REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

Université de Mohamed El-Bachir El-Ibrahimi - Bordj Bou Arreridj

Institut d'Electronique et des Télécommunications (IET)

Département d'Electronique

Mémoire

Présenté pour obtenir

LE DIPLOME DE MASTER

FILIERE : **Electronique**

Spécialité : Electronique des systèmes embarqués

Par

- **ZAOUI Fares**
- **ZIANE Abdelbasset**

Intitulé

Reconnaissance d'objets basée sur les histogrammes de couleur : une alternative aux codes-barres pour la gestion automatisée des stocks

Soutenu le : 29 juin 2026

Devant le Jury composé de :

Nom & Prénom	*Grade*	*Qualité*	*Etablissement*
Dr. Nacira DIFFELLAH	*MCA*	*Président*	*Univ-BBA*
Dr. Rabah HAMDINI	*MCB*	*Encadreur*	*Univ-BBA*
Mme. Fouzia HAMMADACHE	*MAA*	*Examinateur*	*Univ-BBA*

Année Universitaire 2025/2026

REMERCIEMENTS

Nous tenons à exprimer notre profonde gratitude à notre encadreur, Dr. Rabah HAMDINI, pour son accompagnement constant, sa disponibilité, sa patience ainsi que la confiance qu'il nous a accordée tout au long de ce travail. Ses précieux conseils, son expertise scientifique et son soutien indéfectible ont constitué une source de motivation essentielle et ont largement contribué à l'aboutissement de ce mémoire.

Nous adressons également nos sincères remerciements aux membres du jury, Mme Nacira DIFFELLAH et Mme Fouzia HAMMADACHE, pour l'honneur qu'elles nous font en acceptant d'évaluer ce travail. Nous leur sommes reconnaissantes pour le temps consacré à l'examen de ce mémoire ainsi que pour leurs remarques et suggestions qui contribueront à son amélioration.

Nos remerciements vont également à nos camarades et amis pour leurs encouragements, leurs échanges constructifs et leur soutien tout au long de cette expérience académique.

Nous exprimons une gratitude particulière à nos familles, ZAOUI et ZIANE, pour leur soutien moral inconditionnel, leur patience et leur confiance permanente. Leurs encouragements ont été une source de force et de persévérance durant tout notre parcours universitaire.

Enfin, nous remercions chaleureusement toutes les personnes qui, de près ou de loin, ont contribué à la réalisation de ce mémoire. Par leurs conseils, leur aide, leurs encouragements ou leur simple présence, elles ont participé à l'aboutissement de ce travail.

Ce mémoire est le fruit d'un travail soutenu, d'une collaboration enrichissante et d'un engagement constant. Nous sommes profondément reconnaissantes envers toutes celles et tous ceux qui nous ont accompagnées dans cette aventure académique.

DÉDICACE

Á mes chers parents,

Á ma famille,

Á mes amis,

Et à tous ceux qui me sont chers.

Je dédie ce mémoire avec toute ma gratitude.

Fares ZAOUI

Á mes très chers parents,

Pour leur amour, leur soutien et leurs sacrifices,

Á ma famille,

Á mes amis,

Et à tous ceux qui m'ont encouragé tout au long de ce parcours.

Je dédie ce mémoire avec une profonde gratitude et une sincère reconnaissance.

Abdelbasset ZIANE

ملخص

يُعَدّ تصنيف الصور من المجالات المهمة في الرؤية الحاسوبية والذكاء الاصطناعي، حيث يهدف إلى التعرف على محتوى الصور وتصنيفها بشكل آلي. يهدف هذا العمل إلى دراسة ومقارنة مجموعة من طرق استخراج الخصائص البصرية واللونية لتحسين أداء أنظمة تصنيف الصور. ولتحقيق ذلك، تم الاعتماد على خمس تقنيات لاستخراج الخصائص هي، AlexNet و U-Net و Vision Transformer (ViT) و CLIP و Mini-Batch K-Means ، مع الاستفادة من واصفات اللون المعتمدة على فضائي غجـ وحصط والمدرجات اللونية.

تم استخدام ثلاثة مصنفات مختلفة هي CNN و SVM و KNN لتقييم فعالية الخصائص المستخرجة. أُجريت التجارب على قاعدتي البيانات COIL-100 و ALOI اللتين تتضمنان صورًا لأجسام مختلفة في ظروف متنوعة من الإضاءة وزوايا الرؤية. وقد أظهرت النتائج تفوق نموذج Vision Transformer في معظم الاختبارات من حيث الدقة والقدرة على التعميم ، بينما حققت U-Net نتائج جيدة بفضل حفاظها على المعلومات المكانية. كما بينت الدراسة أن دمج الخصائص اللونية مع الخصائص المستخرجة بواسطة نماذج التعلم العميق يساهم في تحسين أداء التصنيف. وتؤكد النتائج أهمية اختيار أسلوب استخراج الخصائص المناسب للحصول على أفضل أداء في تطبيقات تصنيف الصور.

كلمات مفتاحية ـ استخراج الخصائص، الذكاء الاصطناعي، التعلم العميق، الرؤية الحاسوبية، تصنيف الصور، واصفات اللون.

Résumé

Ce mémoire présente une étude comparative de plusieurs méthodes d'extraction de caractéristiques pour la classification d'images. L'objectif est d'évaluer l'impact des descripteurs de couleur sur les performances de classification. Cinq approches ont été étudiées : AlexNet, Vision Transformer (ViT), U-Net, CLIP et Mini-Batch K-Means. Les caractéristiques extraites sont combinées à des informations colorimétriques issues des espaces RGB et HSV, puis classifiées à l'aide des algorithmes CNN, SVM et KNN. Les expérimentations sont réalisées sur les bases COIL-100 et ALOI afin d'analyser la robustesse des méthodes face aux variations d'éclairage et de point de vue. Les résultats montrent que Vision Transformer obtient les meilleures performances globales, tandis que U-Net offre également une excellente capacité de généralisation. L'intégration des caractéristiques de couleur améliore significativement la précision de classification.

Mots-clés : AlexNet, Classification d'images, CLIP, CNN, U-Net, Vision Transformer.

Abstract

This master thesis presents a comparative study of several feature extraction methods for image classification. The objective is to evaluate the impact of color descriptors on classification performance. Five approaches were studied : AlexNet, Vision Transformer (ViT), U-Net, CLIP, and Mini-Batch K-Means. The extracted features are combined with colorimetric information derived from the RGB and HSV color spaces, then classified using CNN, SVM, and KNN algorithms. The experiments are conducted on the COIL-100 and ALOI datasets in order to analyze the robustness of the methods against variations in lighting and viewpoint. The results show that Vision Transformer achieves the best overall performance, while U-Net also provides excellent generalization capability. The integration of color features significantly improves classification accuracy.

Keywords : AlexNet, Image Classification, CLIP, CNN, U-Net, Vision Transformer.

TABLE DES MATIÈRES

LISTE DES FIGURES

LISTE DES TABLEAUX

LISTE DES ALGORITHMES

LISTE DES ABRÉVIATIONS

AI	Artificial Intelligence (Intelligence Artificielle)
CNN	Convolutional Neural Network (Réseau de Neurones Convolutifs)
ViT	Vision Transformer
CLIP	Contrastive Language-Image Pre-Training
KNN	K-Nearest Neighbors (K plus proches voisins)
SVM	Support Vector Machine (Machine à Vecteurs de Support)
RGB	Red, Green, Blue
HSV	Hue, Saturation, Value
ReLU	Rectified Linear Unit
LRN	Local Response Normalization
MBKM	Mini-Batch K-Means
HOG	Histogram of Oriented Gradients
SIFT	Scale-Invariant Feature Transform
COIL	Columbia Object Image Library
ALOI	Amsterdam Library of Object Images
GPU	Graphics Processing Unit (Processeur Graphique)
ML	Machine Learning (Apprentissage Automatique)
DL	Deep Learning (Apprentissage Profond)
PDF	Probability Density Function (Fonction de Densité de Probabilité)

INTRODUCTION GÉNÉRALE

L'intelligence artificielle et la vision par ordinateur trouvent de nombreuses applications dans la reconnaissance d'objets, la surveillance, la médecine, les systèmes de recommandation et les véhicules autonomes. L'objectif est d'attribuer automatiquement une catégorie à une image en fonction de son contenu visuel. [1] .

La **couleur** est l'une des principales caractéristiques que l'on peut trouver dans une image. Elle aide à différencier des objets, à reconnaître des textures et à segmenter une image. Les descripteurs de couleur (espaces de couleur RGB et HSV, histogrammes, etc.) sont couramment employés dans les méthodes traditionnelles d'analyse d'image. Cependant, ces approches sont limitées face à la complexité croissante des données visuelles[8].

Avec l'avènement du **deep learning**, de nouvelles approches ont permis d'améliorer considérablement les performances en classification d'images. Les **réseaux de neurones convolutifs (CNN)** ont marqué une avancée significative en apprenant automatiquement des caractéristiques pertinentes à partir des données brutes [3]. Par la suite, des architectures plus récentes comme les **Vision Transformers (ViT)** ont introduit des mécanismes d'attention permettant une compréhension globale de l'image [22]. D'autres modèles comme **CLIP** ont innové en reliant les images au langage naturel [5], tandis que **U-Net** s'est spécialisé dans la segmentation d'images [6]. En parallèle, des modèles non supervisés tels que **K-Means** restent utiles pour l'analyse et l'organisation des données [7].

Cependant, malgré la puissance de ces modèles, la phase de **classification** repose souvent sur des algorithmes dédiés. Les réseaux de neurones convolutifs (CNN) sont une solution de référence car ils sont capables d'apprendre des représentations discriminatives directement à partir des données. Des méthodes classiques comme les **Support Vector Machines (SVM)** et les **k plus proches voisins (KNN)** demeurent pertinentes, notamment lorsqu'elles sont combinées avec des caractéristiques extraites par des modèles profonds.

Dans ce contexte, une question importante se pose :

> Comment comparer et exploiter efficacement différentes méthodes d'extraction de caractéristiques afin d'améliorer la classification d'images basée sur la couleur ?

Objectifs du mémoire

Ce travail a pour objectif principal de proposer une étude comparative de plusieurs méthodes de traitement d'images. Plus précisément, il s'agit de :

- Extraire des caractéristiques visuelles à l'aide de cinq méthodes : AlexNet, Vision Transformer, CLIP, U-Net et K-Means
- Intégrer des descripteurs de couleur pour enrichir la représentation des images
- Appliquer les classificateurs : CNN, SVM et KNN
- Comparer les performances des différentes combinaisons
- Évaluer l'impact des caractéristiques de couleur sur la classification

Contribution du travail

Les principales contributions de ce mémoire sont :

- La mise en œuvre d'un cadre expérimental unifié permettant la comparaison de cinq méthodes d'extraction de caractéristiques (AlexNet, U-Net, Vision Transformer, CLIP et Mini-Batch K-Means).
- L'évaluation comparative de trois classificateurs (CNN, SVM et KNN) sur plusieurs représentations visuelles.
- L'analyse de l'influence des caractéristiques colorimétriques sur les performances de classification.
- L'identification des avantages et des limites de chaque méthode sur les bases COIL-100 et ALOI.

Organisation du mémoire

Le mémoire est structuré en deux chapitres :

- Le premier chapitre présente les concepts fondamentaux, les méthodes utilisées ainsi que les travaux existants dans le domaine de la classification d'images
- Le second chapitre décrit la méthodologie proposée, les expérimentations réalisées ainsi que l'analyse des résultats obtenus

CHAPITRE 1

FONDEMENTS ET MÉTHODES

La classification d'images est un domaine central en vision par ordinateur, visant à attribuer une étiquette ou une catégorie à une image en fonction de son contenu visuel. Avec l'augmentation massive des données visuelles et les avancées en intelligence artificielle, de nombreuses méthodes ont été développées, allant des approches classiques basées sur des descripteurs manuels aux méthodes modernes fondées sur l'apprentissage profond [1] [9].

Ce chapitre présente les concepts fondamentaux nécessaires à la compréhension de ce travail. Il introduit les descripteurs de couleur, les principales méthodes d'extraction de caractéristiques utilisées (AlexNet, Vision Transformer, CLIP, U-Net, K-Means), ainsi que les classificateurs employés (CNN, KNN et SVM).

I Classification d'images

La classification d'images est une tâche fondamentale en vision par ordinateur. Elle consiste à attribuer automatiquement une étiquette sémantique à une image numérique en la faisant correspondre à l'une des catégories d'un ensemble prédéfini.

Formellement, étant donné une image

$$I \in \mathbb{R}^{H \times W \times C} \tag{1.1}$$

et un ensemble de classes, où H représente la hauteur de l'image, W sa largeur et C le nombre de canaux de couleur. Dans le cas d'une image RGB, $C = 3$, tandis que pour une image en niveaux de gris, $C = 1$.

$$\mathcal{C} = \{c_1, c_2, \ldots, c_K\}, \tag{1.2}$$

l'objectif est d'apprendre une fonction de décision

$$f : \mathbb{R}^{H \times W \times C} \to \mathcal{C} \tag{1.3}$$

qui minimise l'erreur de classification sur des données non observées pendant l'entraînement.

Une même catégorie peut présenter une très grande variabilité visuelle liée aux changements d'échelle, d'orientation, d'éclairage, d'arrière-plan ou encore aux occlusions. La robustesse à ces variations constitue l'un des principaux défis de la classification d'images.

L'évolution de ce domaine peut être divisée en deux grandes périodes : les méthodes classiques et les approches d'apprentissage profond.

I.1 Méthodes classiques

Le Tableau 1.1 présente une comparaison entre les méthodes classiques et les approches d'apprentissage profond utilisées dans le domaine de la classification d'images.

Tableau 1.1 – Comparaison entre les méthodes classiques et les approches d'apprentissage profond

Critère	**Méthodes classiques**	**Apprentissage profond**
Extraction de caractéristiques	Manuelle (SIFT, HOG, etc.)	Automatique (end-to-end)
Optimisation	Séquentielle	Conjointe et bout-en-bout
Expressivité des représentations	Limitée par les a priori	Hiérarchique et adaptative
Données nécessaires	Modérée	Grande quantité de données
Interprétabilité	Élevée	Faible à modérée
Performances sur tâches complexes	Limitées	État de l'art
Coût computationnel	Faible	Élevé
Transfert d'apprentissage	Difficile	Très efficace

Avant l'essor des réseaux profonds, la chaîne de traitement standard pour la classification d'images s'articulait autour de deux étapes distinctes et séquentielles : l'extraction de caractéristiques et la décision de classification. Chacune de ces étapes était conçue indépendamment, et la qualité du résultat global dépendait fortement de la pertinence des descripteurs choisis par l'ingénieur.

La phase d'extraction de caractéristiques visait à transformer l'image brute en un vecteur de caractéristiques compact, qui devait idéalement être discriminant entre les classes, invariant aux transformations non pertinentes (rotations, translations, changements d'échelle, variations d'illumination), et suffisamment robuste au bruit. Parmi les descripteurs les plus influents de cette époque, le SIFT (Scale-Invariant Feature Transform) [10] proposé par Lowe en 2004 s'est imposé comme une référence incontournable. Il repose sur la détection de points d'intérêt stables à différentes échelles et orientations, suivie de la construction d'histogrammes de gradients locaux dans des régions centrées sur ces points, produisant des vecteurs de 128 dimensions invariants aux rotations et aux changements d'échelle.

Parallèlement, le descripteur HOG (Histogram of Oriented Gradients) [11], proposé par Dalal et Triggs en 2005 pour la détection de piétons, exploite la distribution des orientations de gradients dans des cellules spatiales régulières, normalisées par blocs pour assurer une robustesse aux variations d'illumination locale. Ces descripteurs manuels offrent une interprétabilité appréciable et des garanties théoriques de robustesse, mais leur capacité représentationnelle est nécessairement limitée par les hypothèses a priori de l'ingénieur sur la nature pertinente de l'information visuelle.

En ce qui concerne la phase de classification, les approches les plus répandues reposaient sur des modèles statistiques bien établis. Les machines à vecteurs de support (SVM)[12], les forêts aléatoires (Random Forests)[13], les réseaux bayésiens naïfs [14] et les méthodes de boosting comme AdaBoost [15] constituaient les briques algorithmiques principales. Ces méthodes présentent l'avantage d'être appuyées par des garanties théoriques solides — notamment en termes de capacité de généralisation — mais leur performance est fondamentalement bornée par la qualité et l'expressivité des descripteurs qui les alimentent.

Un point de rupture fondamental des méthodes classiques est que la représentation de l'image et la prise de décision sont optimisées séparément, sans retour d'information de l'une vers l'autre. Cette absence d'optimisation conjointe se traduit par une sous-performance systématique sur les tâches de grande complexité, où la pertinence des caractéristiques dépend intrinsèquement de la tâche de classification à accomplir. Cette limitation a constitué la motivation principale pour le développement des approches d'apprentissage profond.

I.2 Méthodes deep learning

L'avènement de l'apprentissage profond a fondamentalement redéfini les frontières du possible en vision par ordinateur. La rupture introduite par ces approches est double : d'une part, l'extraction de caractéristiques et la classification sont optimisées de manière conjointe et bout-en-bout (end-to-end) ; d'autre part, les représentations apprises sont organisées en une hiérarchie de concepts de complexité croissante, des bords et contours simples détectés dans les premières couches aux structures complexes et aux parties d'objets dans les couches profondes.

Cette organisation hiérarchique n'est pas le fruit d'une conception explicite, mais émerge naturellement de l'optimisation de la fonction de perte sur des millions d'exemples annotés. La capacité du modèle à découvrir automatiquement les représentations les plus adaptées à la tâche — ce qu'on appelle la « feature learning » ou apprentissage de caractéristiques — constitue l'avantage comparatif décisif des approches profondes sur les méthodes classiques.

La progression des architectures profondes a suivi une trajectoire remarquable. Après le succès d'AlexNet en 2012 [3], VGGNet (Visual Geometry Group Network) [16] a démontré que la profondeur — mesurée par l'empilement de convolutions 3×3 — est un facteur critique de performance. GoogLeNet [17] a introduit les modules d'inception pour explorer simultanément différentes échelles de filtrage. ResNet [18] a résolu le problème de la disparition du gradient dans les réseaux très profonds grâce aux connexions résiduelles, permettant d'entraîner des réseaux de plus de 150 couches.

Plus récemment, les architectures basées sur les mécanismes d'attention multi-têtes, initialement développées pour le traitement du langage naturel, ont été adaptées avec succès à la vision par ordinateur, donnant naissance aux Vision Transformers [22] et aux modèles multimodaux de type CLIP [5].

Un autre aspect fondamental de l'apprentissage profond est le recours massif au transfert d'apprentissage (transfer learning). Plutôt que d'entraîner un modèle de zéro sur un jeu de données spécifique — souvent trop limité — on part de poids pré-entraînés sur un large corpus généraliste (comme ImageNet) et on affine (fine-tune) le modèle sur la tâche cible. Cette pratique permet d'exploiter les représentations génériques apprises sur des millions d'images et constitue désormais la norme dans la plupart des applications industrielles et académiques de la vision par ordinateur.

II Descripteurs de couleur

La couleur représente une caractéristique visuelle importante permettant de décrire efficacement le contenu d'une image.

Les espaces colorimétriques les plus utilisés dans ce travail sont RGB et HSV.

II.1 Espace colorimétrique RGB

L'espace RGB (*Red, Green, Blue*) repose sur la synthèse additive de la lumière. Chaque pixel est décrit par trois composantes : (R, G, B)
avec :

$$R, G, B \in [0, 255]. \tag{1.4}$$

Le nombre total de couleurs représentables est :

$$256^3 = 16\,777\,216. \tag{1.5}$$

Malgré sa simplicité, RGB présente plusieurs limitations :
— forte corrélation entre les canaux ;
— sensibilité aux variations d'éclairage ;
— faible cohérence avec la perception humaine.

II.2 Espace colorimétrique HSV

L'espace HSV (*Hue, Saturation, Value*) sépare l'information chromatique de la luminosité.

Ses composantes sont :
— H : teinte ;
— S : saturation ;
— V : luminosité.

Cette représentation facilite la segmentation basée sur la couleur et améliore la robustesse aux variations d'éclairage. Le tableau 1.2 résume les principales caractéristiques des espaces colorimétriques RGB et HSV, en mettant en évidence leurs avantages et leurs limites dans les applications de traitement et d'analyse d'images.

Tableau 1.2 – Comparaison entre RGB et HSV

Propriété	**RGB**	**HSV**
Composantes	R, G, B	H, S, V
Type de modèle	Additif	Cylindrique
Robustesse à l'éclairage	Faible	Élevée
Corrélation inter-canaux	Forte	Faible
Interprétabilité perceptuelle	Moyenne	Élevée
Applications	Acquisition, affichage	Segmentation, détection

II.3 Histogrammes de couleur

Les histogrammes de couleur décrivent la distribution statistique des couleurs dans une image.

Pour un histogramme comportant B classes, la valeur du bin b est définie par :

$$H_b = \text{card}\left\{p \in \Omega \ : \ I(p) \in [b\Delta, (b+1)\Delta[\right\}, \quad b = 0, \ldots, B-1 \tag{1.6}$$

où :
— Ω représente le domaine spatial de l'image ;
— $I(p)$ la valeur du pixel p ;
— Δ la largeur d'un intervalle ;
— B le nombre de bins.

L'histogramme normalisé est donné par :

$$h(b) = \frac{H(b)}{|\Omega|}. \tag{1.7}$$

Une mesure de similarité fréquemment utilisée est la distance du Chi-deux :

$$\chi^2(h_1, h_2) = \sum_b \frac{(h_1(b) - h_2(b))^2}{h_1(b) + h_2(b)}. \tag{1.8}$$

Les histogrammes présentent plusieurs avantages :
— simplicité de calcul ;
— invariance aux rotations et translations ;
— indépendance vis-à-vis de la résolution spatiale.

Cependant, ils ne conservent aucune information spatiale sur la localisation des couleurs dans l'image.

III Méthodes d'extraction

Cette section présente les principales méthodes d'extraction de caractéristiques utilisées dans ce travail.

III.1 Mini-Batch K-Means : Clustering à l'Échelle du Web

L'algorithme K-Means [7] constitue l'une des méthodes de partitionnement non supervisé les plus utilisées dans les applications à grande échelle telles que le regroupement de résultats de recherche, la détection de doublons ou l'agrégation de contenus web. Cependant, la version classique du K-Means par lots (*batch K-Means*) devient rapidement coûteuse lorsque la taille des données augmente.

Afin de rendre cette approche scalable, Sculley (2010) propose l'algorithme **Mini-Batch K-Means**, reposant sur l'utilisation de sous-ensembles aléatoires de données appelés *mini-batchs*. Cette méthode réduit considérablement le coût computationnel tout en conservant une qualité de clustering proche du K-Means classique.

III.1.1 Problème d'optimisation

L'objectif du K-Means est de trouver un ensemble de centroïdes

$$C = \{c_1, c_2, \ldots, c_k\} \tag{1.9}$$

minimisant l'inertie intra-cluster :

$$J(C) = \sum_{x \in X} \|f(C, x) - x\|^2 \tag{1.10}$$

où $f(C, x)$ représente le centroïde le plus proche du point x selon la distance euclidienne.

III.1.2 Mini-Batch K-Means

Contrairement au K-Means classique qui utilise l'ensemble complet des données à chaque itération, Mini-Batch K-Means sélectionne aléatoirement un mini-lot de taille b afin de mettre à jour les centroïdes de manière incrémentale.

Algorithme 1 Mini-Batch K-Means

Require: Ensemble de données X, nombre de clusters k, taille du mini-lot b, nombre d'itérations t

Ensure: Ensemble des centroïdes C

Initialiser aléatoirement les centroïdes C
$v[c] \leftarrow 0 \quad \forall c \in C$
for $i = 1$ à t **do**
    Sélectionner un mini-lot M de taille b
    for chaque $x \in M$ **do**
        Trouver le centroïde le plus proche :

$$d[x] \leftarrow \arg\min_{c \in C} \|x - c\|^2$$

    end for
    for chaque $x \in M$ **do**
        $c \leftarrow d[x]$
        $v[c] \leftarrow v[c] + 1$
        $\eta \leftarrow \frac{1}{v[c]}$
        Mise à jour du centroïde :

$$c \leftarrow (1 - \eta)c + \eta x$$

    end for
end for

Cette approche combine les avantages du traitement par lots et de la descente de gradient stochastique (*SGD*) :

— faible coût computationnel ;
— convergence rapide ;
— bonne qualité de clustering ;
— adaptation aux très grands jeux de données.

III.1.3 Centroïdes creux et projection L_1

Dans les applications web à grande échelle, le stockage et la transmission des centroïdes peuvent devenir coûteux. Afin de réduire ce coût, Sculley introduit une projection sur la boule L_1 permettant d'obtenir des centroïdes creux (*sparse centroids*).

L'objectif est de projeter un vecteur c dans une boule L_1 de rayon λ.

Algorithme 2 Projection ε-précise sur la boule L_1

Require: Tolérance ε, rayon λ, vecteur $c \in \mathbb{R}^m$
if $\|c\|_1 \leq \lambda + \varepsilon$ **then**
  return c
end if
$upper \leftarrow \|c\|_\infty$
$lower \leftarrow 0$
while $(current > \lambda(1+\varepsilon)) \vee (current < \lambda)$ **do**
  $\theta \leftarrow \frac{upper + lower}{2}$
  $current \leftarrow \sum_i \max(0, |c_i| - \theta)$
  if $current \leq \lambda$ **then**
    $upper \leftarrow \theta$
  else
    $lower \leftarrow \theta$
  end if
end while
for $i = 1$ to m **do**
  $c_i \leftarrow \text{sign}(c_i) \max(0, |c_i| - \theta)$
end for

III.1.4 Performances

Sur le corpus RCV1 contenant plus de 780 000 documents, Mini-Batch K-Means converge plusieurs ordres de grandeur plus rapidement que le K-Means classique tout en conservant une excellente qualité de clustering.

L'ajout de la projection L_1 permet également de réduire fortement le nombre de composantes non nulles des centroïdes avec une perte minimale de performance.

III.1.5 Conclusion

Mini-Batch K-Means représente une solution efficace et scalable pour le clustering de données massives. Grâce à l'utilisation de mini-lots et de centroïdes creux, cette méthode est particulièrement adaptée aux contraintes computationnelles des applications web modernes.

Les méthodes de clustering comme Mini-Batch K-Means ont joué un rôle important dans l'analyse de données à grande échelle. Cependant, ces approches reposent principalement sur des représentations peu expressives, ce qui limite leurs performances sur des données complexes telles que les images.

L'émergence du deep learning a profondément transformé la vision par ordinateur grâce à des modèles capables d'apprendre automatiquement des représentations hiérarchiques directement à partir des données brutes. Parmi ces architectures, les réseaux de neurones convolutifs (*Convolutional Neural Networks – CNN*) se sont imposés comme une référence majeure.

Le tournant décisif survient en 2012 avec AlexNet, premier réseau convolutif profond à surpasser largement les méthodes traditionnelles sur le défi ImageNet.

Cette avancée marque le début de l'essor moderne du deep learning en vision par ordinateur.

III.2 Les Réseaux de Neurones Convolutifs et AlexNet

Les réseaux de neurones convolutifs (*Convolutional Neural Networks – CNN* [3]) constituent l'une des architectures fondamentales de la vision par ordinateur moderne. Leur succès repose sur leur capacité à apprendre automatiquement des caractéristiques visuelles hiérarchiques à partir des images.

Contrairement aux réseaux entièrement connectés classiques, les CNN exploitent deux propriétés essentielles :

— les connexions locales ;
— le partage des poids (*weight sharing*).

Ces mécanismes permettent de réduire considérablement le nombre de paramètres tout en conservant les structures spatiales importantes des images.

III.2.1 Convolution

Une couche convolutive applique plusieurs filtres appris sur l'image d'entrée afin d'extraire des motifs locaux tels que les contours, textures et formes.

La sortie du filtre j à la position (x, y) est donnée par :

$$S_j(x, y) = \sum_c \sum_s \sum_t W_j(c, s, t) \cdot A(c, x + s, y + t) + b_j \tag{1.11}$$

où :

— A représente l'entrée ;
— W_j correspond au filtre appris ;
— b_j est le biais associé.

III.2.2 Fonction d'activation ReLU

AlexNet utilise la fonction d'activation ReLU (*Rectified Linear Unit*) définie par :

$$f(x) = \max(0, x) \tag{1.12}$$

Cette fonction accélère l'apprentissage et réduit les problèmes de gradients faibles observés avec les fonctions saturantes traditionnelles.

III.2.3 Architecture AlexNet

AlexNet, proposé par Krizhevsky, Sutskever et Hinton en 2012, est composé de :

— 5 couches convolutives ;
— 3 couches entièrement connectées ;
— une couche softmax finale.

Le réseau contient environ :

— 60 millions de paramètres ;
— 650 000 neurones.

L'entrée du modèle est une image RGB de taille :

$$224 \times 224 \times 3 \tag{1.13}$$

L'entraînement a été réalisé sur le dataset ImageNet contenant plus de 1,2 million d'images annotées.

III.2.4 Pooling

AlexNet utilise le *max-pooling* avec :
— une fenêtre 3×3 ;
— un stride de 2.
Le pooling réduit progressivement la dimension spatiale des représentations tout en limitant le surapprentissage.

III.2.5 Normalisation de Réponse Locale

Afin d'améliorer la généralisation, AlexNet applique une normalisation locale définie par :

$$b^i_{x,y} = \frac{a^i_{x,y}}{\left(k + \alpha \sum_{j=\max(0,i-n/2)}^{\min(N-1,i+n/2)} (a^j_{x,y})^2\right)^\beta} \tag{1.14}$$

avec :

$$k = 2, \quad n = 5, \quad \alpha = 10^{-4}, \quad \beta = 0.75$$

III.2.6 Dropout

Pour réduire le surapprentissage, AlexNet applique le *dropout* dans les couches entièrement connectées avec une probabilité :

$$p = 0.5$$

III.2.7 Augmentation des données

Plusieurs techniques d'augmentation de données sont utilisées :
— extraction aléatoire de patches ;
— réflexions horizontales ;
— perturbation des couleurs RGB via PCA.
La perturbation PCA est définie par :

$$[p_1, p_2, p_3][\alpha_1\lambda_1, \alpha_2\lambda_2, \alpha_3\lambda_3]^T \tag{1.15}$$

où :

$$\alpha_i \sim \mathcal{N}(0, 0.1)$$

III.2.8 Algorithme d'entraînement d'AlexNet

Algorithme 3 Entraînement d'AlexNet

Require: Dataset ImageNet (X, Y)
Ensure: Modèle entraîné
1: Initialiser les poids avec une loi gaussienne
2: **for** chaque époque **do**
3: **for** chaque mini-batch **do**
4: Propagation avant :

$$Conv \rightarrow ReLU \rightarrow LRN \rightarrow MaxPool \rightarrow FC \rightarrow Dropout \rightarrow Softmax$$

5: Calcul de la perte cross-entropy
6: Rétropropagation
7: Mise à jour avec SGD :

$$momentum = 0.9, \quad weight\ decay = 5 \times 10^{-4}$$

8: **end for**
9: **end for**

III.2.9 Performances d'AlexNet

AlexNet remporte le défi ILSVRC 2012 avec un taux d'erreur Top-5 de [3] :

$$15.3\%$$

Cette performance marque une rupture majeure avec les approches traditionnelles de vision par ordinateur.

III.2.10 Conclusion

AlexNet marque le début de l'ère moderne du deep learning en vision par ordinateur. Son succès a ouvert la voie à des architectures plus profondes et performantes telles que VGG, ResNet et EfficientNet. Bien qu'AlexNet ait démontré l'efficacité des réseaux convolutifs pour la classification d'images, cette approche reste limitée aux tâches attribuant une seule étiquette à une image entière.

De nombreuses applications nécessitent cependant une compréhension plus fine de la scène visuelle, notamment la localisation précise des objets au niveau du pixel. C'est le cas de la segmentation sémantique, tâche essentielle en imagerie médicale, en conduite autonome ou encore en analyse satellitaire.

Afin de répondre à ce besoin, plusieurs architectures de segmentation ont été développées. Parmi elles, U-Net, proposé en 2015 par Ronneberger, Fischer et Brox, est devenu une référence majeure grâce à sa capacité à produire des segmentations précises même avec un faible volume de données annotées.

III.3 U-Net : Architecture de Segmentation Sémantique

La segmentation sémantique consiste à attribuer une classe à chaque pixel d'une image afin de produire une carte de segmentation complète.

U-Net [6] est une architecture de réseau convolutif entièrement dédiée à cette tâche. Initialement conçue pour l'imagerie biomédicale, elle est aujourd'hui largement utilisée dans de nombreux domaines de la vision par ordinateur.

L'architecture de U-Net repose sur une structure symétrique en forme de « U » composée de deux parties :

— un chemin contractant (*encoder*) ;
— un chemin expansif (*decoder*).

III.3.1 Chemin contractant

Le chemin contractant suit l'architecture classique des CNN.

À chaque niveau :

— deux convolutions 3×3 sont appliquées ;
— chaque convolution est suivie d'une activation ReLU ;
— un max-pooling 2×2 réduit ensuite la résolution spatiale.

Le nombre de cartes de caractéristiques double progressivement :

$$64 \to 128 \to 256 \to 512 \to 1024$$

Cette stratégie permet au réseau de capturer des représentations de plus en plus abstraites.

III.3.2 Bottleneck

Le centre de l'architecture correspond au *bottleneck*, contenant les représentations les plus riches sémantiquement avec 1024 cartes de caractéristiques.

III.3.3 Chemin expansif

Le chemin expansif reconstruit progressivement la résolution spatiale perdue lors de l'encodage.

Chaque étape comprend :

— une up-convolution 2×2 ;
— une concaténation avec les cartes provenant de l'encodeur ;
— deux convolutions 3×3 suivies de ReLU.

Les connexions entre encodeur et décodeur sont appelées *skip connections*. Elles permettent de conserver les informations fines de localisation.

III.3.4 Fonction de perte pondérée

Afin de mieux séparer les objets adjacents, U-Net utilise une fonction de perte pondérée pixel par pixel.

La carte de poids est définie par :

$$w(x) = w_c(x) + w_0 \exp\left(-\frac{(d_1(x) + d_2(x))^2}{2\sigma^2}\right) \tag{1.16}$$

où :

— $w_c(x)$ compense le déséquilibre entre classes ;
— $d_1(x)$ est la distance à la cellule la plus proche ;
— $d_2(x)$ est la distance à la seconde cellule la plus proche.

La fonction de perte globale correspond à une entropie croisée pondérée :

$$E = -\sum_{x \in \Omega} w(x) \log(p_{\ell(x)}(x)) \tag{1.17}$$

III.3.5 Initialisation des poids

U-Net utilise une initialisation de He afin de limiter les problèmes de gradients :

$$Var(W) = \frac{2}{N} \tag{1.18}$$

où N représente le nombre de connexions entrantes.

III.3.6 Augmentation des données

Comme les données annotées sont souvent limitées en imagerie biomédicale, U-Net repose fortement sur l'augmentation des données :

— rotations ;
— translations ;
— variations d'intensité ;
— déformations élastiques.

Ces transformations améliorent fortement la capacité de généralisation du réseau.

III.3.7 Algorithme de segmentation U-Net

Algorithme 4 Segmentation avec U-Net

Require: Image biomédicale I, masques de segmentation G
Ensure: Carte de segmentation S
1: Prétraitement des données :
   Normalisation
   Redimensionnement
   Augmentation des données
2: Construire l'architecture U-Net
3: Encodeur :
   Deux convolutions 3×3 + ReLU
   Max-pooling 2×2
4: Bottleneck
5: Décodeur :
   Up-convolution 2×2
   Skip connections
   Deux convolutions 3×3 + ReLU
6: Convolution finale 1×1 + Softmax
7: Entraînement avec SGD
8: Produire la carte de segmentation finale S

III.3.8 Performances

U-Net obtient d'excellentes performances sur plusieurs benchmarks biomédicaux.

Sur le défi ISBI de segmentation cellulaire, le modèle atteint des scores très élevés tout en utilisant relativement peu de données annotées.

Cette efficacité explique son adoption massive dans :

— la segmentation tumorale ;
— l'analyse histopathologique ;
— l'imagerie microscopique ;
— la segmentation d'organes médicaux.

III.3.9 Conclusion

Grâce à son architecture encodeur-décodeur et à ses connexions de saut (*skip connections*), U-Net permet de préserver les informations spatiales tout en réalisant une localisation précise des structures présentes dans l'image. Cette architecture constitue aujourd'hui l'une des méthodes de référence pour les tâches de segmentation sémantique.

Dans le cadre de ce mémoire, U-Net n'est pas employé comme modèle de classification. Il est utilisé comme extracteur de caractéristiques afin de générer des représentations visuelles discriminantes. Les caractéristiques extraites à partir de l'encodeur ou de la couche de goulot d'étranglement (*bottleneck*) sont ensuite exploitées comme vecteurs descriptifs pour la phase de classification des images.

III.4 Vision Transformer (ViT)

Le Vision Transformer (ViT) [22] constitue une architecture de vision par ordinateur reposant entièrement sur les mécanismes d'attention introduits dans les Transformers.

Contrairement aux CNN, ViT n'utilise pas de convolutions. Les images sont traitées comme des séquences de patches, de manière similaire aux séquences de mots en traitement du langage naturel.

III.4.1 Découpage en patches

Une image d'entrée :

$$x \in \mathbb{R}^{H \times W \times C} \tag{1.19}$$

est divisée en patches de taille :

$$P \times P \tag{1.20}$$

Le nombre total de patches est :

$$N = \frac{HW}{P^2} \tag{1.21}$$

Chaque patch est ensuite aplati en un vecteur puis projeté dans un espace latent de dimension D.

III.4.2 Patch Embeddings

Les patches sont projetés linéairement grâce à une matrice :

$$E \in \mathbb{R}^{(P^2C) \times D} \tag{1.22}$$

Les embeddings obtenus forment une séquence similaire à une phrase en NLP.

Un token spécial de classification $[CLS]$ est ajouté au début de la séquence.

Des embeddings positionnels sont également ajoutés afin de conserver l'information spatiale :

$$E_{pos} \in \mathbb{R}^{(N+1) \times D} \tag{1.23}$$

L'entrée finale du Transformer est :

$$Z_0 = [x_{class}; x_p^1 E; x_p^2 E; \ldots; x_p^N E] + E_{pos} \tag{1.24}$$

III.4.3 Encodeur Transformer

Le modèle est composé de L blocs Transformer identiques.

Chaque bloc contient :

— une couche de Multi-Head Self-Attention (MSA) ;

— un réseau MLP ;

— des connexions résiduelles ;
— une normalisation de couche (*LayerNorm*).

Les équations principales sont :

$$Z'_l = MSA(LN(Z_{l-1})) + Z_{l-1} \tag{1.25}$$

$$Z_l = MLP(LN(Z'_l)) + Z'_l \tag{1.26}$$

III.4.4 Mécanisme d'attention

Pour chaque tête d'attention, les embeddings sont projetés en :
— requêtes Q ;
— clés K ;
— valeurs V.

L'attention est calculée par :

$$Attention(Q, K, V) = Softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1.27}$$

où :

$$d_k = \frac{D}{H} \tag{1.28}$$

correspond à la dimension des clés.

Ce mécanisme permet à chaque patch d'interagir directement avec tous les autres patches de l'image.

III.4.5 Variantes du Vision Transformer

Les principales variantes proposées sont :

Modèle	Couches	Dimension	Têtes	Paramètres
ViT-Base	12	768	12	86M
ViT-Large	24	1024	16	307M
ViT-Huge	32	1280	16	632M

Tableau 1.3 – Principales variantes du Vision Transformer

III.4.6 Algorithme du Vision Transformer

Algorithme 5 Vision Transformer (ViT)

Require: Image x, taille du patch P, nombre de couches L
Ensure: Classe prédite y

Diviser l'image en patches $P \times P$
Aplatir chaque patch
Projection linéaire des patches
Ajouter le token $[CLS]$
Ajouter les embeddings positionnels
for $l = 1$ à L **do**
    Multi-Head Self-Attention
    MLP + connexions résiduelles
end for
Extraire la représentation finale du token $[CLS]$
Calculer :

$$y = Softmax(WZ_L + b)$$

Retourner y

III.4.7 Complexité computationnelle

Le mécanisme d'attention possède une complexité quadratique :

$$O(N^2) \tag{1.29}$$

où N représente le nombre de patches.

Cette propriété permet de capturer des dépendances globales mais augmente fortement le coût computationnel lorsque la résolution des images devient importante.

III.4.8 Performances

Lorsqu'il est pré-entraîné sur de très grands jeux de données comme ImageNet-21k ou JFT-300M, ViT atteint des performances comparables ou supérieures aux CNN modernes [22].

Le modèle obtient notamment :

— 88.55% Top-1 sur ImageNet ;
— 94.55% sur CIFAR-100 ;
— d'excellents résultats sur plusieurs benchmarks de vision.

III.4.9 Conclusion

Le Vision Transformer représente une rupture majeure dans l'évolution de la vision par ordinateur. En remplaçant les convolutions par des mécanismes d'attention globale, ViT ouvre la voie à une nouvelle génération d'architectures de vision basées sur les Transformers.

III.5 CLIP : Contrastive Language-Image Pre-Training

CLIP (*Contrastive Language-Image Pre-Training*) [5] est un modèle multimodal développé par OpenAI permettant d'apprendre conjointement des représentations visuelles et textuelles à partir de données collectées sur Internet.

Contrairement aux approches supervisées classiques, CLIP apprend directement à partir de paires :

$$(image,\ texte)$$

Cette stratégie permet au modèle d'acquérir des représentations visuelles générales sans nécessiter d'étiquettes manuelles spécifiques.

III.5.1 Architecture générale

CLIP est composé de deux encodeurs :
— un encodeur d'images f_I ;
— un encodeur de texte f_T.
L'encodeur visuel peut être :
— un ResNet modifié ;
— un Vision Transformer (ViT).
L'encodeur textuel repose sur une architecture Transformer similaire à celles utilisées en traitement du langage naturel.

Les deux encodeurs projettent leurs représentations dans un espace d'embedding multimodal partagé.

III.5.2 Embeddings multimodaux

Soient :
— I_f les représentations visuelles ;
— T_f les représentations textuelles.
Les embeddings normalisés sont obtenus par :

$$I_e = \frac{W_I I_f}{\|W_I I_f\|_2} \tag{1.30}$$

$$T_e = \frac{W_T T_f}{\|W_T T_f\|_2} \tag{1.31}$$

où :
— W_I et W_T sont des matrices de projection apprises ;
— les vecteurs sont normalisés selon la norme L_2.

III.5.3 Apprentissage contrastif

L'objectif de CLIP consiste à rapprocher les embeddings des paires correctes image-texte tout en éloignant les paires incorrectes.

La matrice de similarité est définie par :

$$logits(I, T) = I_e T_e^{\top} \cdot \exp(\tau) \tag{1.32}$$

où :

$$\tau$$

est un paramètre de température appris pendant l'entraînement.

La fonction de perte contrastive symétrique est donnée par :

$$L = \frac{L_{image \to texte} + L_{texte \to image}}{2} \tag{1.33}$$

III.5.4 Transfert zéro-shot

L'une des propriétés les plus remarquables de CLIP est sa capacité de classification zéro-shot.

Pour classer une image, il suffit de :

— générer une description textuelle pour chaque classe ;
— comparer les embeddings texte-image ;
— sélectionner la similarité maximale.

Par exemple :

$$\text{“}A\ photo\ of\ a\ dog\text{”}$$

Cette approche permet de réaliser des tâches de classification sans réentraînement spécifique.

III.5.5 Prompt Engineering

Les performances de CLIP dépendent fortement de la formulation des prompts textuels.

Le prompt classique est :

$$\text{“}A\ photo\ of\ a\ \{label\}\text{”}$$

L'utilisation de plusieurs formulations contextuelles (*prompt ensembling*) améliore significativement les performances.

III.5.6 Algorithme de pré-entraînement CLIP

Algorithme 6 Pré-entraînement contrastif de CLIP

Require: Mini-batch de N paires (I_i, T_i)
Ensure: Modèle CLIP entraîné
1: Initialiser les encodeurs f_I et f_T
2: **for** chaque mini-batch **do**
3: Encoder les images :
$$I_f \leftarrow f_I(I)$$
4: Encoder les textes :
$$T_f \leftarrow f_T(T)$$
5: Projeter et normaliser les embeddings
6: Calculer la matrice de similarité
7: Calculer la perte contrastive
8: Mettre à jour les paramètres avec AdamW
9: **end for**
10: Retourner les encodeurs entraînés

III.5.7 Algorithme de classification zéro-shot

Algorithme 7 Classification zéro-shot avec CLIP

Require: Image requête I, ensemble de classes $\{c_1, \ldots, c_n\}$
Ensure: Classe prédite $\hat{y}$
1: **for** chaque classe c_i **do**
2: Générer le prompt :
$$p_i = \text{“}A\ photo\ of\ a\ c_i\text{”}$$
3: Encoder le texte :
$$T_i \leftarrow f_T(p_i)$$
4: **end for**
5: Encoder l'image :
$$I_e \leftarrow f_I(I)$$
6: Calculer les similarités cosinus
$$s_i = cosine(I_e, T_i)$$
7: Prédire :
$$\hat{y} = \arg\max_i s_i$$
8: Retourner $\hat{y}$

III.5.8 Performances

CLIP a été évalué sur plus de 30 benchmarks couvrant :

— la classification d'images ;
— la reconnaissance d'actions ;
— l'OCR ;
— la reconnaissance de scènes ;
— la géolocalisation.

En mode zéro-shot, CLIP obtient des performances comparables à des modèles supervisés entraînés directement sur ImageNet.

Ces résultats démontrent la puissance de l'apprentissage multimodal à grande échelle.

III.5.9 Conclusion

CLIP représente une avancée majeure dans l'intégration du langage naturel et de la vision par ordinateur. Grâce à l'apprentissage contrastif multimodal, le modèle apprend des représentations visuelles génériques, robustes et facilement transférables vers de nombreuses tâches sans réentraînement spécifique.

IV Classificateurs

IV.1 Support Vector Machine (SVM)

Le *Support Vector Machine* (SVM) [12] est une méthode de classification supervisée largement utilisée en apprentissage automatique.

Son objectif est de séparer les différentes classes à l'aide d'une frontière de décision optimale. Cette frontière est choisie de manière à maximiser la distance entre les classes. Les échantillons les plus proches de cette frontière sont appelés *vecteurs de support*.

Pour un problème de classification binaire, la fonction de décision est définie par :

$$f(x) = \text{sign}(w^T x + b) \tag{1.34}$$

où :
— w représente le vecteur associé à la frontière de décision ;
— b correspond au biais ;
— x est le vecteur de caractéristiques de l'échantillon.

Lorsque les données ne sont pas séparables de manière linéaire, le SVM peut utiliser une fonction noyau afin de projeter les données dans un espace de dimension plus élevée. Cette transformation facilite la séparation des différentes classes.

Les noyaux les plus utilisés sont :
— le noyau linéaire ;
— le noyau polynomial ;
— le noyau gaussien (*Radial Basis Function*, RBF).

Le SVM offre généralement de bonnes performances lorsque le nombre d'exemples d'apprentissage est limité. Il est également efficace lorsque les caractéristiques extraites sont suffisamment discriminantes pour distinguer les différentes classes.

IV.2 K-Nearest Neighbors (KNN)

Le *K-Nearest Neighbors* (KNN) [19] est un algorithme de classification supervisée fondé sur la proximité entre les échantillons dans l'espace des caractéristiques.

Contrairement à de nombreuses méthodes d'apprentissage, KNN ne construit pas de modèle explicite lors de la phase d'entraînement. L'ensemble des données d'apprentissage est conservé en mémoire et la classification est réalisée lors de la phase de prédiction. Pour un nouvel échantillon, l'algorithme recherche les k voisins les plus proches puis lui attribue la classe majoritaire parmi ces voisins.

La mesure de similarité la plus couramment utilisée est la distance euclidienne :

$$d(x_i, x_j) = \sqrt{\sum_{m=1}^{n} (x_{im} - x_{jm})^2} \tag{1.35}$$

où :

— x_i et x_j représentent deux vecteurs de caractéristiques ;

— n désigne le nombre total de caractéristiques.

Le paramètre k joue un rôle important dans les performances du classificateur. Une valeur faible de k peut rendre le modèle sensible au bruit présent dans les données, tandis qu'une valeur élevée peut diminuer sa capacité à distinguer correctement les différentes classes.

Grâce à sa simplicité de mise en œuvre et à son efficacité sur des ensembles de données de taille modérée, KNN demeure une méthode de référence pour l'évaluation des représentations de caractéristiques en vision par ordinateur et en reconnaissance d'objets.

IV.3 CNN comme classificateur

Les réseaux de neurones convolutifs (*Convolutional Neural Networks*, CNN) [20] peuvent être utilisés non seulement pour l'extraction de caractéristiques, mais également pour la classification directe des images.

Après les couches convolutives et les couches de sous-échantillonnage (*pooling*), les cartes de caractéristiques obtenues sont transformées en un vecteur puis transmises à une ou plusieurs couches entièrement connectées (*Fully Connected*). Ces couches ont pour rôle de combiner les informations extraites afin de prédire la classe de l'image.

La dernière couche du réseau utilise généralement la fonction d'activation *Softmax*, qui convertit les scores produits par le modèle en probabilités associées aux différentes classes :

$$P(y = i|x) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{1.36}$$

où :

— K représente le nombre total de classes ;

— z_i correspond au score attribué à la classe i.

L'entraînement du réseau est réalisé en minimisant la fonction de perte d'entropie croisée (*Cross-Entropy Loss*) :

$$L = -\sum_{i=1}^{K} y_i \log(\hat{y}_i) \tag{1.37}$$

où :
— y_i désigne l'étiquette réelle de la classe ;
— $\hat{y}_i$ représente la probabilité prédite par le réseau.

Grâce à leur capacité à apprendre automatiquement des représentations hiérarchiques à partir des données d'entrée, les CNN figurent parmi les méthodes les plus performantes pour les tâches de classification d'images. Leur efficacité a été démontrée dans de nombreux domaines tels que la reconnaissance d'objets, l'analyse d'images médicales et la vision par ordinateur.

V Conclusion

Ce chapitre a présenté les méthodes utilisées dans ce mémoire. Les espaces RGB et HSV décrivent les couleurs d'une image. Les histogrammes montrent la répartition des couleurs dans l'image. Plusieurs méthodes ont été utilisées pour extraire les caractéristiques des images. Parmi elles figurent Mini-Batch K-Means, AlexNet, U-Net, ViT puis CLIP. Des méthodes de classification ont aussi été présentées. Les modèles CNN, SVM puis KNN font partie de ces méthodes. Le chapitre suivant présente la méthodologie proposée ainsi que les expérimentations réalisées sur les bases COIL-100 et ALOI.

CHAPITRE 2

MÉTHODOLOGIE, EXPÉRIMENTATION ET RÉSULTATS

Dans ce chapitre, nous présentons la méthode de reconnaissance d'objets. L'objectif est de comparer plusieurs méthodes de traitement. Chaque méthode exploite les informations colorimétriques et les caractéristiques de forme présentes dans les images. Cette étude porte sur plusieurs modèles. On utilise également différentes techniques de classification. Les essais sont menés sur deux bases d'images. La première base porte le nom de COIL-100. La deuxième base s'appelle ALOI. Il y a de nombreux objets différents dans ces bases. Les images ont été prises dans différentes conditions. Cette étude vise à comparer les performances des différentes méthodes à l'aide de métriques quantitatives adaptées. C'est cette analyse qui montre le nombre d'itérations par méthode et son efficacité. Dans ce chapitre, nous étudions les performances de cinq méthodes d'extraction de caractéristiques (AlexNet, U-Net, Vision Transformer, CLIP et Mini-Batch K-Means) couplées avec trois classificateurs (CNN, SVM et KNN), dans le but de comparer leurs performances pour effectuer la tâche de classification. Sur les trois bases de données (COIL-100, ALOI_red2_ill et ALOI_red2_view), nous évaluons au total quarante-cinq configurations expérimentales (5 extracteurs $\times$ 3 classificateurs $\times$ 3 bases de données).

I Description des jeux de données

La qualité d'un système de classification dépend beaucoup des données utilisées pour son évaluation. Pour être représentative, nous avons sélectionné deux bases de données largement utilisées dans la littérature, qui ont été analysées avec les différentes méthodes étudiées.

Ces bases peuvent être utilisées pour comparer la robustesse d'un modèle en présence de différents changements : changement d'éclairage, changement d'angle de vue, changement d'apparence de différents objets.

I.1 Base COIL-100

La COIL-100 (Columbia Object Image Library) constitue l'une des bases de données de référence pour évaluer les techniques de reconnaissance d'objets 2.1.

L'université Columbia a construit cette collection comprenant une centaine d'objets différents, répartis en plusieurs catégories. Chaque objet est photographié sous soixante-douze vues différentes, obtenues par rotation autour d'un axe vertical.

L'angle entre deux images consécutives est de cinq degrés. Cette configuration permet de représenter entièrement les différentes vues d'un même objet.

La base comprend donc les éléments suivants : 100 objets ; 72 images pour chaque objet ; au total, 7200 images .

Les images possèdent un arrière-plan uniforme qui facilite l'évaluation des performances des techniques d'extraction de caractéristiques. Les variations d'orientation, bien que relativement simples, constituent un défi majeur pour les systèmes de classification. COIL-100 est particulièrement adapté à l'étude de la robustesse des représentations visuelles vis-à-vis des changements de perspective.

Figure 2.1 – Exemples tirés de la bibliothèque d'images d'objets de Columbia

I.2 Base ALOI

ALOI est une base de données plus complexe, constituée de plusieurs milliers d'images représentant un grand nombre d'objets réels. Les objets sont photographiés dans différentes conditions expérimentales, comme suit :

— Les variations d'éclairage.
— Les angles de points de vue.
— Les variations de couleur.
— Les variations de l'environnement d'acquisition.

Cette diversité fait d'ALOI une base particulièrement adaptée à l'évaluation des techniques de vision par ordinateur (figure 2.2).

Figure 2.2 – Exemples tirés de la bibliothèque d'images d'objets d'Amsterdam

Ici, deux sous-ensembles ont été utilisés dans cette dissertation.

I.2.1 ALOI_red2_ill

Ce sous-ensemble contient des images de différentes conditions d'illumination (figure 2.3).

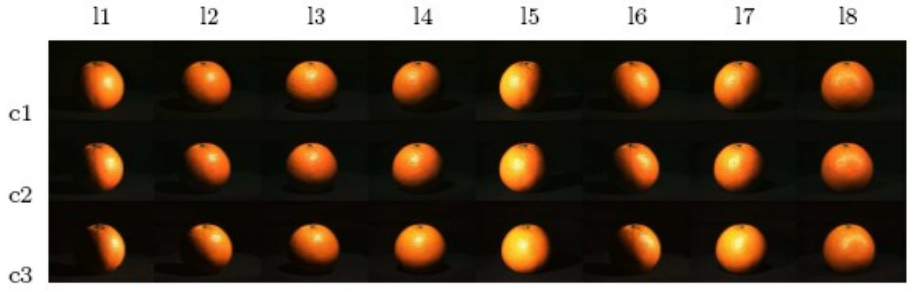

Figure 2.3 – Exemples d'objets d'ALOI observés sous 24 directions d'éclairage différentes

Les changements de lumière modifient de manière spectaculaire l'apparence des objets et ont le potentiel d'influencer la qualité des caractéristiques extraites.

L'usage de cette base assure ainsi la robustesse des modèles aux variations d'éclairage.

I.2.2 ALOI_red2_view

Ce sous-ensemble est alloué aux variations de perspective.

Chaque objet est vu à partir de plusieurs points de vue différents. Cette configuration permet d'évaluer la capacité des modèles à reconnaître un objet malgré les variations du point de vue de la caméra (figure 2.4).

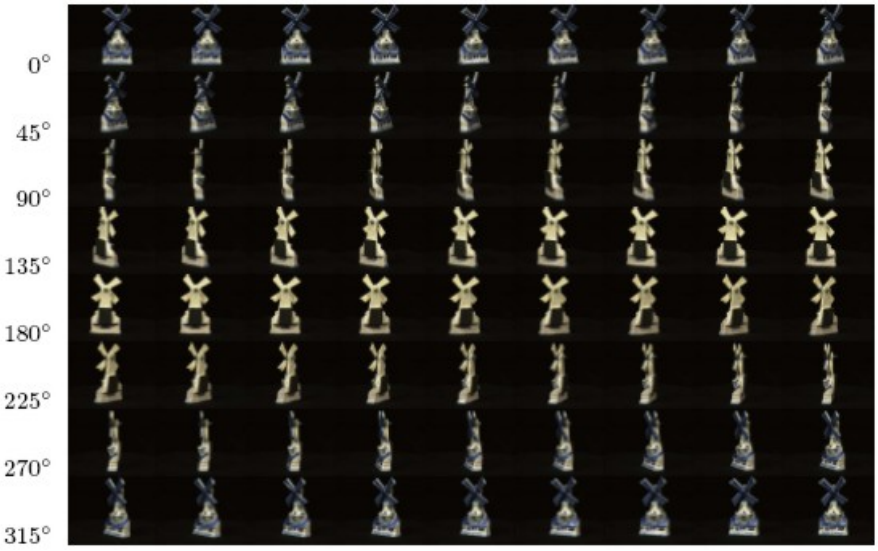

Figure 2.4 – Exemples d'objets d'ALOI vus sous différents angles

I.3 Comparaison des jeux de données

Les deux bases ont des propriétés qui se complètent. COIL-100 met principalement l'accent sur les variations géométriques tandis qu'ALOI introduit des conditions d'acquisition plus complexes. Les principales caractéristiques des bases de données utilisées dans cette étude sont résumées dans le Tableau 2.1.

Tableau 2.1 – Caractéristiques des bases de données utilisées

Dataset	Nombre d'objets	Nombre d'images	Variations principales
COIL-100	100	7200	Rotation
ALOI_red2_ill	1000	24000	Illumination
ALOI_red2_view	1000	72000	Angle de vue

Cette complémentarité permet d'avoir une image plus riche des méthodes étudiées. En combinant ces bases, on peut étudier les performances des modèles dans divers contextes qui sont représentatifs de nombreuses situations pratiques.

II Environnement expérimental

Pour garantir la reproductibilité des expériences que l'on a pu réaliser, il est important de décrire le matériel et le logiciel employé dans les tests différents menés.

Toutes les expérimentations ont été menées dans un environnement Python avec les principales bibliothèques de vision par ordinateur et d'apprentissage automatique.

II.1 Configuration matérielle

Les expériences ont été réalisées sur une station de travail possédant un processeur multicoeur et une mémoire suffisante pour traiter les jeux de données utilisés.

L'utilisation d'une carte graphique permet de gagner beaucoup de temps dans les étapes d'extraction de caractéristiques à partir de réseaux de neurones profonds.

La configuration matérielle utilisée est la suivante :

— Processeur : Intel Core i7.
— Mémoire vive : 16 Go.
— Système d'exploitation : Windows 11.

II.2 Configuration logicielle

Les principaux outils utilisés sont :

— Python.
— TensorFlow.
— PyTorch.
— OpenCV.
— Scikit-learn .
— NumPy.
— Matplotlib.

Ces bibliothèques fournissent l'ensemble des fonctions nécessaires au traitement des images, à l'extraction des caractéristiques ainsi qu'à l'évaluation des performances.

III Pré-traitements d'images

Avant d'extraire les caractéristiques, on applique plusieurs opérations de prétraitement pour uniformiser les données et réduire les variations indésirables.

Les étapes de prétraitement sont essentielles pour la qualité des représentations obtenues.

III.1 Redimensionnement

Les images présentes dans les différentes bases ont des dimensions variables.

Toutes les images sont redimensionnées à une taille fixe afin de garantir la compatibilité avec les architectures profondes utilisées. Cette opération réduit également le coût computationnel des traitements ultérieurs.Ce calcul à lui seul fait aussi baisser la complexité de calcul.

III.2 Normalisation

Les valeurs des pixels sont normalisées dans l'intervalle [0,1].

Cette normalisation améliore la stabilité numérique des algorithmes d'apprentissage et accélère leur convergence.

IV Méthode proposée

L'objectif de ce chapitre est de comparer diverses méthodes d'extraction de caractéristiques pour la classification d'images par la couleur. Plus précisément, nous visons à évaluer l'impact des caractéristiques extraites par les différentes approches sur les performances de classification.

Nous examinons cinq méthodes d'extraction de caractéristiques : AlexNet, U-Net, le Vision Transformer (ViT), CLIP et Mini-Batch K-Means. Pour chacun d'eux, les caractéristiques visuelles extraites servent à produire une représentation descriptive des images qui est ensuite utilisée lors de la phase de classification.

Les caractéristiques extraites sont ensuite représentées sous forme de vecteurs utilisés comme entrée des trois classificateurs étudiés : un réseau de neurones à convolution (CNN), une machine à vecteurs de support (SVM) et un classificateur des plus proches voisins (KNN). Cela permettra non seulement de tester le potentiel de chaque méthode d'extraction, mais aussi de rendre compte de l'indépendance du classifieur vis-à-vis de l'efficacité finale du système.

Les expériences ont été effectuées sur trois bases de données : COIL-100, ALOI_red2_ill et ALOI_red2_view. Sur chacune de ces bases, 15 configurations expérimentales ont été testées, correspondant aux combinaisons des cinq extracteurs de caractéristiques et des trois classificateurs, totalisant donc 45 expériences réalisées dans ce travail. Le présent travail propose donc une étude portant sur un total de quarante-cinq configurations expérimentales.

IV.1 Architecture générale du système

Le système développé suit une chaîne de traitement identique pour chacune des méthodes étudiées. Le pipeline général est présenté dans la Figure 2.5.

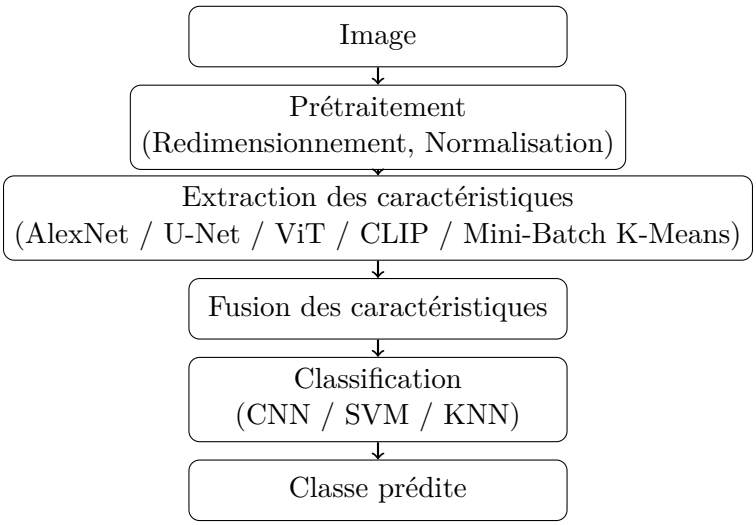

Figure 2.5 – Architecture générale du système proposé

Les images sont d'abord pré-traitées pour qu'elles soient uniformisées et adaptées aux différentes méthodes d'extraction. On extrait ensuite les caractéristiques visuelles

à l'aide de l'un des cinq modèles étudiés : AlexNet, U-Net, Vision Transformer (ViT), CLIP ou Mini-Batch K-Means.

Les caractéristiques extraites permettent de construire un vecteur descriptif représentant chaque image. Ce vecteur est ensuite fourni à l'un des trois classificateurs étudiés dans cette étude (CNN, SVM ou KNN). Enfin, on compare les performances obtenues pour chaque combinaison extracteur–classificateur afin d'identifier les configurations les plus performantes sur les différents jeux de données utilisés.

IV.2 AlexNet pour l'extraction de caractéristiques

AlexNet est l'un des premiers réseaux de neurones convolutifs profonds performants capables de faire de la reconnaissance d'images et donc de montrer sur de vraies données l'intérêt du deep learning.

Dans cette étude, AlexNet est utilisé comme extracteur de caractéristiques et non comme classificateur final. On soumet chaque image prétraitée au réseau pré-entraîné sur ImageNet. Dans le modèle, ce sont les couches convolutives qui extraient de manière hiérarchique les informations visuelles.

Les premières couches détectent principalement les contours, les couleurs et les textures simples. Les couches profondes vont capturer peu à peu des structures plus évoluées comme les formes ou encore les parties d'objets. Le vecteur de caractéristiques est extrait à partir des couches profondes du réseau, avant la dernière couche Softmax.

Ensuite ces caractéristiques sont utilisées pour représenter chaque image dans un espace de grande dimension dans lequel les objets similaires ont des représentations proches. Le principal avantage d'AlexNet est qu'il peut apprendre automatiquement des caractéristiques discriminantes, sans nécessiter d'intervention manuelle.

Pour AlexNet, nous avons utilisé les principaux paramètres suivants lors de nos expérimentations :

Tableau 2.2 – Paramètres utilisés pour AlexNet

Paramètre	**Valeur**
Taille d'entrée	224×224
Nombre de canaux	3
Modèle pré-entraîné	ImageNet
Fonction d'activation	ReLU
Dimension des caractéristiques	4096

Dans AlexNet (Tableau 2.2), la taille d'entrée de 224×224 pixels correspond au format d'image exigé par AlexNet. Les images de la base de données sont d'abord redimensionnées à cette taille avant traitement. On fixe le nombre de canaux à 3, représentant les composantes Rouge, Vert et Bleu (RGB) des images couleur. Il est alors pertinent d'utiliser un modèle pré-entraîné sur la base de données ImageNet. Cela permet de bénéficier des connaissances qu'a déjà acquises le modèle sur un très grand nombre d'images. Enfin, la dimension des caractéristiques est fixée à 4096, correspondant à la taille du vecteur de caractéristiques extrait de la couche entièrement connectée utilisée lors de la phase de classification.

IV.3 Extraction des caractéristiques avec U-Net

Contrairement à AlexNet, U-Net a été initialement développé pour les tâches de segmentation sémantique. Cependant, son architecture encodeur-décodeur permet également d'obtenir des représentations visuelles particulièrement riches.

Dans cette étude, seule la partie encodeur est exploitée pour extraire les caractéristiques. L'image traverse successivement plusieurs couches convolutives qui produisent des cartes de caractéristiques de plus en plus abstraites.

Les représentations obtenues au niveau du bottleneck contiennent une synthèse compacte des informations importantes de l'image. L'utilisation de U-Net présente plusieurs avantages :

— Premièrement, les connexions de saut permettent de préserver les informations spatiales fines.
— Deuxièmement, le modèle conserve davantage d'informations liées à la localisation des objets.
— Troisièmement, la segmentation implicite réalisée par le réseau favorise une meilleure séparation entre l'objet et son arrière-plan.

Les caractéristiques extraites sont ensuite transformées en vecteurs numériques utilisés lors de la phase de classification.

Tableau 2.3 – Paramètres utilisés pour U-Net

Paramètre	**Valeur**
Taille d'entrée	224×224
Convolutions	3×3
Pooling	2×2
Fonction d'activation	ReLU
Couche d'extraction	Bottleneck

Concernant U-Net (Tableau 2.3), les images d'entrée sont dimensionnées à une taille de (224×224) pixels, ceci afin d'assurer une homogénéité des données et de limiter le coût de calcul. Les couches de convolution appliquent des noyaux de taille 3×3, ce qui permet d'extraire de manière efficace les caractéristiques locales (les contours, les textures, les formes). On applique les opérations de pooling (2×2) qui permettent de diminuer petit à petit la dimension spatiale des cartes de caractéristiques (feature maps) tout en préservant les informations les plus significatives. On utilise la fonction d'activation ReLU après les convolutions pour introduire de la non-linéarité et rendre l'apprentissage du réseau plus facile. Finalement, l'extraction des caractéristiques a lieu en sortie de la couche *Bottleneck* située dans le cœur de l'architecture U-Net où les données visuelles sont représentées sous une forme compactée et fortement discriminante.

IV.4 Extraction de caractéristiques avec Vision Transformer

Le Vision Transformer est une évolution récente des techniques de vision par ordinateur. Cette architecture utilise uniquement les mécanismes d'attention, à la

différence des CNN classiques.

L'image est tout d'abord découpée en plusieurs patches de taille fixe. Chaque patch est ensuite transformé en un vecteur numérique, appelé patch embedding. La totalité des embeddings constitue une séquence qui ressemble à une phrase du traitement automatique du langage naturel. Cette séquence passe ensuite à travers plusieurs couches Transformer qui utilisent le mécanisme de self-attention.

Le mécanisme d'attention permet à chaque zone de l'image de communiquer directement avec toutes les autres zones. Cette propriété permet de voir de façon globale le contenu de l'image et permet de mieux modéliser les relations spatiales à longue distance. À l'issue du traitement, le token de classification donne un vecteur qui représente l'image entière. Ce vecteur sert de représentation visuelle pour les étapes de classification suivantes.

En raison de sa capacité à capturer les dépendances globales, Vision Transformer est particulièrement bien adapté aux tâches complexes de reconnaissance visuelle.

Tableau 2.4 – Paramètres utilisés pour Vision Transformer

Paramètre	**Valeur**
Taille d'entrée	224×224
Taille des patches	16×16
Nombre de couches	12
Nombre de têtes d'attention	12
Dimension des embeddings	768

Concernant le Vision Transformer (Tableau 2.4), les images sont redimensionnées à la taille (224×224) pixels avant d'être traitées. Chaque image est ensuite divisée en patches de (16×16) pixels, qui sont les unités d'entrée du modèle. L'architecture utilisée comprend 12 couches de transformers qui permettent d'apprendre des représentations hiérarchiques des données visuelles. Chaque couche inclut 12 têtes d'attention qui examinent en même temps différentes relations entre les patches de l'image, ce qui aide à saisir les dépendances globales. Chaque patch est finalement représenté par un vecteur d'embedding de dimension 768, ce qui donne une représentation riche et discriminante qui sert à l'extraction des caractéristiques et à la classification.

IV.5 Extraction de caractéristiques avec CLIP

CLIP est un modèle multimodal conçu pour apprendre des représentations visuelles et textuelles en même temps. Dans cette étude, CLIP sert d'extracteur de caractéristiques visuelles. L'image est introduite dans l'encodeur visuel du modèle. L'encodeur fournit un vecteur d'embedding qui représente le contenu sémantique de l'image.

Les représentations générées par CLIP sont affectées par l'apprentissage multimodal effectué lors du pré-entraînement, contrairement aux CNN classiques. Les caractéristiques ainsi obtenues possèdent une forte capacité de généralisation.

L'avantage majeur de CLIP est qu'il peut produire des représentations robustes à partir de données d'apprentissage relativement peu nombreuses. Les embeddings extraits servent ensuite de vecteurs d'entrée aux différents classificateurs.

Tableau 2.5 – Paramètres utilisés pour CLIP

Paramètre	**Valeur**
Taille d'entrée	224×224
Modèle visuel	ViT-B/32
Type d'embedding	L2 normalisé
Dimension des embeddings	512

Pour le modèle CLIP (Tableau 2.5), nous prenons le modèle visuel ViT-B/32, à savoir une architecture Vision Transformer qui segmente l'image en patches de (32×32) pixels afin d'obtenir des représentations visuelles riches. Les images sont redimensionnées à (224×224) pixels pour convenir au format d'entrée du réseau. Les caractéristiques extraites sont ensuite transformées en vecteurs d'embedding de dimension 512, ce qui permet une représentation compacte et discriminante des images. Enfin, les embeddings sont normalisés (au sens de la norme (L_2)), ce qui assure que tous les vecteurs sont de longueur unitaire et simplifie ainsi les comparaisons de similarité entre les différentes représentations visuelles.

IV.6 Extraction de caractéristiques avec Mini-Batch K-Means

Mini-Batch K-Means est une technique de clustering sans supervision qui permet de regrouper des données similaires entre elles. Dans le cadre de ce travail, cette approche sert à construire des descripteurs visuels compacts.

On regroupe en plusieurs clusters les pixels ou caractéristiques extraits des images. Un centroïde représente chaque cluster et décrit une région spécifique dans l'espace des caractéristiques. Une fois l'algorithme convergent, chaque image est décrite par un descripteur qui donne la fréquence d'apparition de chaque cluster. Cette représentation permet de décrire de façon statistique le contenu visuel de l'image. Les vecteurs ainsi obtenus sont ensuite donnés aux classificateurs de l'étude.

Tableau 2.6 – Paramètres utilisés pour Mini-Batch K-Means

Paramètre	**Valeur**
Nombre de clusters (k)	100
Taille du mini-batch	1000
Nombre maximal d'itérations	100
Initialisation	K-Means++

Pour l'algorithme Mini-Batch K-Means (Tableau 2.6), le nombre de clusters a été défini à (k = 100), ce qui correspond au nombre de classes présentes dans la base de données utilisée. Dans cette organisation, chaque cluster peut être associé à une catégorie potentielle d'objets. La taille du mini-batch est fixée à 1000 échantillons

afin de réduire le temps de calcul sans compromettre la qualité du regroupement. Le nombre maximal d'itérations est fixé à 100 afin que l'algorithme converge sans excéder de manière excessive le coût de calcul. Enfin, l'initialisation des centres de clusters se fait avec la méthode K-Means++, ce qui rend l'apprentissage plus stable et favorise une meilleure répartition initiale des centres. En général, cela améliore les performances du clustering.

Les caractéristiques extraites par les divers modèles sont ensuite regroupées en 100 clusters à l'aide de Mini-Batch K-Means. Les centres de ces clusters forment ainsi un dictionnaire visuel qui sert à représenter chaque image sous la forme d'un histogramme de fréquences, selon le modèle Bag of Visual Words.

IV.7 Paramètres du classificateur KNN

Le classificateur K-Nearest Neighbors (KNN) repose sur la recherche des voisins les plus proches dans l'espace des caractéristiques. La classe d'une image est déterminée à partir des classes des (k) échantillons les plus proches selon une mesure de distance.

Plusieurs valeurs du paramètre (k) ont été testées lors des expérimentations préliminaires. Les meilleurs résultats ont été obtenus avec la configuration présentée dans le Tableau 2.7.

Tableau 2.7 – Paramètres utilisés pour le classificateur KNN

Paramètre	**Valeur**
Nombre de voisins (k)	5
Mesure de distance	Euclidienne
Pondération	Uniforme
Algorithme de recherche	Auto

La distance euclidienne entre deux vecteurs de caractéristiques (x_i) et (x_j) est définie par :

$$d(x_i, x_j) = \sqrt{\sum_{m=1}^{n} (x_{im} - x_{jm})^2} \tag{2.1}$$

où (n) représente la dimension des vecteurs de caractéristiques.

Cette mesure est largement utilisée dans les problèmes de classification d'images en raison de sa simplicité, de son faible coût de calcul et de son efficacité pour évaluer la similarité entre les échantillons.

IV.8 Paramètres du classificateur SVM

Le classificateur Support Vector Machine (SVM) est utilisé afin d'évaluer la qualité des caractéristiques extraites par les différentes méthodes d'extraction.

Après plusieurs expérimentations préliminaires, le noyau RBF (*Radial Basis Function*) a été retenu en raison de sa capacité à traiter des données non linéairement séparables et à modéliser des frontières de décision complexes.

Les paramètres utilisés sont présentés dans le Tableau 2.8.

Tableau 2.8 – Paramètres utilisés pour le classificateur SVM

Paramètre	**Valeur**
Noyau	RBF
Paramètre (C)	1
Gamma	Scale
Critère d'arrêt (Tolérance)	10^{-3}

Le paramètre (C) contrôle le compromis entre la maximisation de la marge de séparation et la minimisation des erreurs de classification. Le paramètre *gamma* détermine l'influence des exemples d'apprentissage sur la frontière de décision.

Le noyau RBF permet de construire des frontières de décision non linéaires capables de séparer efficacement les différentes classes, ce qui le rend particulièrement adapté aux problèmes de classification d'images.

IV.9 Paramètres du classificateur CNN

Dans cette étude, un réseau de neurones convolutif (CNN) est également utilisé comme classificateur final afin d'évaluer les performances des différentes représentations extraites.

Les paramètres d'apprentissage retenus sont présentés dans le Tableau 2.9.

Tableau 2.9 – Paramètres utilisés pour le classificateur CNN

Paramètre	**Valeur**
Nombre d'époques	50
Taille du batch	32
Optimiseur	Adam
Taux d'apprentissage	0,001
Fonction de perte	Cross-Entropy
Fonction d'activation	ReLU
Fonction de sortie	Softmax

Par rapport à la fonction d'activation, nous avons utilisé ReLU pour introduire de la non-linéarité dans le réseau, puis nous avons choisi la fonction Softmax en sortie pour obtenir les probabilités des différentes classes. Enfin, l'optimiseur Adam a été choisi pour sa rapidité de convergence et sa capacité à ajuster automatiquement le taux d'apprentissage au cours de l'entraînement.

Cette méthode est souvent utilisée en classification d'images, car elle constitue un bon compromis entre la capacité du modèle à bien prédire et le temps nécessaire à l'apprentissage.

IV.10 Protocole expérimental

Pour chaque base de données, les images ont été divisées en deux ensembles :
— 70 % pour l'apprentissage ;
— 30 % pour le test.

La sélection des images a été réalisée de manière aléatoire. Chaque expérience a été réalisée 3 fois et les résultats rapportés correspondent à la moyenne obtenue.

V Critères d'évaluation

Évaluation : elle mesure les résultats que l'on obtient avec chaque méthode. Elle montre ce que le système est capable de reconnaître comme objet. Elle permet aussi de comparer les différentes méthodes mises en œuvre.

Dans ce texte, deux mesures sont employées. La première est le taux global de reconnaissance. La deuxième mesure est le temps d'exécution.

Le taux de reconnaissance global renseigne sur le nombre d'images correctement reconnues. Cette mesure compare le nombre de bonnes réponses au nombre total d'images testées.

On calcule l'Accuracy selon la formule suivante :

$$\text{Accuracy} = \frac{VP + VN}{VP + VN + FP + FN} \tag{2.2}$$

VP est le nombre de vrais positifs. VN est égal au nombre de vrais négatifs. FP est le nombre de faux positifs. FN=Nombre de Faux-Négatifs.

L'accuracy a une valeur comprise entre zéro et un. Une valeur s'approchant de un indique de bons résultats. Une valeur faible de l'accuracy indique des performances de classification limitées [21].

Dans ce mémoire, cette mesure est utilisée pour la comparaison des méthodes. Elle permet de repérer les meilleures performances.

Le temps d'exécution est la durée du traitement. Il évalue le temps nécessaire pour obtenir un résultat.

Cette mesure tient compte de l'extraction des caractéristiques. Elle intègre également la phase de classification.

Un temps plus petit correspond à une méthode plus rapide. Cette mesure permet de comparer la rapidité des méthodes.

L'Accuracy indique la qualité des résultats produits. Le temps d'exécution indique la rapidité du traitement. Ces deux mesures permettent d'évaluer complètement les méthodes étudiées.

Après avoir défini le protocole expérimental et les critères d'évaluation, les performances des différentes combinaisons extracteur–classificateur sont présentées et analysées sur les trois bases de données étudiées.

VI Résultats expérimentaux

Cette partie rassemble les résultats des différentes expériences réalisées sur les trois bases de données COIL-100, ALOI_red2_ill et ALOI_red2_view. Nous présentons les performances en apprentissage des cinq méthodes d'extraction de caractéristiques (resp. d'AlexNet, U- Net, CLIP, Mini-Batch K-Means et Vision Transfor- mer) couplées aux trois classificateurs (CNN, KNN et SVM).

Dans cette section, nous présentons les résultats des expérimentations. Les résultats sont exprimés sous forme de taux de reconnaissance (Accuracy) et de temps de calcul. Les données présentées dans cette section proviennent des expérimentations réalisées dans le cadre de ce mémoire.

VI.1 Résultats obtenus sur ALOI_red2_ill

Les résultats obtenus avec le classificateur CNN sur la base ALOI_red2_ill sont présentés dans le Tableau 2.10.

Tableau 2.10 – Résultats obtenus avec le classificateur CNN sur ALOI_red2_ill

Méthode	**Accuracy (%)**	**Temps (s)**
AlexNet	93.44	3018
U-Net	92.67	100.95
CLIP	75.31	3027
Mini-Batch K-Means	92.90	69
Vision Transformer	98.03	1606

L'analyse conclut que Vision Transformer affiche la meilleure performance. Il a un taux de reconnaissance à 98,03 %. AlexNet donne également de très bons résultats. U-Net donne aussi de très bons résultats. La méthode Mini-Batch K-Means obtient des performances proches de celles des autres approches d'extraction de caractéristiques. CLIP obtient les performances les plus faibles. Sa qualité de détermination ne surpasse que 75,31 %. Dans l'ensemble, le KNN classificateur reste le plus performant. Et la meilleure performance est acquise par le modèle U-Net à 99,08 %. Ces résultats confirment l'efficacité du Vision Transformer sur cette base de données. Malgré son ancienneté, AlexNet continue d'obtenir des performances élevées. Le K-Means en lots donne également de bons résultats. CLIP est la méthode qui donne les plus mauvais résultats.

Les résultats obtenus avec le classificateur KNN sur la base ALOI_red2_ill sont présentés dans le Tableau 2.11.

Tableau 2.11 – Résultats obtenus avec le classificateur KNN sur ALOI_red2_ill

Méthode	**Accuracy (%)**	**Temps (s)**
AlexNet	97.61	76
U-Net	99.08	46
CLIP	37.44	2721
Mini-Batch K-Means	97.24	73
Vision Transformer	97.25	66

Le classificateur KNN permet d'obtenir les meilleures performances globales sur cette base. U-Net atteint le meilleur résultat avec un taux de reconnaissance de 99.08 %. Vision Transformer, AlexNet et Mini-Batch K-Means produisent également des performances très élevées. CLIP présente encore une fois les résultats les plus faibles.

Le Tableau 2.12 présente les performances obtenues avec le classificateur SVM sur la base ALOI_red2_ill.

Tableau 2.12 – Résultats obtenus avec le classificateur SVM sur ALOI_red2_ill

Méthode	**Accuracy (%)**	**Temps (s)**
AlexNet	97.94	213
U-Net	95.06	134
CLIP	87.92	2903
Mini-Batch K-Means	51.19	77
Vision Transformer	98.58	353

L'association Vision Transformer–SVM fournit le meilleur résultat de l'ensemble des expérimentations sur cette base avec un taux de reconnaissance de 98.58 %. AlexNet obtient également d'excellentes performances. En revanche, Mini-Batch K-Means associé à SVM montre une forte dégradation avec seulement 51.19 %.

VI.1.1 Analyse globale de ALOI_red2_ill

Les résultats montrent que les architectures profondes actuelles sont remarquablement robustes face aux variations de lumière. En somme, le Vision Transformer diffère des autres méthodes, et l'U-Net couplé au KNN donne également d'excellents résultats.

CLIP a beau être entraîné de façon multilingue, il semble moins adapté à cette tâche précise. L'efficacité du Mini-Batch K-Means dépend fortement du classificateur utilisé.

VI.2 Résultats obtenus sur ALOI_red2_view

Cette seconde série d'expériences évalue la capacité des modèles à reconnaître les objets malgré les changements d'angle de vue. Les performances du classificateur CNN sur la base ALOI_red2_view sont résumées dans le Tableau 2.13.

Tableau 2.13 – Résultats obtenus avec le classificateur CNN sur ALOI_red2_view

Méthode	Accuracy (%)	Temps (s)
AlexNet	96.59	11197.00
U-Net	94.68	225.05
CLIP	78.54	8746.00
Mini-Batch K-Means	95.63	238.00
Vision Transformer	98.47	4348.00

Vision Transformer obtient une nouvelle fois les meilleurs résultats. AlexNet et Mini-Batch K-Means affichent également des performances élevées supérieures à 95 %.

Les résultats obtenus avec le classificateur KNN sur la base ALOI_red2_view sont présentés dans le Tableau 2.14.

Tableau 2.14 – Résultats obtenus avec le classificateur KNN sur ALOI_red2_view

Méthode	Accuracy (%)	Temps (s)
AlexNet	95.44	529
U-Net	97.92	227
CLIP	34.15	7514
Mini-Batch K-Means	98.03	168
Vision Transformer	98.06	146

Vision Transformer et Mini-Batch K-Means produisent des performances quasiment identiques. U-Net reste très performant tandis que CLIP présente encore les résultats les plus faibles.

Les résultats obtenus avec le classificateur SVM sur la base ALOI_red2_view sont présentés dans le Tableau 2.15.

Tableau 2.15 – Résultats obtenus avec le classificateur SVM sur ALOI_red2_view

Méthode	Accuracy (%)	Temps (s)
AlexNet	82.04	1989
U-Net	92.93	53185
CLIP	89.07	8640
Mini-Batch K-Means	44.49	153
Vision Transformer	95.91	691

Vision Transformer conserve sa position dominante avec un taux de reconnaissance de 95.91 %. U-Net fournit également de très bons résultats mais avec un temps d'exécution particulièrement élevé.

VI.2.1 Analyse globale de ALOI_red2_view

Certaines méthodes, notamment AlexNet avec SVM, sont plus sensibles aux variations d'angle de vue. Toutefois, Vision Transformer garde une très bonne stabilité

et réalise les meilleurs scores globaux. Les mécanismes d'attention permettent sans doute une meilleure modélisation des dépendances spatiales sur les différentes zones de l'image.

VI.3 Résultats obtenus sur COIL-100

COIL-100 constitue la base la plus favorable aux méthodes de classification étudiées. Les performances du classificateur CNN sur la base COIL-100 sont résumées dans le Tableau 2.16.

Tableau 2.16 – Résultats obtenus avec le classificateur CNN sur COIL-100

Méthode	**Accuracy (%)**	**Temps (s)**
AlexNet	95.83	1025
U-Net	98.61	22
CLIP	91.62	654
Mini-Batch K-Means	94.72	15
Vision Transformer	99.91	675

Vision Transformer atteint ici un résultat exceptionnel de 99.91 %, représentant la meilleure performance obtenue dans l'ensemble des expérimentations.

Le Tableau 2.17 présente les performances obtenues avec le classificateur KNN sur la base COIL-100.

Tableau 2.17 – Résultats obtenus avec le classificateur KNN sur COIL-100

Méthode	**Accuracy (%)**	**Temps (s)**
AlexNet	99.26	5
U-Net	99.72	4
CLIP	67.50	661
Mini-Batch K-Means	99.44	6
Vision Transformer	99.49	8

Le meilleur résultat est obtenu par U-Net associé à KNN avec un taux de reconnaissance de 99.72 %. AlexNet, Vision Transformer et Mini-Batch K-Means obtiennent également des performances remarquables.

Le Tableau 2.18 présente les performances obtenues avec le classificateur SVM sur la base COIL-100.

Tableau 2.18 – Résultats obtenus avec le classificateur SVM sur COIL-100

Méthode	**Accuracy (%)**	**Temps (s)**
AlexNet	92.69	12
U-Net	99.44	15
CLIP	94.44	618
Mini-Batch K-Means	61.48	6
Vision Transformer	99.31	37

U-Net et Vision Transformer obtiennent les meilleures performances. Mini-Batch K-Means reste moins performant lorsqu'il est associé au classificateur SVM.

VI.3.1 Analyse globale de COIL-100

Les résultats expérimentaux dans leur ensemble montrent que Vision Transformer est la méthode qui donne les meilleurs résultats dans lamajorité des configurations testées. Il reconnaît jusqu'à 99,91% des images de la base COIL-100. Le U-Net donne également d'excellents résultats, en particulier lorsqu'il est combiné au classificateur KNN.

Malgré son ancienneté, AlexNet conserve des résultats très compétitifs. Les performances de Mini-Batch K-Means varient en fonction du classificateur utilisé, alors que CLIP semble globalement moins adapté à la classification fine des objets dans les bases étudiées.

Ces résultats illustrent à la fois l'intérêt des architectures modernes profondes pour le problème de classification d'images par couleur et valident l'efficacité de la stratégie de fusion de caractéristiques adoptée dans cet article.

VII Comparaison globale

Une fois que les résultats obtenus ont été présentés sur les différentes bases de données, il convient de faire une analyse comparative globale, afin de déceler les méthodes les plus performantes, mais aussi d'apporter des éclairages sur l'influence des extracteurs de caractéristiques et des classificateurs utilisés.

Cette comparaison est faite tant au niveau de la classification des performances que de la durée d'exécution lors des expérimentations. Elle permet également la mesure de la robustesse des différentes approches aux variations d'éclairage et de point de vue dans les bases ALOIs, et aux variations d'orientation de la base COIL-100. Les résultats de cette partie sont obtenus à partir des expériences décrites précédemment.

VII.1 Comparaison des méthodes d'extraction de caractéristiques

Les cinq méthodes étudiées présentent des comportements différents selon la nature des données et le classificateur utilisé.

VII.1.1 Vision Transformer

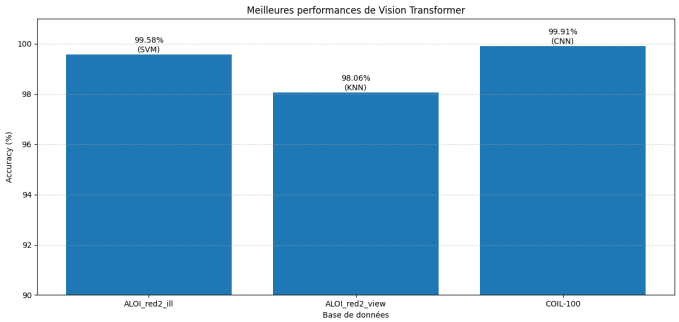

Figure 2.6 – Meilleures performances de Vision Transformer

Vision Transformer apparaît comme la méthode la plus performante dans la majorité des expériences réalisées.

Sur la base ALOI_red2_ill, le couple Vision Transformer–SVM atteint un taux de reconnaissance de 98.58 %. Sur ALOI_red2_view, Vision Transformer associé à KNN obtient 98.06 %. Enfin, sur COIL-100, le modèle atteint 99.91% lorsqu'il est utilisé avec CNN (figure 2.6).

Ces résultats montrent que les mécanismes d'attention permettent d'obtenir un apprentissage de représentations particulièrement discriminantes et robustes par rapport aux différentes variations présentes dans les images. L'analyse globale montre que, parmi les méthodes évaluées, Vision Transformer est le meilleur extracteur de caractéristiques.

VII.1.2 U-Net

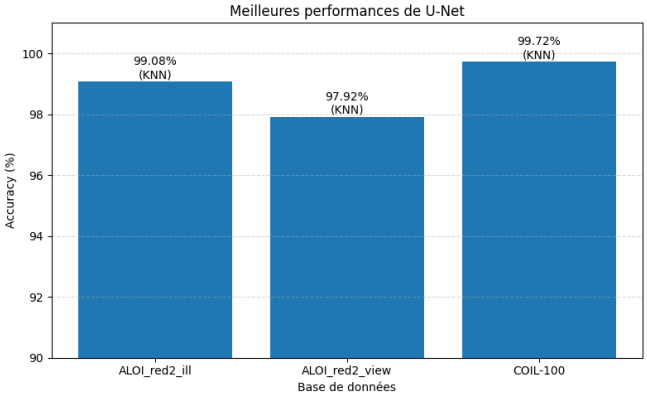

Figure 2.7 – Meilleures performances de U-Net

U-Net obtient également des performances très élevées sur l'ensemble des bases de données (figure 2.7).

Les meilleurs résultats sont observés avec le classificateur KNN où le modèle atteint :

— 99.08 % sur ALOI_red2_ill.
— 97.92 % sur ALOI_red2_view.
— 99.72 % sur COIL-100.

Les excellents résultats obtenus s'expliquent par la capacité de U-Net à préserver les informations spatiales grâce aux connexions de saut présentes dans son architecture. Cette méthode apparaît particulièrement adaptée lorsque les informations de couleur jouent un rôle important dans la discrimination des objets.

VII.1.3 AlexNet

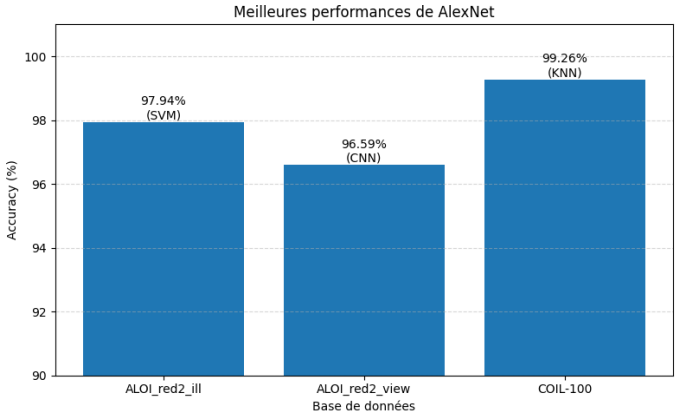

Figure 2.8 – Meilleures performances de AlexNet

L'algorithme de AlexNet fournit des performances globalement élevées malgré son ancienneté. Les taux de reconnaissance dépassent généralement 90 % sur l'ensemble des bases étudiées (figure 2.8). Le meilleur résultat obtenu par AlexNet atteint 99.26 % sur COIL-100 avec le classificateur KNN. Ces résultats démontrent que les architectures convolutives classiques demeurent compétitives pour les tâches de reconnaissance d'objets.

VII.1.4 Mini-Batch K-Means

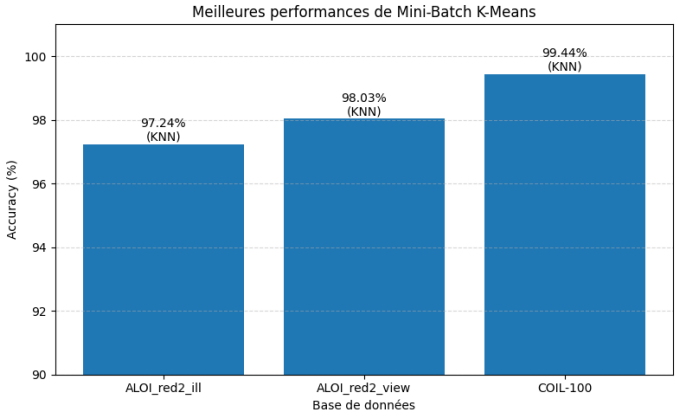

Figure 2.9 – Meilleures performances de Mini-Batch K-Means

Le comportement de Mini-Batch K-Means peut varier plus en fonction du classificateur utilisé. Quand on combine les caractéristiques générées avec KNN, on observe une performance particulièrement notable :

— 97.24 % sur ALOI_red2_ill.
— 98.03 % sur ALOI_red2_view.
— 99.44 % sur COIL-100.

À l'inverse, l'association avec SVM provoque une baisse importante des performances, en particulier sur ALO_red2_view où le taux de reconnaissance descend à 44,49 % (figure 2.9). Ces résultats suggèrent que la qualité des représentations fournies par Mini-Batch K-Means dépend beaucoup de la méthode de classification employée.

VII.1.5 CLIP

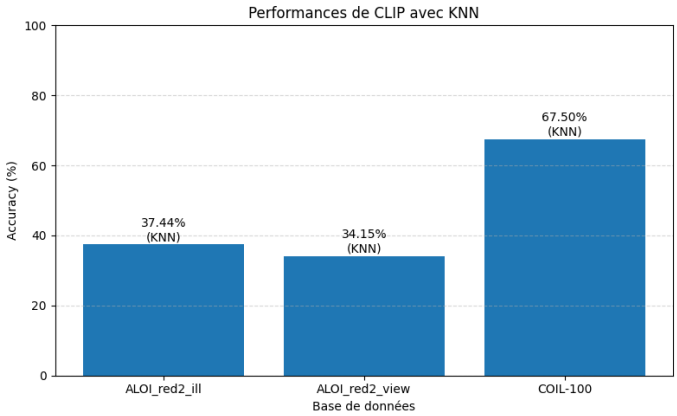

Figure 2.10 – Meilleures performances de CLIP

CLIP est la méthode qui obtient les résultats les plus faibles dans la majorité des expériences. Les performances restent acceptables avec CNN et SVM mais deviennent

nettement insuffisantes avec KNN (figure 2.10) :

— 37.44 % sur ALOI_red2_ill.
— 34.15 % sur ALOI_red2_view.
— 67.50 % sur COIL-100.

Cette situation peut s'expliquer par le fait que CLIP est principalement optimisé pour l'apprentissage multimodal généraliste et non pour la reconnaissance fine d'objets similaires.

VII.2 Conclusion de la comparaison

Les performances des techniques de caractérisation et de classification présentées dans les sections précédentes peuvent être évaluées à partir des résultats expérimentaux. Cette section comprend une analyse détaillée des résultats obtenus et une discussion sur l'impact de la couleur des caractéristiques, des extracteurs de caractéristiques, et classificateurs utilisés. L'objectif est de mettre en lumière les avantages et les limites des différentes approches étudiées, afin de mieux comprendre les facteurs qui influencent les performances du système proposé.

VII.2.1 Influence des méthodes d'extraction de caractéristiques

La Figure 2.11 présente une comparaison globale des performances obtenues par les différentes méthodes d'extraction.

Les résultats montrent que la qualité des caractéristiques extraites a un impact déterminant sur les performances finales du système de classification.

Les approches fondées sur l'apprentissage profond donnent en général des représentations plus discriminantes que les méthodes classiques. Cette supériorité provient de leur aptitude à apprendre de façon automatique les caractéristiques appropriées à la structure des données visuelles.

Parmi toutes les méthodes étudiées, Vision Transformer obtient les meilleurs résultats sur la majorité des expériences réalisées. Le modèle atteint notamment un taux de reconnaissance de 98.58 % sur ALOI_red2_ill avec SVM et de 99.91 % sur COIL-100 avec CNN.

Cette supériorité peut s'expliquer par le mécanisme d'attention qui permet au modèle d'analyser simultanément l'ensemble des régions de l'image. À la différence des réseaux convolutifs classiques, Vision Transformer réussit mieux à prendre en compte les relations globales entre les diverses parties de l'objet.

U-Net fournit également d'excellentes performances, notamment lorsqu'il est couplé au classificateur KNN. Les résultats obtenus montrent que les représentations extraites par son encodeur conservent bien les informations spatiales et colorimétriques nécessaires à la reconnaissance des objets. Cette propriété est probablement due à l'utilisation des connexions de saut qui préservent les détails fins de l'image.

Malgré son âge, AlexNet reste compétitif. Les résultats obtenus montrent que les réseaux convolutifs classiques sont toujours capables de générer des caractéristiques robustes pour les tâches de classification d'images. Leur performance est toutefois un peu moindre que celle des architectures plus récentes.

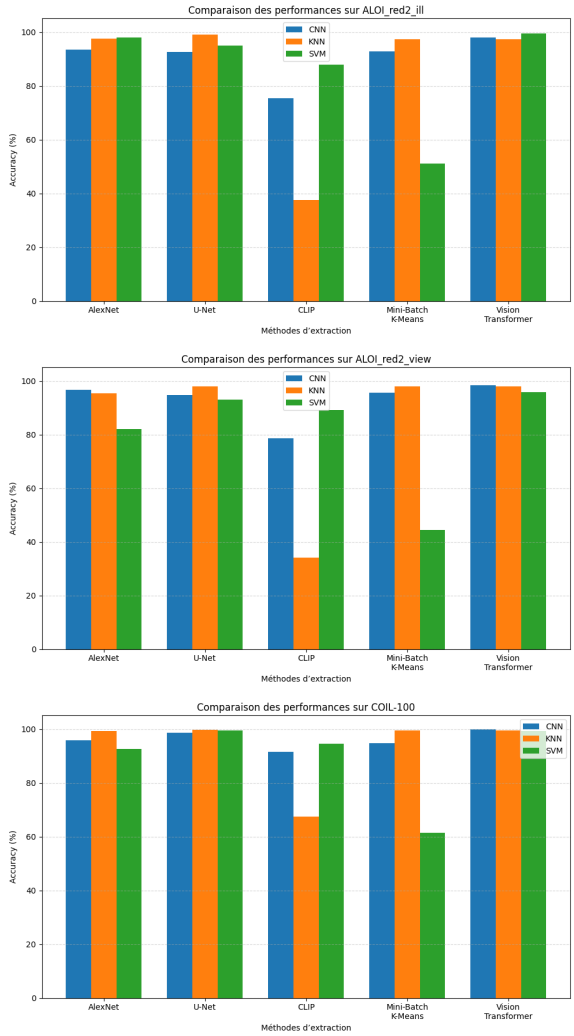

Figure 2.11 – Comparaison des performances des méthodes d'extraction de caractéristiques avec les classificateurs CNN, KNN et SVM sur les différentes bases de données.

VII.3 Comparaison des classificateurs

L'analyse des résultats montre également des différences importantes entre les trois classificateurs étudiés. Le classificateur CNN fournit généralement des performances élevées et stables. Il permet notamment d'obtenir le meilleur résultat absolu de l'étude avec Vision Transformer sur COIL-100 : 99.91 %. Cependant, les temps d'exécution sont souvent plus élevés que ceux observés avec KNN ou SVM. KNN apparaît comme le classificateur le plus efficace dans de nombreuses configurations. Il obtient fréquemment les meilleurs résultats avec U-Net, AlexNet et Mini-Batch K-Means. Sa simplicité de mise en œuvre ainsi que ses faibles temps d'exécution expliquent en grande partie ses bonnes performances. Les résultats montrent que les caractéristiques extraites sont suffisamment discriminantes pour permettre une classification efficace basée uniquement sur la proximité entre les vecteurs. Le clas-

sificateur SVM fournit également d'excellents résultats lorsque les caractéristiques sont de bonne qualité. Il obtient notamment le meilleur score sur ALOI_red2_ill avec Vision Transformer : 98.58 %. Toutefois, les performances de SVM diminuent fortement lorsque les représentations extraites sont moins adaptées, comme dans le cas de Mini-Batch K-Means.

VII.3.1 Comparaison des temps d'exécution

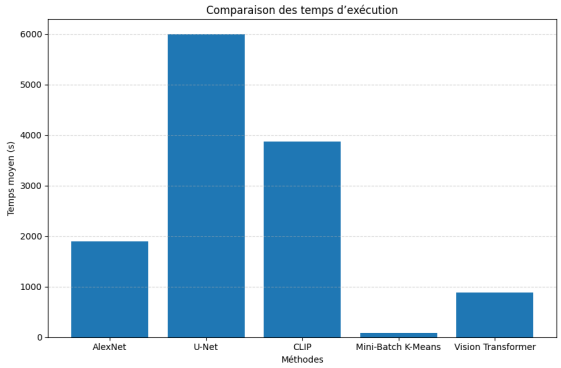

Figure 2.12 – Comparaison des temps d'exécution des différentes méthodes d'extraction de caractéristiques.

La Figure 2.12 montre une comparaison des temps moyens d'extraction des caractéristiques pour les différentes méthodes étudiées. Chaque valeur représente la moyenne des temps mesurés sur les bases ALOI_red2_ill, ALOI_red2_view et COIL-100 et ainsi, permet d'évaluer le coût de calcul global de chaque approche.Les temps d'exécution obtenus montrent d'importantes différences entre les méthodes. Les modèles profonds (ex. : AlexNet, CLIP, Vision Transformer) nécessitent en général davantage de puissance de calcul en raison de leur complexité. Par ailleurs, dans la plupart des cas, les plus faibles temps sont obtenus à l'aide de Mini-Batch K-Means et KNN. L'analyse montre que Vision Transformer offre le meilleur compromis entre précision et coût computationnel. Bien que son temps d'exécution soit supérieur à celui des méthodes classiques, les gains obtenus en termes de précision justifient largement ce surcoût.

VIII Discussion

VIII.1 Robustesse face aux variations d'illumination

Les expériences réalisées sur ALOI_red2_ill permettent d'évaluer la sensibilité des différentes méthodes aux changements d'éclairage. Les résultats montrent que Vision Transformer et U-Net conservent des performances élevées malgré les variations lumineuses. Cette robustesse est probablement liée à leur capacité à apprendre des représentations abstraites moins sensibles aux modifications locales de l'intensité

lumineuse. L'espace HSV joue également un rôle important dans cette robustesse puisque la séparation entre la teinte et la luminosité permet de limiter l'impact des variations d'éclairage sur les caractéristiques colorimétriques. Les performances plus faibles observées avec CLIP suggèrent que les représentations multimodales apprises lors du pré-entraînement sont moins adaptées à ce type de variations spécifiques.

VIII.2 Robustesse face aux variations d'angle de vue

Les résultats obtenus sur ALOI_red2_view mettent en évidence la capacité des modèles à reconnaître les objets malgré les changements de perspective. Vision Transformer conserve les meilleures performances dans ce contexte. Cette stabilité confirme l'intérêt du mécanisme d'attention globale qui permet au modèle d'exploiter simultanément plusieurs régions de l'image indépendamment de leur position. Les performances élevées de U-Net montrent également que les informations spatiales conservées par les connexions de saut contribuent à améliorer la reconnaissance des objets observés sous différents angles. Ces observations démontrent que les architectures profondes modernes sont capables de produire des représentations relativement invariantes aux transformations géométriques.

VIII.3 Analyse du coût computationnel

L'analyse des temps de calcul montre que les méthodes étudiées diffèrent notablement entre elles. Des modèles profonds comme Vision Transformer, AlexNet et CLIP nécessitent plus de ressources de calcul à cause de leur complexité architecturale et du nombre important de paramètres. Mini-Batch K-Means et KNN affichent généralement des temps d'exécution moins élevés. Cependant, le gain en précision de ces modèles profonds justifie souvent leur surcoût computationnel. Dans bon nombre d'applications, l'augmentation de la performance de reconnaissance est un progrès plus important que la suppression d'un temps de calcul. Le choix de la méthode dépend donc du compromis recherché entre les performances de classification et le coût computationnel.

VIII.4 Pourquoi Vision Transformer obtient-il les meilleurs résultats ?

On observe que Vision Transformer (ViT) réalise à la fois les meilleures performances globales sur les trois bases de données considérées. Cette supériorité peut être attribuée à sa capacité de modélisation des relations globales entre les différentes régions de l'image, via le mécanisme d'attention. ViT décompose une image en un ensemble de patches et les traite tous en même temps, contrairement aux réseaux convolutifs traditionnels, qui travaillent principalement sur des voisinages locaux. Grâce à cette propriété, il est capable de détecter les caractéristiques discriminantes liées à la forme et à la couleur des objets. De plus, le modèle utilisé dans cette étude bénéficie d'un pré-entraînement sur de très grandes bases de données, ce qui lui permet d'apprendre des représentations visuelles riches et généralisables. Cette

capacité se traduit par une meilleure robustesse face aux variations d'éclairage présentes dans ALOI_red2_ill ainsi qu'aux changements d'angle de vue observés dans ALOI_red2_view. Les résultats obtenus démontrent ainsi que les représentations extraites par ViT sont les plus discriminantes parmi les méthodes évaluées. Les résultats obtenus montrent que la capacité du mécanisme d'attention à modéliser les relations globales entre les différentes régions de l'image constitue un avantage déterminant pour la reconnaissance d'objets basée sur la couleur.

VIII.5 Pourquoi U-Net obtient-il de bonnes performances ?

Les résultats montrent que U-Net fournit des performances compétitives sur les différentes bases de données étudiées. Cela est dû à son encodeur profond qui lui permet d'apprendre des représentations riches et hiérarchiques. Contrairement aux méthodes classiques, U-Net extrait automatiquement des caractéristiques adaptées aux objets observés. Les connexions de saut permettent en outre de conserver des informations spatiales fines qui peuvent être utiles pour distinguer des objets présentant des couleurs ou des formes similaires. Toutefois, comme l'architecture a été conçue à l'origine pour la segmentation plutôt que pour la classification, ses performances restent généralement légèrement inférieures à celles du Vision Transformer, dont le mécanisme d'attention capture plus efficacement les dépendances globales entre les différentes régions de l'image.

VIII.6 Pourquoi CLIP est-il moins performant ?

CLIP est largement reconnu pour ses excellentes capacités de généralisation dans de nombreuses tâches de vision par ordinateur, mais dans le cadre de cette étude, ses performances sont inférieures à celles de Vision Transformer, U-Net et AlexNet. Cette différence s'explique par la nature même du modèle. CLIP s'appuie sur un apprentissage contrastif multimodal qui permet d'établir des correspondances entre les images et le langage naturel. Son but premier n'est donc pas d'extraire des caractéristiques optimisées pour une reconnaissance fine d'objets basée sur la couleur, mais d'apprendre des représentations sémantiques générales. Les caractéristiques obtenues par CLIP décrivent davantage le contenu global et le sens de l'image que les détails visuels discriminants nécessaires à la différenciation d'objets aux apparences proches. Par ailleurs, les bases COIL-100 et ALO sont constituées d'objets dont la distinction se fait principalement par leur apparence visuelle, leurs couleurs et leurs détails fins. Les caractéristiques extraites par CLIP paraissent moins appropriées dans ce contexte que celles produites par des architectures spécialement conçues pour l'analyse visuelle, telles que le Vision Transformer ou les réseaux de neurones convolutifs. Cette limitation se fait particulièrement ressentir lors de l'utilisation du classificateur KNN, où les taux de reconnaissance chutent fortement. Cela signifie que les représentations produites par CLIP ont une séparabilité moindre entre les classes dans l'espace des caractéristiques. Cette analyse est cohérente avec les résultats expérimentaux obtenus. Dans les trois bases étudiées, CLIP présente systématiquement des performances inférieures à celles de Vision Transformer et souvent à celles

d'AlexNet et d'U-Net. L'écart devient particulièrement important lorsqu'il est associé au classificateur KNN, ce qui confirme que les représentations extraites par CLIP sont moins discriminantes pour cette tâche spécifique de classification d'objets basée sur la couleur. Ces résultats ne remettent toutefois pas en cause l'efficacité générale de CLIP. Ils indiquent plutôt que les représentations multimodales apprises pour l'association image–texte ne sont pas nécessairement les plus adaptées à une tâche spécialisée de classification d'objets basée principalement sur la couleur.

VIII.7 Pourquoi KNN obtient-il parfois des performances plus faibles ?

Les résultats montrent que le classificateur KNN est généralement moins performant que SVM et CNN sur plusieurs configurations expérimentales. Cette différence s'explique principalement par le fonctionnement même de l'algorithme. KNN repose uniquement sur la proximité entre les vecteurs de caractéristiques. Lorsque la dimension de ces vecteurs devient importante, comme c'est le cas pour les représentations extraites par les modèles profonds, les distances entre échantillons deviennent moins discriminantes. Ce phénomène, connu sous le nom de « malédiction de la dimension », peut dégrader significativement les performances du classificateur. De plus, KNN est particulièrement sensible à la présence de bruit et aux variations intra-classe. Dans les bases comportant des changements d'éclairage ou de point de vue, les caractéristiques extraites peuvent présenter une dispersion importante, rendant plus difficile l'identification des voisins réellement pertinents. À l'inverse, SVM et CNN apprennent explicitement une frontière de décision optimisée, ce qui leur permet généralement de mieux séparer les différentes classes. Ce phénomène est particulièrement marqué lorsque les vecteurs de caractéristiques possèdent une dimension élevée. Dans ce cas, les distances entre échantillons deviennent moins discriminantes, ce qui réduit l'efficacité du principe de voisinage utilisé par KNN.

VIII.8 Influence réelle de la couleur sur la classification

Les résultats obtenus montrent que la couleur est une information discriminante importante pour la reconnaissance des objets. Les descripteurs colorimétriques permettent de compléter les caractéristiques de forme extraites par les différents modèles et améliorent ainsi la représentation globale des images. L'influence de la couleur se fait particulièrement sentir sur les bases contenant des objets aux teintes marquées. Dans ces conditions, les données provenant des domaines RGB et HSV permettent une meilleure distinction entre les classes et favorisent une amélioration des résultats de classification. Cependant, l'influence de la couleur reste dépendante des conditions d'acquisition. Les variations d'éclairage observées dans ALOI_red2_ill peuvent modifier significativement les valeurs colorimétriques et réduire leur pouvoir discriminant. Les méthodes capables d'extraire simultanément des informations de forme et de couleur, comme ViT ou U-Net, apparaissent alors plus robustes. Ces observations montrent que la couleur constitue un complément efficace aux caractéristiques visuelles profondes, mais qu'elle ne peut à elle seule garantir une reconnaissance fiable

dans toutes les situations.

IX Conclusion

La discussion des résultats expérimentaux montre que les architectures profondes modernes, en particulier Vision Transformer et U-Net, offrent les meilleures performances pour la classification d'images basée sur la couleur. Les informations colorimétriques constituent un complément pertinent aux caractéristiques profondes et contribuent à améliorer la représentation des objets. Les résultats obtenus confirment l'efficacité de l'approche proposée et démontrent l'intérêt de combiner plusieurs sources d'information visuelle afin d'améliorer les performances des systèmes de reconnaissance d'images. Malgré les résultats obtenus, cette étude présente certaines limites. Les expérimentations ont été réalisées sur un nombre limité de bases de données et uniquement avec trois classificateurs. D'autres architectures récentes et d'autres ensembles de données pourraient conduire à des conclusions complémentaires.

CONCLUSION GÉNÉRALE

L'objectif de ce mémoire était d'étudier et de comparer diverses méthodes d'extraction de caractéristiques pour la classification d'images fondée sur la couleur. La reconnaissance d'objets est un domaine important de la vision par ordinateur, où la qualité des caractéristiques extraites est essentielle pour les performances des systèmes de classification.

Une étude bibliographique a d'abord été menée dans un premier temps pour exposer les concepts majeurs relatifs à la classification d'images, aux descripteurs de couleur et aux méthodes modernes d'extraction de caractéristiques. On a étudié les architectures AlexNet, U-Net, Vision Transformer (ViT), CLIP, Mini-Batch K-Means ainsi que les classificateurs CNN, SVM et KNN.

Ensuite, une méthodologie expérimentale a été mise en place pour évaluer ces différentes approches sur les bases de données COIL-100, ALOI_red2_ill et ALOI_red2_view. On a combiné les caractéristiques extraites par chaque méthode avec trois classificateurs, afin d'analyser leurs performances dans différents contextes de variation d'illumination et d'angle de vue.

Les résultats expérimentaux montrent que les méthodes fondées sur l'apprentissage profond dépassent, dans l'ensemble, les approches plus classiques. Le Vision Transformer a affiché les meilleures performances sur la majorité des expériences menées, parmi toutes les méthodes testées. Cette supériorité tient principalement à sa capacité à exploiter les relations globales entre les différentes régions de l'image grâce au mécanisme d'attention. U-Net a également montré de très bons résultats en tant qu'extracteur de caractéristiques, prouvant que les représentations apprises par son encodeur peuvent être réutilisées efficacement pour la classification. AlexNet conserve des performances compétitives malgré son âge, tandis que Mini-Batch K-Means offre l'avantage d'un faible coût computationnel. À l'inverse, CLIP présente des performances plus limitées dans cette étude, probablement du fait de sa conception orientée vers l'apprentissage multimodal image-texte plutôt que vers la discrimination fine d'objets basée sur la couleur.

L'analyse des résultats montre aussi que le rôle des informations colorimétriques est particulièrement important pour les tâches de classification qui ont été abordées. Les expériences menées montrent que la couleur est un facteur discriminant important pour la reconnaissance des objets des bases de données utilisées. Les résultats acquis

aboutissent donc à la résolution de la première problématique définie au début de ce travail. L'emploi de méthodes modernes d'extraction de caractéristiques, notamment Vision Transformer, couplé à des informations colorimétriques pertinentes, permet d'améliorer sensiblement les performances de la classification d'images fondée sur la couleur.

Cette étude présente cependant certaines limites malgré les résultats encourageants obtenus. Les tests ont été effectués sur un faible nombre de bases de données et uniquement avec trois classificateurs. Par ailleurs, seules cinq techniques d'extraction de caractéristiques ont été prises en compte, alors que de nombreuses architectures récentes continuent d'apparaître dans le domaine de la vision par ordinateur. On peut envisager plusieurs perspectives dans la continuité de ce travail :

- Tester les méthodes proposées sur des bases de données plus complexes et plus diversifiées afin de valider leur capacité de généralisation.
- Explorer d'autres architectures récentes basées sur les Transformers, telles que Swin Transformer ou DeiT.
- Intégrer des mécanismes d'attention spécialement conçus pour exploiter les caractéristiques colorimétriques.
- Découvrir des techniques de fusion multimodale intégrant simultanément couleur, texture et forme.
- Étudier l'emploi de stratégies de fine-tuning pour adapter les modèles pré-entraînés aux caractéristiques propres des bases de données utilisées.
- Appliquer les méthodes étudiées à des problèmes réels comme la reconnaissance d'objets industriels, l'analyse d'images médicales ou les systèmes intelligents de surveillance.
- Étendre l'étude à d'autres méthodes de classification et à des approches d'apprentissage auto-supervisé récemment proposées dans la littérature.

En conclusion, ce travail montre l'intérêt des méthodes modernes d'extraction de caractéristiques pour la classification d'images et souligne le potentiel des architectures Transformer, aujourd'hui l'une des voies les plus prometteuses de la vision par ordinateur. donne moi directement le code corriger

BIBLIOGRAPHIE

[1] Minaee, S., Boykov, Y., Porikli, F., Plaza, A., Kehtarnavaz, N., & Terzopoulos, D. (2024). Image Recognition Using Deep Learning : A Survey. *Pattern Recognition*, 146, 109783. https://doi.org/10.1016/j.patcog.2023.109783

[2] Khan, S., Rahmani, H., Shah, S. A. A., & Bennamoun, M. (2023). Recent Advances in Color Image Representation and Recognition. *IEEE Access*, 11, 45678–45702.

[3] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems*, 25. https://doi.org/10.1145/3065386

[4] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., et al. (2021). An Image is Worth 16x16 Words : Transformers for Image Recognition at Scale. *International Conference on Learning Representations (ICLR)*. https://doi.org/10.48550/arXiv.2010.11929

[5] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., et al. (2021). Learning Transferable Visual Models from Natural Language Supervision. In *Proceedings of the International Conference on Machine Learning (ICML)*. https://doi.org/10.48550/arXiv.2103.00020

[6] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net : Convolutional Networks for Biomedical Image Segmentation. In *MICCAI 2015*, pp. 234–241. https://doi.org/10.1007/978-3-319-24574-4_28

[7] Sculley, D. (2010). Web-Scale K-Means Clustering. In *Proceedings of the 19th International Conference on World Wide Web*, pp. 1177–1178. https://doi.org/10.1145/1772690.1772862

[8] Bianconi, F., Fernández, A., Smeraldi, F., & Pascoletti, G. (2021). Colour and Texture Descriptors for Visual Recognition : A Historical Overview. *Journal of Imaging*, 7(11), 245. https://doi.org/10.3390/jimaging7110245

[9] Minaee, S., Boykov, Y., Porikli, F., Plaza, A., Kehtarnavaz, N., & Terzopoulos, D. (2024). Image Recognition Using Deep Learning : A Survey. *Pattern Recognition*, 146, 109783.

[10] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2), 91–110. https://doi.org/10.1023/B:VISI.0000029664.99615.94

[11] Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 886–893. https://doi.org/10.1109/CVPR.2005.177

[12] Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3), 273–297. https://doi.org/10.1007/BF00994018

[13] Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32. https://doi.org/10.1023/A:1010933404324

[14] Russell, S., & Norvig, P. (2010). Naive Bayes models. In *Artificial Intelligence : A Modern Approach* (3rd ed., pp. 499–508). Prentice Hall.

[15] Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1), 119–139. https://doi.org/10.1006/jcss.1997.1504

[16] Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*. https://arxiv.org/abs/1409.1556

[17] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1–9. https://doi.org/10.1109/CVPR.2015.7298594

[18] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778. https://doi.org/10.1109/CVPR.2016.90

[19] Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1), 21–27. https://doi.org/10.1109/TIT.1967.1053964

[20] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278–2324. https://doi.org/10.1109/5.726791

[21] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer, New York. https://doi.org/10.1007/978-0-387-45528-0

[22] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2021). *An Image is Worth 16x16 Words : Transformers for Image Recognition at Scale*. International Conference on Learning Representations (ICLR). https://arxiv.org/abs/2010.11929

ملخص

يُعَدّ تصنيف الصور من المجالات المهمة في الرؤية الحاسوبية والذكاء الاصطناعي، حيث يهدف إلى التعرف على محتوى الصور وتصنيفها بشكل آلي. يهدف هذا العمل إلى دراسة ومقارنة مجموعة من طرق استخراج الخصائص البصرية واللونية لتحسين أداء أنظمة تصنيف الصور. ولتحقيق ذلك، تم الاعتماد على خمس تقنيات لاستخراج الخصائص هي، AlexNet و U-Net و Vision Transformer (ViT) و CLIP و Mini-Batch K-Means ، مع الاستفادة من واصفات اللون المعتمدة على فضائي غجـ وحصط والمدرجات اللونية.

تم استخدام ثلاثة مصنفات مختلفة هي CNN و SVM و KNN لتقييم فعالية الخصائص المستخرجة. أُجريت التجارب على قاعدتي البيانات COIL-100 و ALOI اللتين تتضمنان صورًا لأجسام مختلفة في ظروف متنوعة من الإضاءة وزوايا الرؤية. وقد أظهرت النتائج تفوق نموذج Vision Transformer في معظم الاختبارات من حيث الدقة والقدرة على التعميم ، بينما حققت U-Net نتائج جيدة بفضل حفاظها على المعلومات المكانية. كما بينت الدراسة أن دمج الخصائص اللونية مع الخصائص المستخرجة بواسطة نماذج التعلم العميق يساهم في تحسين أداء التصنيف. وتؤكد النتائج أهمية اختيار أسلوب استخراج الخصائص المناسب للحصول على أفضل أداء في تطبيقات تصنيف الصور.

كلمات مفتاحية ـ استخراج الخصائص، الذكاء الاصطناعي، التعلم العميق، الرؤية الحاسوبية، تصنيف الصور، واصفات اللون.

Résumé

Ce mémoire présente une étude comparative de plusieurs méthodes d'extraction de caractéristiques pour la classification d'images. L'objectif est d'évaluer l'impact des descripteurs de couleur sur les performances de classification. Cinq approches ont été étudiées : AlexNet, Vision Transformer (ViT), U-Net, CLIP et Mini-Batch K-Means. Les caractéristiques extraites sont combinées à des informations colorimétriques issues des espaces RGB et HSV, puis classifiées à l'aide des algorithmes CNN, SVM et KNN. Les expérimentations sont réalisées sur les bases COIL-100 et ALOI afin d'analyser la robustesse des méthodes face aux variations d'éclairage et de point de vue. Les résultats montrent que Vision Transformer obtient les meilleures performances globales, tandis que U-Net offre également une excellente capacité de généralisation. L'intégration des caractéristiques de couleur améliore significativement la précision de classification.

Mots-clés : AlexNet, Classification d'images, CLIP, CNN, U-Net, Vision Transformer.

Abstract

This master thesis presents a comparative study of several feature extraction methods for image classification. The objective is to evaluate the impact of color descriptors on classification performance. Five approaches were studied : AlexNet, Vision Transformer (ViT), U-Net, CLIP, and Mini-Batch K-Means. The extracted features are combined with colorimetric information derived from the RGB and HSV color spaces, then classified using CNN, SVM, and KNN algorithms. The experiments are conducted on the COIL-100 and ALOI datasets in order to analyze the robustness of the methods against variations in lighting and viewpoint. The results show that Vision Transformer achieves the best overall performance, while U-Net also provides excellent generalization capability. The integration of color features significantly improves classification accuracy.

Keywords : AlexNet, Image Classification, CLIP, CNN, U-Net, Vision Transformer.