelligence and Data Science (ICCIDS 2020), 2020: 1-6.

[48] Yang Yang, Chen Qian, Haomiao Li, Yuchao Gao, Jinran Wu, Chan-Juan Liu, Shangrui Zhao. «An efficient DBSCAN optimized by arithmetic optimization algorithm with opposition-based learning.» The journal of supercomputing, 2022: 19566-19604.

[49] M.Parimala, Daphne Lopez, N.C. Senthilkumar. «A Survey on Density Based Clustering Algorithms for Mining Large Spatial Databases.» International Journal of Advanced Science and Technology, 2011: 60-70.

[50] G. Karypis, E. H. Han, V. Kumar, CHAMELEON: A Hierarchical Clustering Algorithm using Dynamic Modeling, Computer, vol. 32, no. 8, pp. 68-–75, 1999.

[51] Rasmussen CE (1999) The Infinite Gaussian Mixture Model. In: Advances in Neural Information Processing Systems 12, MIT Press, Cambridge, MA, pp 554–560.

[52] Andrew Bryant, Krzysztof Cios. «RNN-DBSCAN: A Density-Based Clustering Algorithm using Reverse Nearest Neighbor Density Estimates.» IEEE Transactions on Knowledge and Data Engineering, 2018: 1109-1121.

[53] N. Soni and A. Ganatra, AGED : Automatic Generation of Eps for DBSCAN, Int. J. of Computer Science and Information Security (IJSIS),, vol. 14, no. 5, pp. 536-559, 2016.

[54] Xiaoyun Chen, Yufang Min, Yan Zhao, Ping Wang. «GMDBSCAN: Multi-Density DBSCAN Cluster Based on Grid.» IEEE International Conference on E- Business Engineering (ICEBE 2008). 2008.

[55] Ahmed Fahim , Clustering Algorithm For MultiDensity Datasets, Romanian Journal of Information Science and Technology, Vol. 22, Number 3–4, 2019, 244–258.

[56] Hennig, Christian. «What are the true clusters?» Pattern Recognition Letters, 2015: 53-62.

[57] Jin Chu Wu, Charles L. Wilson. «Using Chebyshev's Inequality to Determine Sample Size in Biometric Evaluation of Fingerprint Data.» NIST Interagency / Internal Report (NISTIR) 7273, 2005.

[58] Donghua Yu, Guojun Liu, Maozu Guo, Xiaoyan Liu, Shuang Yao. «Density Peaks Clustering Based on Weighted Local Density Sequence and Nearest Neighbor Assignment.» IEEE Access, 2019: 31447-31456.