كلية العلوم و التقنيات فاس
ⵜⴰⵖⵡⵍⵎⵉⵜ ⵏ ⵜⵇⵇⴰⵏⵉⵢⵏ ⴷ ⵜⵓⵙⵙⵙⵉⵍⵜ
Faculté des Sciences et Techniques de Fès

جامعة سيدي محمد بن عبد الله
ⵜⴰⵙⴷⴰⵡⵉⵜ ⵙⵉⴷⵉ ⵎⵓⵃⴰⵎⴷ ⴱⵏ ⵄⴱⴷⵍⵍⴰⵀ
Université Sidi Mohamed Ben Abdellah

Département d'Informatique

Projet de fin d'études

MASTER SCIENCES ET TECHNIQUES
SYSTÈMES INTELLIGENTS & RÉSEAUX

RECONNAISSANCE DES CHIFFRES MANUSCRITS EN UTULISANT KNN ET PCA

LIEU DU STAGE : laboratoire Systèmes Intelligents et Applications

L.S.I.A de la Faculté des Sciences et Techniques

Réalisé par :

- OUTAYBI Salma

Encadré par :

- Pr. ABBAD Khalid
- Pr. ZENKOUAR Khalid

Soutenu le 22.05.2021 devant le jury composé de :

- Pr. MRABTI Fatiha	Faculté des Sciences et Techniques de Fès	(Présidente)
- Pr. DHASI Yasir	Faculté des Sciences et Techniques de Fès	(Examinateur)
- Pr. ZENKOUAR Khalid	Faculté des Sciences et Techniques de	(Encadrant)
- Pr. ABBAD Khalid	Fès Faculté des Sciences et Techniques de Fès	(Encadrant)

Année Universitaire 2020 - 2021

Dédicaces

Je dédie ce modeste travail comme preuve de reconnaissance :

À mes chers parents,

Aucune dédicace ne serait suffisante pour témoigner mon profond amour, mon immense gratitude et mon plus grand respect. C'est à travers vos encouragements et vos critiques qu'on s'est réalisée. J'espère avoir répondu aux espoirs que vous avez fondé en moi.

À Mon époux,

Merci pour votre amour et soutien, que dieu vous protège et vous procure joie et bonheur.

À ma chère sœur,

Ton aide, ta générosité, ton soutien ont été pour moi une source de courage et de confiance. Qu'il me soit permis aujourd'hui de t'assurer mon profond amour et ma grande reconnaissance.

À mes très chers amis,

Pour tous les moments du bonheur et de peine que j'ai partagé avec vous.

À mes professeurs et mes encadrants

Je vous dédie ce travail, car vous n'avez épargné aucun effort pour me soutenir et m'orienter, tout au long de mon cursus et mon stage. Vous avez contribué à la réussite de mon parcours par vos conseils et vos instructions précieuses.

Remerciements

Au terme de ce travail, je tiens à présenter mes sincères remerciements à ceux qui m'ont beaucoup appris au cours de ce stage, et à ceux qui ont eu la gentillesse de m'aider par leurs présences et leurs conseils.

Mes remerciements destinés au premier lieu, à DIEU le tout puissant de m'avoir donné la volonté, la santé et le courage pour réaliser ce projet.

J'adresse mes vifs remerciements à mes deux encadrants, M. KHALID ABBAD et M. KHALID ZENKOUAR pour leur aide et encadrement durant toute la période de travail sur ce projet de fin d'étude, pour leurs écoutes et leurs disponibilités. Leurs suivis et leurs précieuses consignes m'ont été d'une grande utilité afin d'aboutir aux résultats escomptés.

Je veux également rendre un hommage particulier au corps professoral de la faculté des sciences et techniques de Fès, et les intervenants professionnels responsables de la formation Master Sciences et Techniques « Systèmes Intelligents et Réseaux ».

Finalement, je remercie les membres du jury pour m'avoir honorée en acceptant d'évaluer et de juger ce travail.

Résumé

Depuis plusieurs décennies, la reconnaissance des chiffres manuscrits est une voie traditionnelle, qui a attiré l'attention des chercheurs et il reste un domaine de recherche très ouvert, dû à son grand nombre d'applications pratiques. En raison du progrès des technologies actuelles, telles que les dispositifs de capture de l'écriture manuscrite et les ordinateurs portables les plus puissants, le scénario actuel appelle à la nécessité d'une reconnaissance performante des chiffres manuscrits dans les banques, pour identifier les chiffres sur un chèque bancaire. De plus, il peut être utilisé dans les bureaux de poste pour identifier les numéros de boîte à code PIN. La création d'un système de reconnaissance à haute fiabilité définit la motivation principale de ce projet.

L'objectif de ce projet est de créer un modèle qui sera capable de reconnaître et de déterminer les chiffres manuscrits de la base de données MNIST en utilisant les concepts de de l'algorithme kNN (Méthode des k plus proches voisins). L'espace des caractéristiques de la base de données MNIST des images numériques manuscrites (0-9) est de 784 dimensions., elle a donc de nombreuses dimensions. Un outil de réduction de dimension est mis en œuvre pour réduire les éléments en un petit ensemble informatif de caractéristiques, avant d'utiliser les données dans notre modèle d'apprentissage automatique. Cela permet un apprentissage rapide de modèle et une capacité de bien visualiser les clusters.

Après la réduction de la dimensionnalité de l'ACP, le KNN final a atteint une précision de classification de plus de 97% dans un espace des caractéristiques de 100 dimensions.

Mots-clés : Reconnaissance des chiffres manuscrits, Réduction de dimensionnalité, Classification, Extraction des caractéristiques, KNN, ACP, MNIST.

Abstract

For several decades, handwritten digit recognition has been one of the most traditional methods of recognition, which attracts researchers' attention and remains a very open field of research with its various practical applications. Due to the advancement of new technologies, such as handwriting capture devices and laptops that are more powerful. The actual scenario calls for the importance of efficient handwritten digit recognition in banks to identify digits on a bank check. In addition, it can be used in post offices to identify PIN code box numbers. The creation of a high reliability recognition system defines the main purpose of this project.

Human perception of dimensions is usually limited to two or three degrees. Any further increase in the number of dimensions usually leads to the difficulty in visual imagination for any person. Hence, Machine Learning researchers often commonly have to overcome the curse of dimensionality in high dimensional feature sets with dimensionality reduction techniques.

The objective of this project is to create a model that will be able to recognize and determine the handwritten numbers of the MNIST database using the concepts of the kNN algorithm (k nearest neighbours). However, resolution is very high with a feature size of 784.In this proposed model, we are carried out on linear transformations of the feature sets using Principal Component Analysis (PCA) as dimensionality reduction technique .The lower dimension vectors obtained, are then used to classify the digits using K Nearsest Neighbors (KNN).

A conclusion: Using PCA as the dimensionality reduction technique resulted in the experimental model achieving the highest accuracy of 97% in a feature space of 100 dimensions on the MNIST dataset with the time efficiency.

Keywords: Handwritten Digit Recognition, Dimensionality Reduction, Classification, Feature Extraction, KNN, PCA, MNIST.

Tables des matières

Liste des figures

Liste des abréviations

CNN	Convolutional Neural Network
DR	Dimensionality Reduction
FE	Feature extraction
FS	Feature Selection
HDR	Handwritten Digits Recognition
HOG	Histogram of Oriented Gradients
IA	Intelligence Artificial
KNN	K-Nearest Neighbors
MNIST	Mixed National Institute of Standards and Technology
ML	Machine Learning
OCR	Optical Character Recognition
PCA	Principal Component Analysis
RF	Random Forest
SVM	Support Vector Machine
T-SNE	T-distributed Stochastic Neighbor Embedding
LDA	Linear Discriminant Analysis

Introduction générale

La reconnaissance des chiffres manuscrits (RCM) est un sous domaine de la reconnaissance optique des caractères(OCR), il a fait l'objet d'un nombre important de travaux de recherche, grâce à ses applications diverses et potentielles. Le recours à la RCM s'impose dans la plupart des domaines de la vie courante. A titre d'exemple, nous citons le tri postal qui est l'une des premières applications, où tous les jours des milliers d'enveloppes sont automatiquement triés. A l'instar de cette application, on distingue la lecture du montant numérique des chèques bancaires et l'identification du numéro de sécurité sociale.

Toutefois, malgré le progrès impressionnant des techniques utilisées, ainsi que l'explosion dans la puissance de calcul des ordinateurs, la recherche sur la RCM avance avec une performance de reconnaissance qui reste loin de celle de l'œil humain. Cela implique que le problème de reconnaître un chiffre manuscrit est un sujet de recherche important, ce qui définit notre motivation principale de ce projet.

Les principaux problèmes de la reconnaissance des chiffres manuscrits dépendent des styles d'écriture qui varient selon l'auteur. Il est habituellement très difficile, même pour l'homme, de reconnaître les chiffres manuscrits en raison de la différence importante entre les styles d'écriture, ainsi la similitude entre les chiffres écrits à la main, par exemple six et quatre qui peuvent ressembler au même chiffre selon le style d'écriture de l'auteur. La reconnaissance des chiffres manuscrits est la solution à ce problème qui utilise l'image d'un chiffre et reconnaît le chiffre présent dans l'image.

Les systèmes de reconnaissance de chiffres manuscrits sont basés sur les étapes suivantes de prétraitement, d'extraction de caractéristiques et de classification, éventuellement suivies d'une étape de post-traitement. A présent, nous nous intéressons en particulier aux méthodes d'extraction des caractéristiques et la classification. En effet, le fonctionnement du système de reconnaissance dépend de deux choses majeures, la façon dont les caractéristiques sont extraites d'une image numérique et le choix d'un meilleur classificateur. Les performances du classificateur peuvent dépendre de la qualité des caractéristiques du classificateur lui-même. [1]

La perception humaine des dimensions est généralement limitée à deux ou trois degrés. Toute nouvelle augmentation du nombre de dimensions entraîne généralement des difficultés d'imagination visuelle pour toute personne. Par conséquent, les chercheurs en apprentissage

automatique doivent souvent surmonter le fléau de la dimensionnalité, dans les ensembles de caractéristiques de haute dimension avec des techniques de réduction de la dimensionnalité.

L'objectif de ce projet est de créer un modèle qui sera capable de reconnaître et de déterminer les nombres manuscrits de la base de données MNIST, en utilisant les concepts de l'algorithme, cet est un algorithme est très simple à comprendre et à mettre en œuvre, en même temps, il offre une efficacité étonnamment élevée dans les applications pratiques. Sa simplicité, qui est certainement son avantage, et aussi une caractéristique qui le rend parfois inutilement négligé lors de la résolution de problèmes plus complexes. Fait intéressant, l'utilisation potentielle de l'algorithme est très large, nous pouvons l'utiliser pour fournir à la fois un apprentissage non supervisé et supervisé, et ce dernier dans la régression et la classification.

Dans ce projet, nous décidons de vérifier son efficacité dans la reconnaissance des chiffres manuscrit cependant, la résolution de l'image (28x28) pixels est très élevée avec une taille de caractéristique de 784.Dans ce modèle proposé, nous effectuons des transformations linéaires des ensembles de techniques de caractéristiques en utilisant l'analyse en composantes principales (ACP) comme réduction de dimensionnalité. Les vecteurs de dimension inférieure obtenus sont ensuite utilisés pour classer les chiffres numériques à l'aide des K voisins les plus proches (KNN). Cette méthode proposée montre une excellente performance avec une précision élevée.

La structure de ce mémoire s'articule autour de 4 chapitres :

- **Dans le chapitre 1** : nous présentons le cadre général du travail, le laboratoire d'accueil, la problématique de recherche et les objectifs à atteindre.
- **Dans le chapitre 2** : Afin d'affranchir le seuil de notre sujet, nous avons besoin de présenter les étapes constituant un système de reconnaissance des chiffres manuscrits, pour avoir une idée sur le contexte général de notre projet .Dans le cadre de ce travail, nous avons établi une étude bibliographique sur les systèmes des reconnaissance de chiffres manuscrits, ainsi nous allons présenter un état d'art sur les étapes suivies pour construire un système fiable, en mettant l'accent sur la tâche de la réduction de dimensionnalité ,dans la phase d'extraction de caractéristiques et la classification .
- **Dans le chapitre 3** : nous détaillerons notre approche proposée appliquée sur la base de données MNIST, qui combine les trois étapes, tels que le prétraitement d'images, l'extraction de caractéristiques avec L'ACP et enfin la classification en utilisant KNN.
- **Enfin le chapitre 4** : a été alloué pour présenter les résultats obtenus de notre système ainsi que sa validation.

Chapitre I: Cadre général du projet

Introduction

Afin de se situer dans le contexte du projet, nous allons présenter dans ce chapitre le laboratoire d'accueil, nous présentons ainsi les différents aspects de la reconnaissance de l'écriture manuscrite, ensuite nous définissons un système de reconnaissance des chiffres manuscrits ainsi ses domaines d'application, en mettant l'accent sur les différents problèmes de la reconnaissance des chiffres manuscrits.

I.1 Laboratoire d'accueil

Le projet a été proposé par le Pr. Zenkouar Khalid, Professeur d'Enseignement Supérieur. Et le Pr. Abbad Khalid, Professeur d'Enseignement Supérieur.

Il est effectué au Laboratoire Systèmes Intelligents & Applications : LSIA, qui est sous la direction du Pr. Arsalane ZARGHILI, Professeur d'Enseignement Supérieur.

Le laboratoire LSIA, crée en 2011, est une unité de Recherche du Centre d'Etudes Doctorales en Sciences et Techniques de l'Ingénieur domicilié à la Faculté des Sciences et Techniques de Fès et regroupant

Des laboratoires de recherche tous accrédités par l'Université Sidi Mohamed Ben Abdellah de Fès, et domiciliés à la Facultés des Sciences et Techniques, l'Ecole Supérieure de Technologie, la Faculté Poly disciplinaire de Taza, l'ENS de Fès et la Faculté de médecine et pharmacie de Fès.

Le LSIA est composé de 15 enseignants-chercheurs du département d'Informatique de la FST de Fès et de 34 doctorants. Cette imbrication étroite entre enseignement et recherche, est un élément essentiel de la dynamique du laboratoire.

Les thématiques de recherche se situent au cœur des Sciences et Technologies de l'Information et de la Communication et s'articulent essentiellement autour des thématiques

de recherche des enseignants chercheurs du laboratoire et assure une large couverture thématique présentant un atout très important pour le Laboratoire.

I.1.1 Equipes

Le laboratoire est composé de 3 équipes de recherche :

- Systèmes de Communication et Traitement de Connaissances (SCTC)
  - **Responsable : Pr. Jamal KHARROUBI**
  - **Thématiques de recherche :**
    - Traitement automatique de la parole
    - Traitement des langues naturelles
    - Intelligence Artificielle
    - Reconnaissance de formes
- Environnement Intelligents & Applications (VIA)
  - **Responsable : Pr. Ahlame BEGDOURI**
  - **Thématiques de recherche :**
    - Adaptation au contexte dans un environnement ambiant
    - M-learning / Social learning
    - Communautés de pratique
    - Réseaux adhoc : performances et sécurité
- **V**ision **A**rtificielle & **S**ystèmes **E**mbarqués (VASE)
  - **Responsable : Pr. Arsalane ZARGHILI**
  - **Thématiques de recherche :**
    - Traitement automatique de la langue Arabe
    - Traitement et Analyse d'images
    - Reconnaissance de formes
    - Intelligence Artificielle
    - Systèmes Embarqués et Théorie de codes.

I.1.2 Formation

Le laboratoire offre deux formations :

- Licence Sciences et Techniques « Génie Informatique »
- Master Sciences et Techniques « Systèmes Intelligents & Réseaux »

I.2 Aspects de la reconnaissance d'écriture manuscrite

La reconnaissance en ligne et hors-ligne sont deux modes différents de reconnaissance d'écriture manuscrite, ayant chacun ses outils propres d'acquisition et ses algorithmes correspondants de reconnaissance.

La première étape dans un système de reconnaissance, qui permet de reconnaître n'importe quel caractère dans n'importe quel format, consiste à convertir l'écriture en grandeurs numériques adaptées au système de traitement, avec un minimum de dégradations possibles.

Il existe plusieurs modes des systèmes de reconnaissance de l'écriture manuscrite selon le mode d'acquisition :

- La reconnaissance d'écriture en ligne est effectuée en temps réel, c'est-à-dire elle est effectuée pendant le traçage de caractère, ce qui permet d'obtenir une bonne correction et modification, selon la réponse donnée à la phase de reconnaissance chevauchée à la phase d'acquisition. [2] Les moyens de saisie en ligne couramment utilisés sont la tablette graphique avec un stylo électronique et l'écran tactile. [2]
- La reconnaissance d'écriture hors-ligne où l'information se présente sous forme d'un ensemble de pixels qui représente l'image d'un texte déjà existant, obtenue par un scanner ou une caméra.

Figure 1: La reconnaissance hors ligne et en ligne [2]

La reconnaissance de l'écriture hors-ligne est plus complexe que celle qui est en ligne, due à la présence du bruit dans le procédé d'acquisition des images et la perte d'information temporelle telle que l'ordre d'écriture et la vitesse. [2]

.

I.3 Système de reconnaissance des chiffres manuscrits

I.3.1 Définition

Le système de reconnaissance des chiffres est le fonctionnement d'une machine pour s'entraîner ou reconnaître les chiffres de différentes sources telles que les e-mails, les chèques bancaires, les papiers, les images, les plaques des véhicules, le traitement des montants des chèques bancaires, les entrées numériques dans les formulaires remplis à la main (par exemple, les formulaires fiscaux), etc.

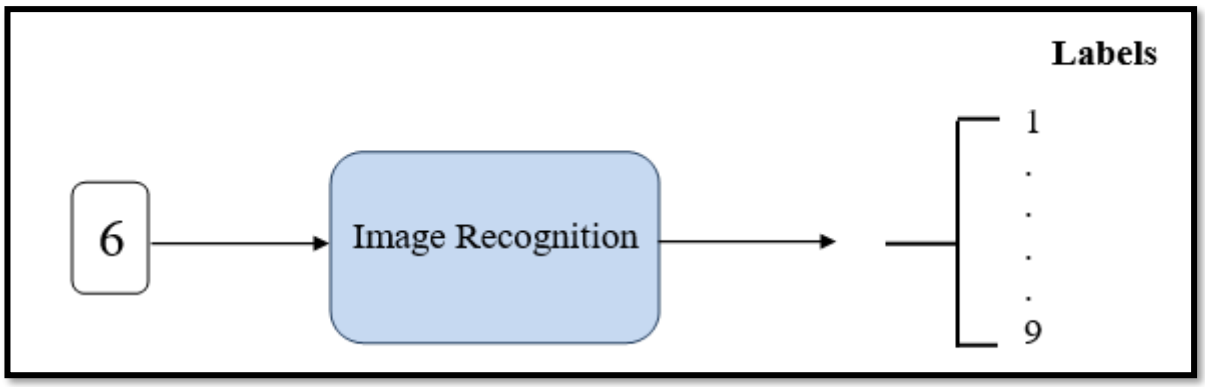


Figure 2 : Flux général du système de reconnaissance d'image.

I.3.2 Domaines d'application

En raison d'une variété d'applications potentielles telles que :

- La lecture de codes postaux.
- La lecture d'ordonnances médicales.
- L'interprétation d'adresses manuscrites.
- Le traitement des chèques bancaires.
- L'authentification de crédit.
- La protection sociale.
- L'analyse médico-légale des preuves de crime qui comprend une note manuscrite, etc.,

la reconnaissance numérique manuscrite est toujours un domaine de recherche actif [3].

I.3.3 Problématiques de projet

Les principaux problèmes de la reconnaissance des chiffres manuscrits à l'aide des approches d'apprentissage automatiques dépendent des chiffres manuscrits qui ne sont pas toujours de la même taille, largeur et orientation, car l'écriture diffère d'une de personne à une autre, de sorte que le problème général serait de classer les chiffres en raison de la similitude entre les chiffres tels que 1 et 7, 5 et 6, 3 et 8, 2 et 5, 2 et 7, etc.

Le problème se pose davantage lorsque de nombreuses personnes écrivent un seul chiffre avec des écritures différentes. Enfin, le caractère unique et la variété de l'écriture manuscrite de différents individus influencent également la formation et l'apparence des chiffres [4]

Les systèmes de reconnaissance des chiffres manuscrits sont basés sur les étapes suivantes de prétraitement, d'extraction de caractéristiques et de classification, éventuellement suivies d'une étape de post-traitement. En effet, Le fonctionnement du système de reconnaissance dépend de deux choses majeures :

- La façon dont les caractéristiques sont extraites d'une image numérique
- Le choix d'un meilleur classificateur. Les performances du classificateur peuvent dépendre de la qualité des caractéristiques du classificateur lui-même. [1]

Nous nous intéressons à améliorer la précision de la reconnaissance des chiffres manuscrits. Nous choisissons la fameuse base des données MNIST contenant 70 000 images d'entrainement et 10 000 tests à l'origine, qui sont en niveaux de gris 28 x 28 (0-255), étiquetés et au format bitmap. C'est une excellente base des données pour se familiariser avec l'apprentissage automatique tout en nécessitant un minimum d'efforts de prétraitement. Cependant Il existe de nombreuses caractéristiques dans MNIST, elles ont donc de nombreuses dimensions. Un outil de réduction de dimension, qui est appliqué pour réduire les éléments en un petit ensemble informatif de caractéristiques avant d'utiliser les données dans notre modèle d'apprentissage automatique.

Le fléau de dimensionnalité pose une problématique lors de l'application de L'algorithme des k plus proches voisins (kNN), ce dernier est un algorithme d'apprentissage automatique supervisé simple qui peut être utilisé pour résoudre des problèmes de classification et de régression. Il est facile à mettre en œuvre et à comprendre, mais il présente un inconvénient majeur: son ralentissement est important à mesure que la taille des données utilisées augmente.

I.3.4 Objectifs

Pour démontrer la réduction dimensionnelle, nous utiliserons le jeu de données MNIST bien connu qui contient des images de chiffres manuscrits. Nous verrons que toutes les caractéristiques ne sont pas nécessaires pour classer les chiffres.

Dans un premier temps, nous démontrerons l'utilisation de l'ACP et comment il peut être important pour l'évolutivité d'un algorithme prédictif comme k plus proche voisin (KNN), et que la réduction de la dimensionnalité des données réduit considérablement le temps de calcul et l'espace requis et donne également une meilleure précision aide à éviter le sur -ajustement.

L'objectif de ce projet est de créer un modèle qui sera capable de reconnaître et de déterminer les chiffres manuscrits de la base de données MNIST en utilisant les concepts de de l'algorithme kNN (Méthode des k plus proches voisins). La méthode de réduction de dimensionnalité PCA est mise en œuvre pour réduire les éléments de la base MNIST en un petit ensemble informatif de caractéristiques avant d'utiliser les données dans notre modèle d'apprentissage automatique. Cela permet un apprentissage rapide de modèle et une capacité de bien visualiser les clusters.

Conclusion

Dans ce chapitre, nous allons défini le contexte général de notre projet de fin d'études en présentant, tout d'abord ,le laboratoire d'accueil en précisant les équipes et les thématiques de recherches et leur responsables .Nous allons également introduit le contexte de projet en présentant une définition sur les systèmes de reconnaissances des chiffres manuscrits , Les différents domaines d'application de ces systèmes ,enfin nous détaillons la problématique de recherche et les objectifs à atteindre .

Dans le chapitre suivant, nous nous intéressons sur l'état de l'art des différentes étapes pour constituer un système fiable en mettent l'accent sur la réduction de dimensionnalité et les techniques de la classification.

Chapitre II: Etat d'art

Introduction

Dans le cadre de ce travail, nous nous intéressons sur deux parties cruciales dans l'établissement d'un système de reconnaissance, qui sont l'extraction des caractéristiques et la classification. Nous présentons ainsi un état d'art sur les propositions classiques pour s'attaquer au fléau de la dimensionnalité qui se pose dans les domaines des caractéristiques de grands dimension. Nous détaillons plus tard Les techniques de classification pertinentes, qui contribuent à la reconnaissance des chiffres manuscrits. Nous fournissons ensuite un aperçu approfondi et détaillé de la littérature récente correspondant à cette étude en détailleront les différents travaux en fonction de l'approche utilisé et le résultat obtenu.

II.1 Drosophile des chercheurs dans L'IA

La reconnaissance d'écriture manuscrite est l'un des plus vieux problèmes qui ait été posé à l'intelligence artificielle, depuis son avènement dans les années 1950. Terrain de jeu incontournable des nouveaux algorithmes d'apprentissage automatique, elle reste un véritable défi scientifique et technique.

L'analyse des chiffres manuscrite est un processus lourd et organisé ,qui repose sur une connaissance approfondie de la façon dont les gens forment les chiffres ou les lettres, et qui exploite les caractéristiques uniques des chiffres et des lettres, par exemple les formes, les tailles et les styles d'écriture individuels que les gens utilisent [3].

Les styles d'écriture personnels peuvent varier selon les outils d'écriture et l'environnement et ils peuvent laisser des indices sur l'identité de l'auteur. Dans le domaine de l'analyse médico-légale qui comprend l'enquête sur les scènes de crime, les tests ADN, l'analyse des fibres, l'analyse des empreintes digitales, l'étude de l'écriture manuscrite joue un rôle important. [3].

En règle générale, les experts en écriture utilisent des modèles de classification sophistiqués, pour analyser les images de caractères manuscrits. Dans le cadre de ce

processus, ils extraient des caractéristiques des échantillons, notamment les inclinaisons, l'orientation et l'alignement au centre des lettres.

II.2 Processus général d'un système RCM

Les systèmes de reconnaissance des chiffres manuscrits sont généralement basés sur les étapes principales suivantes :

- Acquisition.
- Prétraitement.
- Segmentation.
- Extraction des caractéristiques.
- Classification.

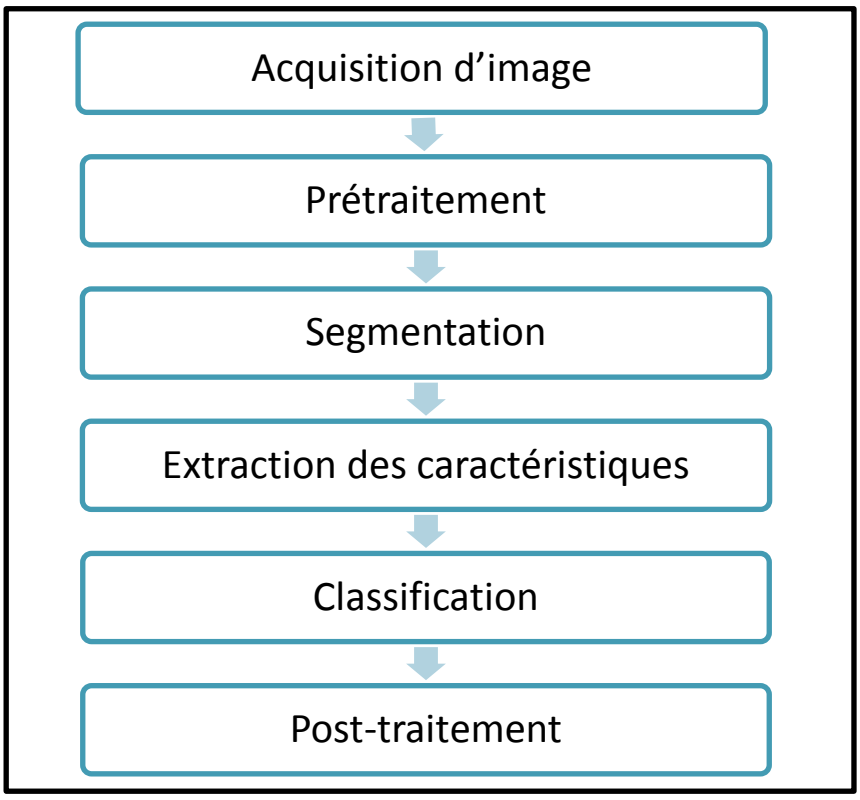


Figure 3: Schéma général d'un système de reconnaissance des chiffres manuscrits [4]

II.2.1 Phase d'acquisition

La phase d'acquisition se base essentiellement sur la capture de l'image de l'écriture manuscrite, au moyen des capteurs physiques et de la convertir en élévations numériques adaptées au système de traitement informatisé, avec un minimum de dégradation possible. Dans le cas où l'information est disponible sur un support souvent papier, les capteurs physiques sont des scanners ou des caméras numériques.

Durant cette phase, malgré la bonne qualité des systèmes d'acquisition, des bruits parasites peuvent apparaitre et ils peuvent causer une hétérogénéité du fond. Ceci est dû à la nature de la texture, l'aire de travail et de son éclairage. [5] [6]

II.2.2 Phase de prétraitement

Lorsque l'acquisition est effectuée, la plupart des systèmes comportent une étape de prétraitement. Généralement, ces prétraitements ne sont pas spécifiques à la reconnaissance de l'écriture, mais sont des prétraitements classiques en traitement d'image.

Le prétraitement a pour but de préparer l'image du tracé à la phase suivante d'analyse. Il s'agit essentiellement de réduire le bruit superposé aux données et ne garder, autant que possible, que l'information significative de la forme présentée.

Le bruit peut être dû au dispositif d'acquisition, aux conditions d'acquisition (éclairage, mise incorrecte du document…), ou encore à la qualité du document d'origine. Parmi les opérations de prétraitements généralement utilisées sont : la binarisation, le lissage, la squelettisation, la normalisation et la squelettisation. [5] [6]

Les options de prétraitement consistent à normaliser la taille et le rapport hauteur / largeur de l'image, les distorsions élastiques et les techniques d'interpolation pour les valeurs de pixel, etc. Le but du prétraitement est de supprimer le bruit, de lisser et de normaliser les données d'entrée, ce qui est essentiel pour une meilleure différenciation des motifs dans l'espace des caractéristiques.

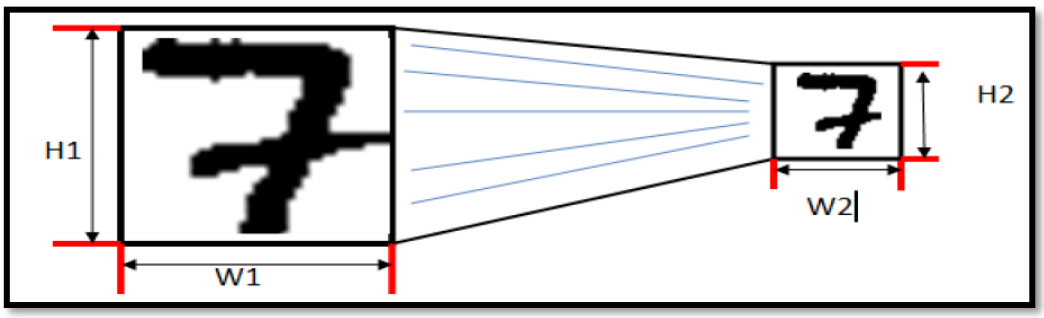


Figure 4: Exemple de la normalisation de chiffre 7

II.2.3 Phase de Segmentation

Cette phase sert à la séparation des graphèmes, des mots ou des chaînes numériques. Le problème de la segmentation des caractères manuscrits constitue le défi principal de la reconnaissance de l'écriture manuscrite.

L'une des applications particulières de la reconnaissance est la reconnaissance du montant chiffre sur les chèques bancaires du fait de la présence de bruits, le chevauchement de chiffres et même la fragmentation d'un chiffre en plusieurs parties difficilement identifiables. [5] [6].

II.2.4 Phase d'extraction de caractéristiques

Dans un système de reconnaissance des chiffres manuscrits. Le but d'extraction de caractéristiques est d'obtenir le volume d'informations le plus pertinent qui sera fourni au système. C'est une étape critique lors de la construction d'un système de reconnaissance. Le défi qui se compose dans cette phase et ses techniques d'extraction peuvent accompagner une perte d'information.

La réduction du nombre de caractéristiques peut aider à :

- Visualiser des donnés.
- Réduire les temps d'apprentissage de classification des systèmes.
- Améliorer les performances en classification.
- Réduire la taille des bases d'apprentissage. [5] [6].

Les types de caractéristiques peuvent être classés en quatre catégories principales de

Primitives: structurelles, statistiques, globales (transformations globales) :

- **Les primitives structurelles** (ou primitives locales) basées sur une représentation linéaire du caractère (décomposition du caractère en segments de droites et

courbes, contours du caractère, squelette). Les primitives structurelles sont généralement extraites non pas de l'image brute, mais via une représentation de la forme par le squelette ou par le contour [13]. Ainsi, parmi ces caractéristiques, il s'agit principalement des segments de droite, des arcs, boucles et concavités, des pentes, la hauteur et la largeur du caractère... etc. [5]

Les primitives statistiques portantes des informations concernant la distribution des pixels dans l'image du caractère ou chiffres. Les primitives statistiques décrivent une forme en termes d'un ensemble de mesures extraites à partir de cette forme. [5]

- **Les primitives globales** basées sur une transformation globale de l'image. La primitive globale dépend de la totalité des pixels d'une image, ces primitives sont donc dérivées de la distribution des pixels. Ils dirigent trois familles de caractéristiques telles que : les moments invariants, les projections et les profils. [5]
- **Primitives topologiques ou métriques Dans** ils se basent essentiellement sur des densités de pixels. Il s'agit d'effectuer une mesure sur l'échantillon au moyen d'une métrique. Parmi les mesures opérées, nous pouvons noter :
    - Compter le nombre de trous.
    - Evaluer les concavités.
    - Mesurer des pentes, des courbures et évaluer des orientations principales.
    - Mesurer la longueur, surfaces, les périmètres et l'épaisseur des traits. [5]

II.2.5 Phase de Classification

Après la segmentation de caractères et l'extraction des caractéristiques, une étape de reconnaissance basée sur la classification de caractères est employée. Le type d'une méthode de classification se décline généralement en deux familles :

- **Classification supervisée** :cette technique est basée sur l'étiquetage des observations, en affectant chaque observation à une classe (où la sortie correcte doit être fournie à l'avance). [7]
- **Classification non supervisée** : Aucune des observations n'est étiquetée (où la sortie correcte n'est pas exigée à l'avance, elle résulte après une étape d'apprentissage). [7]

II.2.6 Phase de post-traitement

Le post-traitement permet de corriger l'image, qui a été mal classée plus tôt et de modifier la détection. Cette méthode donne l'amélioration des résultats et fournit la précision de haut niveau.

II.3 Réduction de dimensionnalité

La réduction de la dimensionnalité est un processus étudié en mathématiques et en informatique qui consiste, à prendre des données dans un espace de grande dimension et à les remplacer par des données dans un espace de plus petite dimension. Pour que l'opération soit utile, il faut que les données en sortie représentent bien les données d'entrée.

La réduction de la dimensionnalité est utilisée comme étape de prétraitement, ce qui peut éliminer les données non pertinentes, le bruit et les caractéristiques redondantes. La réduction de dimensionnalité (DR) a été effectuée sur la base de deux méthodes principales, qui sont la sélection de caractéristiques (FS) et l'extraction de caractéristiques (FE). [8]

II.3.1 Les aspects de réduction de dimensionnalité

Il existe plusieurs approches pour faire cette opération et plusieurs objectifs possibles à atteindre. Les méthodes classiques sont la sélection de caractéristiques, qui consiste à sélectionner un ensemble de variables qui vont être conservées et l'extraction de caractéristiques, qui consiste à créer de nouvelles variables plus pertinentes.

.II.3.1.1 La sélection de caractéristiques

La sélection de caractéristiques est généralement définie comme un processus de recherche permettant de trouver un sous-ensemble "pertinent" de caractéristiques parmi celles de l'ensemble de départ. La notion de pertinence d'un sous- ensemble de caractéristiques dépend toujours des objectifs et des critères du système.

Soit $F = \{f_{1,}, f_{2,}, \dots, f_{N,}\}$un ensemble de caractéristiques de taille N ou N représente le nombre total de caractéristiques étudiées. Soit E_v une fonction qui permet d'évaluer un sous-ensemble de caractéristiques. Nous supposons que la plus grande valeur deE_v soit obtenue pour le meilleur sous-ensemble de caractéristiques. L'objectif de la sélection est de trouver un sous-ensemble $F'(F' \subseteq F)$de taille $N'(N' \leq N)$tel que :

$$E_v(F') = \max_{Z \subset F} E_v(Z)$$

Ou $|Z| = N'$ et N'Ou est, soit un nombre prédéfini par l'utilisateur ou soit contrôlé par une des méthodes de génération de sous-ensembles

La phase de l'algorithme de sélection des caractéristiques est divisée en deux phases telles que :

1) Génération de sous-ensemble.
2) Évaluation de sous-ensemble.

Dans la génération de sous-ensemble, nous devons générer un sous-ensemble à partir du jeu de données d'entrée et pour utiliser les évaluations de sous-ensemble, nous devons vérifier si le sous-ensemble généré est optimal ou non.

La Figure ci-dessus montre la méthode globale du processus de sélection des caractéristiques.

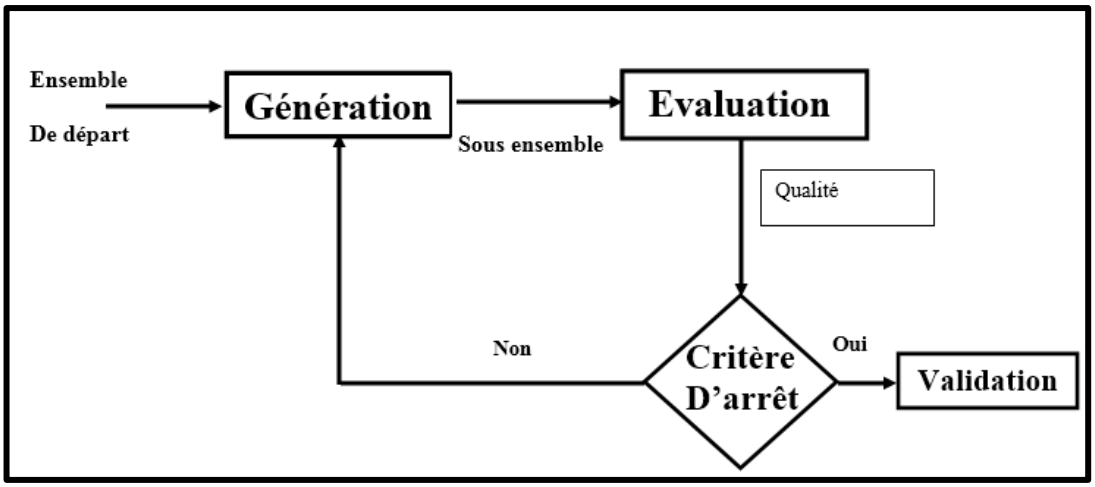


Figure 5: Processus de sélection de caractéristique [8]

Il existe trois types de stratégies de sélection de caractéristiques :

- Dans la première stratégie, la taille du sous-ensemble à sélectionner (N0 par exemple) est prédéfini et l'algorithme de sélection cherche é trouver le meilleur sous-ensemble de cette taille.
- Dans la deuxième stratégie consiste à sélectionner le plus petit sous-ensemble dont la performance est plus grande ou égale à un seuil prédéfini.
- Enfin la troisième stratégie cherche à trouver un compromis entre l'amélioration de la performance (L'erreur de classification par exemple) et la réduction de la taille du sous ensemble. Le but est de sélectionner le sous-ensemble qui optimise les deux objectifs en même temps. [8]

•

La figure ci-dessus montre Les méthodes utilisées pour évaluer un sous-ensemble de caractéristiques. Les algorithmes de sélection peuvent être classées en trois catégories principales : "Filter", "Wrapper" et "Embedded". Nous présentons deux module Filter et wrapper

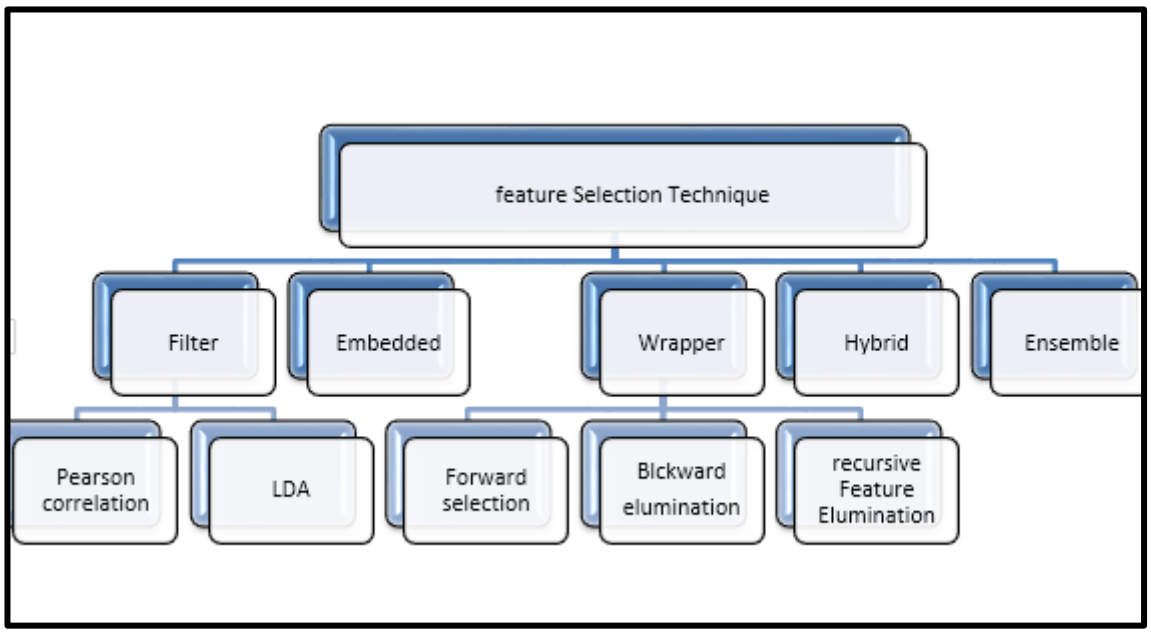


Figure 6: Techniques de sélection des caractéristiques [8]

- **Le modèle Filtter** : a été le premier utilisé pour la sélection de caractéristiques. Dans celui-ci, le critère d'évaluation utilisé évalué la pertinence d'une caractéristique selon des mesures qui reposent sur les propriétés des données d'apprentissage. Cette méthode est considérée, d'avantage comme une étape de prétraitement (filltrage) avant la phase d'apprentissage. En d'autres termes, l' évaluation se fait généralement indépendamment d'un classificateur. [8]
- **Le modèle Wrapper :** Le principal inconvénient des approches "Fillter" est le fait qu'elles ignorent l'influence des caractéristiques sélectionnées sur la performance du classificateur à utiliser par la suite. Pour résoudre ce problème, Les méthodes "wrapper", appelées aussi méthodes enveloppantes, évaluent un sous-ensemble de caractéristiques par sa performance de classification en utilisant un algorithme d'apprentissage [8]

.II.3.1.2 Réduction basée sur une transformation de données

La réduction de la dimensionnalité par une transformation de données (appelée aussi extraction de caractéristiques) ne se fait pas par une sélection de certaines caractéristiques, mais par une construction de nouvelles caractéristiques obtenues en combinant les caractéristiques initiales. Une transformation de données risque de faire perdre la sémantique de l'ensemble initial de caractéristiques et donc l'utilisation de cette famille de méthodes n'est applicable que dans le cas où la sémantique n'intervient plus dans les étapes qui suivent la réduction.

Les sections suivantes décrivent brièvement plusieurs techniques de réduction connues. Elles sont généralement groupées en deux catégories : les méthodes linéaires et les méthodes non linéaires.

La figure suivante décrit le processus global de la méthode d'extraction de caractéristiques.

Figure 7: Le processus d'extraction de caractéristique [8]

II.3.2 Techniques de réduction de dimensionnalité

La plupart des techniques de réduction dimensionnelle peuvent être regroupées en deux catégories, les approches linéaires et non linéaires. Ci-dessous, quelques propositions représentatives trouvées dans la littérature, celles qui peuvent être considérées comme des méthodes traditionnelles, sont décrites. En général, des propositions classiques de réduction dimensionnelle ont été développées à l'aide de techniques linéaires. Voici quelques-uns d'entre eux : [9].

.II.3.2.1 Méthode linéaire

.II.3.2.1.1 Analyse en Composantes Principales

L'Analyse en Composantes principales (ACP) fait partie du groupe des méthodes descriptives multidimensionnelles appelées méthodes factorielles.

L'ACP est une technique qui permet de trouver des espaces de dimensions plus petites dans lesquels, il est possible d'observer au mieux les individus. Sa démarche essentielle consiste à transformer les variables quantitatives initiales, plus ou moins corrélées entre elles, en des variables quantitatives, non corrélées, combinaisons linéaires des variables initiales et appelées composantes principales. Les composantes principales sont donc de nouvelles variables indépendantes, combinaisons linéaires des variables initiales, possédant une variance maximale. Globalement l'ACP consiste à rechercher la direction suivant laquelle le nuage de points des observations s'étire au maximum. A cette direction correspond la première composante principale. La seconde composante principale est déterminée de telle sorte qu'elle soit la plus indépendante possible de la première ; elle est donc perpendiculaire à celle-ci.

Ces deux composantes forment le premier plan principal. Cette opération est Erétrie de Manière à trouver toutes les composantes principales expliquant le maximum de variance.

La figure suivante montre à gauche, un exemple de données linéaires et à droite le résultat de leur projection dans un plan généré par les deux premières composantes principales calculées sur ces données. [10]

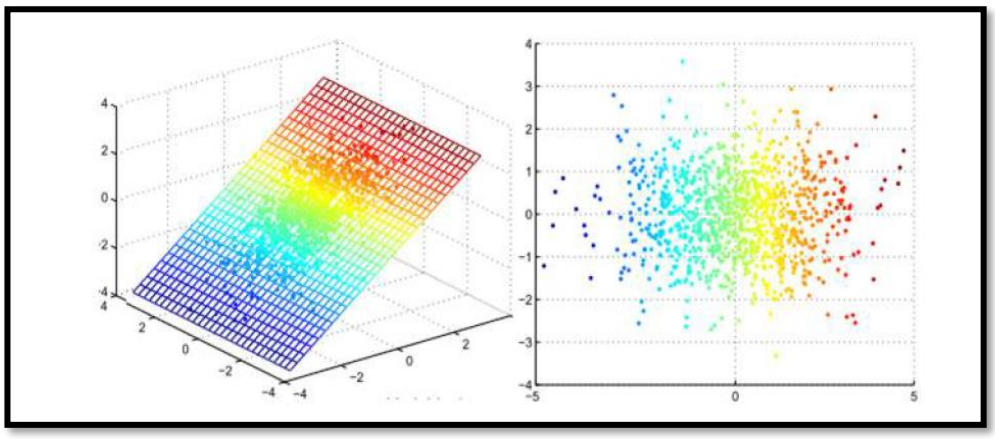


Figure 8: Principe d'ACP sur des données linéaires [10]

Supposons que nous ayons un ensemble de données : $X = \{X_1, X_2, \dots, X_N\}$Composé de M observations ou chaque observation $x_i = \{x_{i1}, x_{i2}, x_{iN}\}$ est composée de N caractéristiques.

Xest associé à une matrice de données Ade taille $N * M$ ou chaque colonne représente une caractéristique. En pratique, le calcul de l'ACP pour la matrice X revient à réaliser les opérations ci-dessous afin de trouver les composantes principales :

1. Calculer le vecteur $\mu = (\mu_1, \mu_2, \dots, \mu_m)^T$ qui représente le vecteur moyen ou μ_i est la moyenne de la i ème composante des données.

2. Calculer la matrice X en soustrayant le vecteur moyen a toutes les colonnes de A dans le but d'obtenir des données centrées.

3. Calculer la matrice S (de taille $\times N$) de covariance de X avec ($S = X \times X^T$).

4. Calculer la matrice e U (de taille $\times N$) qui est composée des coordonnées des vecteurs propres $\overrightarrow{u_j}$de S triés par ordre décroissant des modules des valeurs propres λ_j (la première colonne de U est le vecteur propre qui correspond à la plus grande valeur propre)

5. Garder les R premières colonnes de U pour former la matrice $\vec{U}: N \times R$qui représente les R premières composantes principales.

L'ACP étant une méthode de réduction de dimension, il est important de savoir qu'elle ne peut pas retenir la totalité de l'information contenue dans le nuage de points initial. Enfin, l'ACP prend uniquement en compte les dépendances linéaires entre les variables et ne peut donc pas fournir une projection pertinente pour une distribution non-linéaire de points.

La figure suivante montre à gauche, un exemple de données non-linéaires (non réparties dans un plan) et à droite le résultat de leur projection dans un plan généré par les deux premières composantes principales calculées sur ces données. [10]

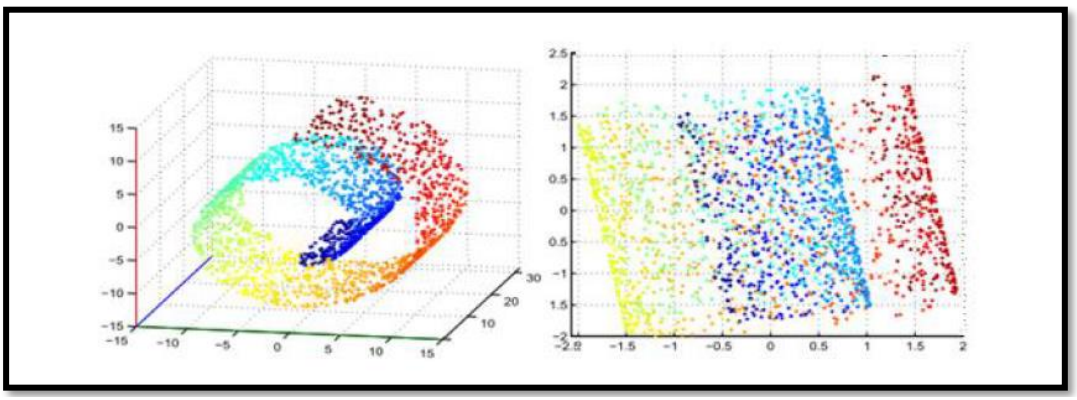


Figure 9: Application d'ACP sur des données non linéaire [10]

.II.3.2.2 Analyse Linéaire Discriminante

L'analyse linéaire discriminante, appelée aussi analyse discriminante linéaire de Fisher, est une méthode de réduction du nombre de dimensions proposée par Fisher en 1936 (Fisher [1936]). Cette méthode s'applique lorsque les classes des individus sont connues. L'idée de Fisher a été de créer une méthode pour choisir entre les combinaisons linéaires des variables celles qui maximisent l'homogénéité de chaque classe.

L'analyse discriminante linéaire (LDA) est une technique de réduction dimensionnelle qui cherche la meilleure façon possible de faire la distinction entre les classes du sous-espace sous-jacent. Plutôt que de faire de la discrimination sur la base de données. Formellement, il produit les plus grandes différences moyennes entre les catégories de résultats souhaités en utilisant des caractéristiques indépendantes par rapport aux données décrites. Son objectif est de formuler une projection A telle qu'elle maximise le rapport de Sb et Sw (critère de Fisher) qui sont entre les classes et dispersion dans la classe, respectivement comme dans l'équation suivante

$$arg\,\max_{A} \frac{|AS_bA^T|}{|AS_wA^T|}$$

Figure 10:Equation de LDA

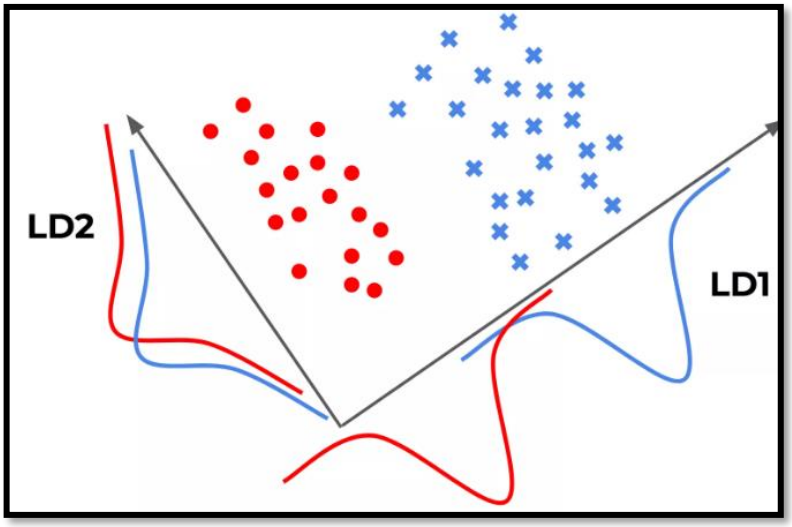


Figure 11:Principe de LDA

.II.3.2.3 Positionnement Multidimensionnel MDS

La méthode de positionnement multidimensionnel (Messick et Abelson [1956]), MDS (Multi Dimensional Scaling) en anglais, permet de construire une représentation en faible dimension des points de l'espace. Son objectif est de construire, à partir d'une matrice de distances ou des mesures de similarité calculées sur chaque paire de points, une représentation euclidienne des individus dans un espace de dimension réduite qui préserve "au mieux" ces distances.

Malgré leur popularité, les solutions linéaires classiques de réduction dimensionnelle posent le problème de ne pas pouvoir traiter correctement des données non linéaires complexes. Pour cette raison, des propositions non linéaires pour la réduction dimensionnelle sont apparues, entre autres. Ces techniques permettent de travailler correctement avec des données non linéaires complexes. C'est un avantage lorsqu'on travaille avec des données réelles, qui sont habituellement de ce type. [9]

.II.3.2.1 Méthode non linéaire

Les méthodes non linéaires ont pour objectif d'optimiser les représentations afin qu'elles reflètent au mieux la topologie initiale des données.

.II.3.2.1.1 Isomap

Isomap est une technique de réduction de dimensions qui, comme la méthode de positionnement multidimensionnel (MDS), part de la connaissance d'une matrice de dissimilaires entre les paires d'individus. Le but est cette fois de trouver une variété (non linéaire) contenant les données.

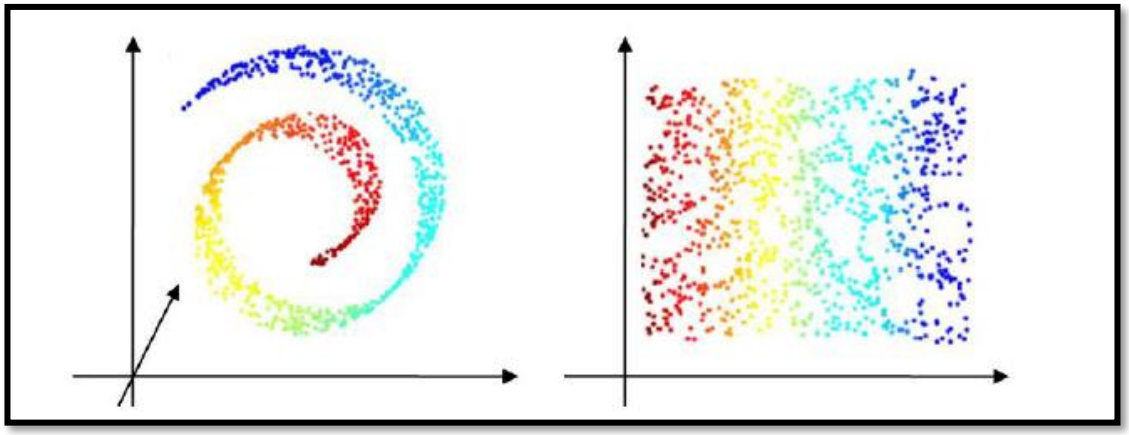


Figure 12 : Isomap sur les données non linéaire.

.II.3.2.1.2 T-SNE

L'algorithme t-SNE se base sur une interprétation probabiliste des proximités. Une distribution de probabilité est définie sur les paires de points de l'espace d'origine de telle sorte que des points proches l'un de l'autre ont une forte probabilité d'être choisis tandis que des points éloignés ont une faible probabilité d'être sélectionnés. Une distribution de probabilité est également définie de la même manière pour l'espace de visualisation. L'algorithme t-SNE consiste à faire concorder les deux densités de probabilité, en minimisant la divergence de Kullback-Leibler entre les deux distributions par rapport à l'emplacement des points sur la carte.

Cette technique est extrêmement populaire dans la communauté d'apprentissage en profondeur. Malheureusement, la fonction de coût de t-SNE implique des machines mathématiques non triviales et nécessite des efforts importants pour la comprendre.

Mais, grosso modo, ce que t-SNE essaie d'optimiser, c'est la préservation de la topologie des données. Pour chaque point, il construit une notion dont les autres points sont ses «voisins», en essayant de faire en sorte que tous les points aient le même nombre de voisins. Ensuite, il essaie de les incorporer afin que ces points aient tous le même nombre de voisins.

II.4 Apprentissage automatique et classification supervisée

L'apprentissage supervisé (supervised learning en anglais) est une tâche d'apprentissage automatique consistant à apprendre une fonction de prédiction à partir d'exemples annotés, au contraire de l'apprentissage non supervisé. On distingue les problèmes de régression des problèmes de classement. Ainsi, on considère que les problèmes de prédiction d'une variable quantitative sont des problèmes de régression tandis que les problèmes de prédiction d'une variable qualitative sont des problèmes de classification

Les méthodes de classification ont pour but d'identifier les classes auxquelles appartiennent des objets à partir de certains traits descriptifs. Elles s’appliquent a un grand nombre d'activités humaines et conviennent en particulier au problème de la prise de décision automatisée. La procédure de classification sera extraite automatiquement à partir d'un ensemble d'exemples. Un exemple consiste en la description d'un cas avec la classification correspondante.

Dans le cadre de cette projet, nous parlerons surtout de l'apprentissage supervisé. Dans la suite, nous présentons les principaux algorithmes d classification supervisée proposés dans la littérature. Il ne s'agit pas de faire une présentation exhaustive de toutes les méthodes mais seulement de préciser les méthodes les plus classiques que nous utiliserons dans le cadre de notre travail en fonction de leurs propriétés particulières.

II.4.1 Approches de classification

Plusieurs approches ont été utilisées dans le domaine de la reconnaissance de l'écriture manuscrite en ligne et hors ligne comme les méthodes statistiques, les méthodes structurelles, le réseau neuronal et les méthodes syntaxiques. Certains systèmes de reconnaissance identifient les traits, d'autres appliquent la reconnaissance à un seul caractère ou à des mots entiers. Le système de reconnaissance de l'écriture manuscrite est donc un moyen de communication entre l'homme et les machines.Il existe trois approches principales et une approche hybride :

.II.4.1.1 Approche statistique

Depuis les débuts de la reconnaissance des formes, de nombreuses méthodes exploitants une description statistique d'attributs ont été développées. L'objet de ces approches est de pouvoir décrire des comportements spécifiques à partir de modèles simples à utiliser.

- Les méthodes paramétriques (la règle de Bayes, les réseaux de neurones, les chaînes de Markov…)
- Les méthodes non paramétriques (méthode des k plus proches voisins, …) [5]

Le schéma général d'exploitation est de prendre, un ou plusieurs attributs de description de forme et de décrire l'organisation des individus dans l'espace des attributs à travers des modèles.

.II.4.1.2 Approche structurelle

D'un point de vue général, la classification structurelle utilise une décomposition du caractère en objets primitifs et permet de décrire une partie de l'organisation de ces objets les uns par rapport aux autres.

Plusieurs types de représentation permettent de stocker ces structures, soit à travers des cartes topologiques, soit par l'intermédiaire de graphes d'adjacences. [5]

.II.4.1.3 Approche stochastique

L'approche stochastique utilise un modèle de la reconnaissance, prenant en compte la grande variabilité de la forme. Dans ce type d'approche, les modèles sont souvent discrets et de nombreux travaux reposent sur la théorie des champs de Markov et l'estimation bayésienne.

Les champs de Markov permettent de ramener des propriétés globales à des contraintes locales. Le modèle décrit ces états à l'aide de probabilités de transitions d'états et de probabilités d'observation par état.

La comparaison consiste à chercher dans ce graphe d'état, le chemin de probabilité forte correspondant à une suite d'éléments observés dans la chaîne d'entrée. Les méthodes les plus répondues dans cette approche sont les méthodes utilisant les modèles de Markov cachés (HMM). [5]

.II.4.1.4 Approche hybride

Pour améliorer les performances de reconnaissance, la tendance aujourd'hui est de construire des systèmes hybrides qui utilisent différents types de caractéristiques et qui combinent plusieurs classificateurs en couches.

Pour surmonter les faiblesses de chaque approche et obtenir des résultats plus précis, meilleure que les résultats qui auraient été obtenus en cas de l'application de chaque approche séparément, comme les approche qui ont été fusionnés, pour former celui intégré. [5].

II.4.2 Techniques de classification supervisée

Dans le développement des systèmes OCR, la plupart des classificateurs peuvent être utilisés pour la classification tels que les classificateurs statistiques paramétriques et non paramétriques, KNN, SVM, CNN, RF et les classificateurs hybrides, etc.

.II.4.2.1 K plus proche voisin

Le classificateur K plus proche voisin ou (Knn k-nearest neighbor en anglais)est l'algorithme de classification d'image le plus simple. La classification k plus proche voisin n'apprend rien. Cet algorithme repose sur la distance entre les vecteurs de caractéristiques.

L'algorithme K plus proche voisin classe les points de données inconnus, en trouvant la classe la plus courante parmi les k exemples les plus proches. Chaque point de données dans le k le plus proche exprime un vote et le plus grand nombre de votes de catégorie gagne.

Dans la figure suivante à gauche, la classification est simple quel que soit le nombre de voisins choisis : le nouvel objet est noir. A droite, en revanche, tout dépend du nombre de voisins choisis et de l'heuristique de classification.

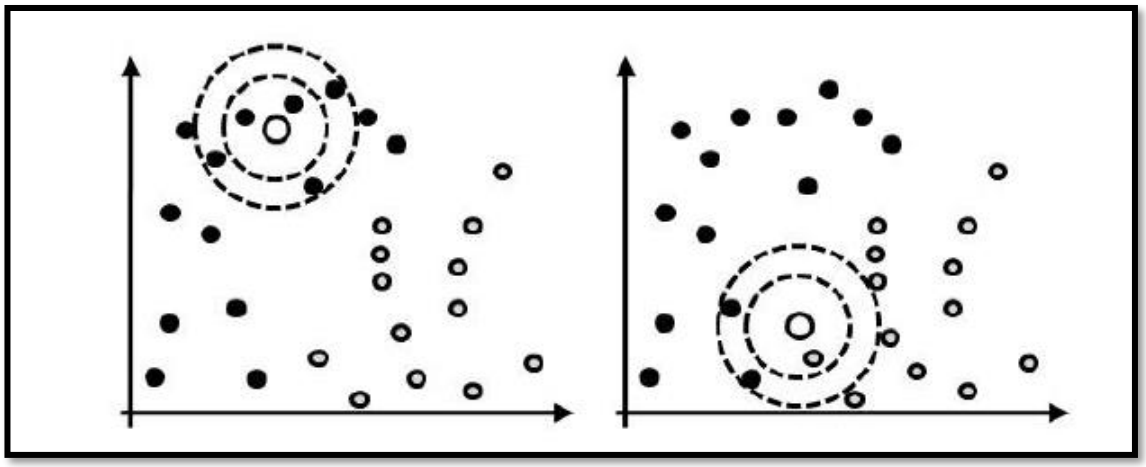


Figure 13: Principe de l'algorithme KNN

- Pour k = 1, le nouvel objet est gris.
- Pour k= 3, si les trois voisins ont le même poids, alors le nouvel objet est noir.
- Par contre, si le poids est pondéré par l'inverse de la distance alors le nouvel objet peut être gris. Cela revient à pondérer l'affectation de classe avec la distance : plus un voisin est éloigné, plus son influence est faible.

Pour effectuer une prédiction, l'algorithme KNN va se baser sur le jeu de données en entier. En effet, pour une observation qui ne fait pas parti du jeu de données, qu'on souhaite prédire, l'algorithme va chercher les K instances du jeu de données les plus proches de notre observation au sens de la distance utilisée. Ensuite pour ces K voisins, l'algorithme se basera sur leurs variables de sortie y pour calculer la valeur de la variable y de l'observation qu'on souhaite prédire . [11]

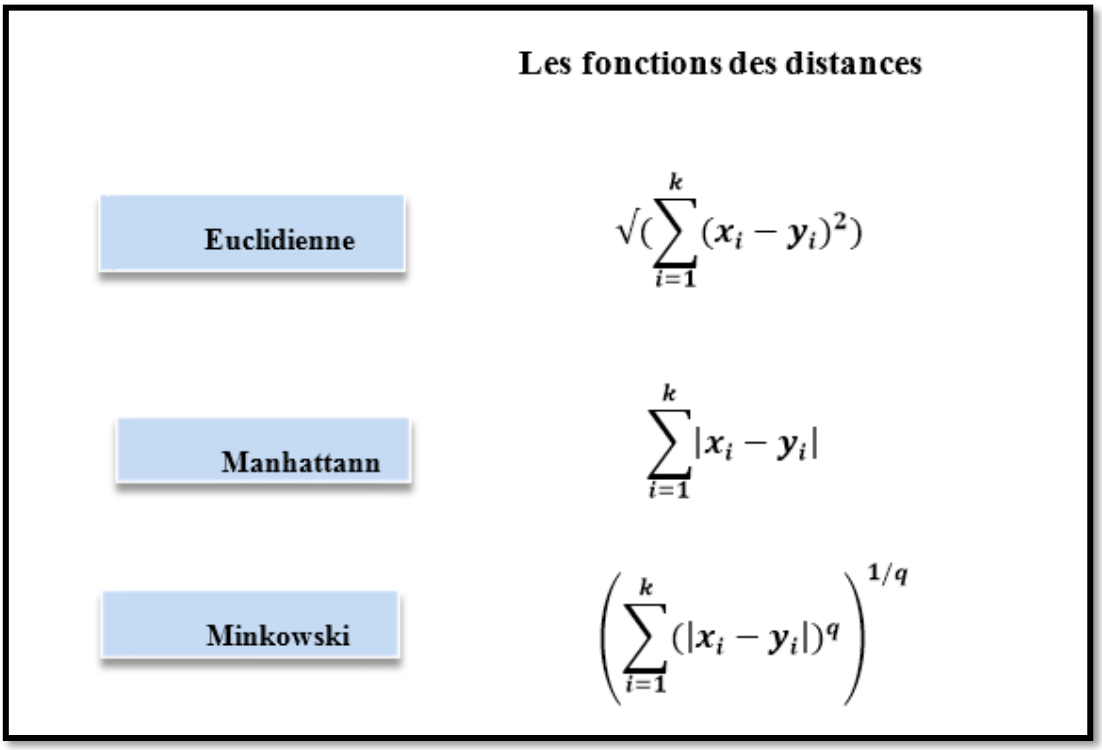


Figure 14: Les différentes fonctions de distance utilisées dans KNN

.II.4.2.2 Séparateurs à vastes marges

Les séparateurs à vastes marges ou les SVMs sont un ensemble de techniques d'apprentissage supervisé destinées à résoudre des problèmes de la classification binaire et de la régression. Les SVMs reposent sur deux idées principales à savoir la notion de la marge maximale et la notion de la fonction noyau.

La marge maximale est employée pour les problèmes de la classification linéaire. Elle représente la distance entre la frontière de séparation et les échantillons d'apprentissages les plus proches. Ces derniers sont les vecteurs supports.

Les fonctions noyau sont employées dans le cas des problèmes de la classification non-linéaire pour transformer l'espace de représentation des données d'entrées en un espace de plus grande dimension ,dans lequel il est probable qu'il existe de séparateurs linéaires. [11]

Le classificateur SVM est un algorithme qui maximise la marge entre les classes du problème à résoudre et réduit au minimum l'erreur de classification. L'objectif de la marge maximale est de faire séparer deux classes par un hyperplan, de telle sorte la distance par rapport aux vecteurs supports soit maximale . [11]

Dans la tâche de classification, un SVM construit l'hyperplan optimal de séparation des attributs caractéristiques dans un espace de haute dimension. Le calcul de cet hyperplan est fondé sur la maximisation de la marge entre les exemples d'apprentissages les plus proches qui appartiennent à différentes classes.

La représentation graphique du fonctionnement d'un classificateur SVM est la suivante :

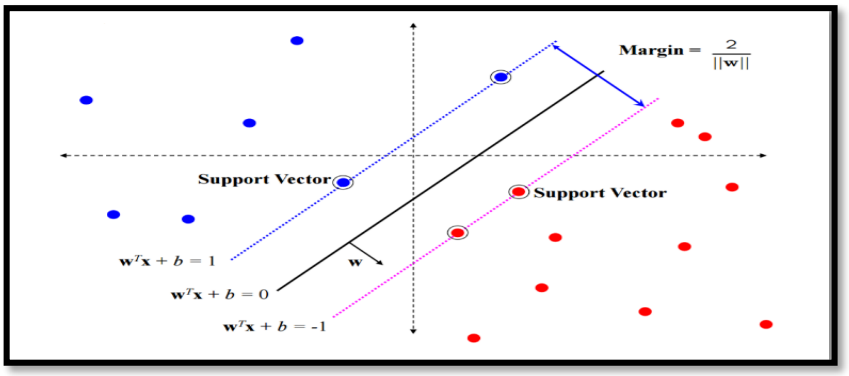


Figure 15:Principe de SVM [11]

Certains instituts de recherche ont proposé SVM comme classificateur d'apprentissage pour le contrôle de la capacité avec des problèmes de régression et de classification binaire. Il a également été certifié comme étant très excellent dans de nombreuses autres applications, telles que la détection de visages, la classification de texte et la reconnaissance de nombres manuscrits. Par exemple, un article de 2017 de Phangtriastu, Harefa et Tanoto a atteint la plus haute précision de reconnaissance de caractères manuscrits de 94,43% ,en utilisant le classificateur SVM avec la combinaison d'algorithmes d'extraction de caratéristiques ,qui sont l'histogramme de projection et HOG. [12]

.II.4.2.3 Forêts aléatoires

Les forêts aléatoire RF est une méthode d'ensemble utilisée pour la classification ou la régression. RF fonctionne à l'aide d'une énorme collection d'arbres décisionnels dissociés. Dans cette matrice, les données d'apprentissage forment une matrice en entrée. En utilisant cette matrice, un grand nombre de nouvelles matrices, avec des éléments aléatoires sont créées.

En utilisant chacune de ces matrices, un arbre décisionnel correspondant est formé pour la classification des données de test. Lorsque les données de test sont entrées, tous ces arbres de décision classent les données de test d'entrée et prédisent la classe à laquelle l'entrée

appartient. Le résultat trouvé est basé sur le résultat de prédiction qui a le nombre maximum comme résultat des classificateurs. Pour faire des prédictions, une fois l'apprentissage est terminée, la moyenne des prédictions de toutes les arbres de régression individuelles est prise à l'aide de la formule suivante [11].

$$\hat{f} = \frac{1}{B}\sum_{b=1}^{B} \widehat{f_b}(x')$$

Figure 16: Equation des RFs

.II.4.2.1 Réseaux neuronaux convolutifs

Un réseau neuronal convolutif (CNN) est un type de réseau neuronal artificiel d'alimentation, dans lequel le modèle de connectivité entre ses neurones est inspiré par l'organisation du cortex visuel animal [11].

Les réseaux neuronaux convolutifs sont constitués de neurones ,qui ont des poids et des biais. Chaque neurone reçoit une entrée, puis il effectue un produit scalaire et éventuellement le suit avec une non-linéarité. [11]

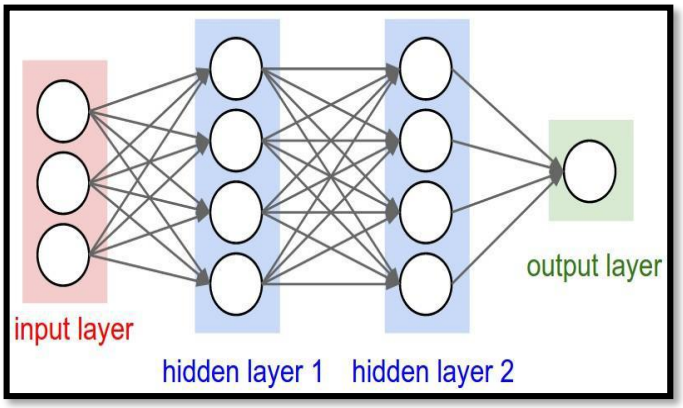


Figure 17: Configuration de base du réseau neuronal convolutif

L'ensemble du réseau neuronal convolutif exprime une fonction de Scorefunction différenciable qui est ensuite suivie d'une fonction Softmax. L'entrée de données dans le réseau neuronal convolutif est organisée sous la forme de sa largeur, sa hauteur et sa profondeur comme indiqué sur la figure ci-dessous :

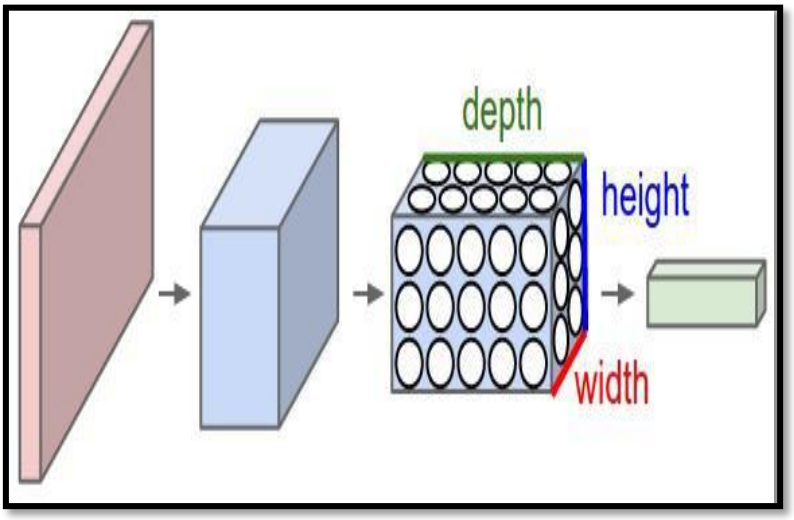


Figure 18 : Disposition des neurones dans CNN

Le réseau neuronal convolutif se compose de beaucoup de couches. Ces couches lorsqu'elles sont utilisées à plusieurs reprises, conduisent à l'apprentissage d'un réseau neuronal profond. Trois principaux types de couches utilisées pour construire un CNN sont :

- **L'entrée** : cette couche contient les valeurs brutes des pixels de l'image.
- **Couche de convolution[CONV]:** Cette couche obtient les résultats de la couche neuronale qui est connectée aux régions d'entrée. Nous définissons le nombre de filtres à utiliser dans cette couche. Chaque filtre peut être une fenêtre 5x5 qui glisse sur les données d'entrée et obtient le pixel avec l'intensité maximale que la sortie.
- **Couche unité linéaire rectifiée [ReLU]:** cette couche applique une fonction d'activation par éléments surdonnées d'image [10]. Nous savons qu'un CNN utilise la rétropropagation. Donc, afin de conserver les mêmes valeurs des pixels et de ne pas être modifiés par la propagation arrière, nous appliquons la fonction ReLU.
- **Couche de regroupement**: cette couche effectue une opération de sous-échantillonnage le long des dimensions spatiales (largeur, hauteur), résultant en volume.
- **Couche entièrement connectée[FC]:** cette couche est utilisée pour calculer les classes de score, c'est-à-dire quelle classe a le score maximum correspondant aux chiffres d'entrée [11].

II.5 Travaux réalisés

Pour concevoir un système de reconnaissance d'un chiffre manuscrite, il faut avoir les diffèrent travaux ultérieurs qui 'ils ont réalisé sur la base de données MNIST. Dans ce chapitre, nous décrivons de ces travaux. En précisant les performances et les résultat obtenus.

Dans cette section, on va présenter les différents travaux consacrés à la reconnaissance des chiffres manuscrits en fonction de l'approche utilisé et le résultat obtenu.

II.5.1 Nurul Ilmi ,W Tjokorda Agung Bud, R Kurniawan Nur(2016)

Nurul Ilmi, Tjokorda Agung Budi W et Kurniawan Nur R dans leur article « Handwriting Digit Recognition using Local Binary Pattern Variance and K-Nearest Neighbors Classification », utilisant Local Binary Pattern (LBP) comme extraction de caractéristiques et la classification KNN sur leur système de reconnaissance d'écriture manuscrite sur le formulaire C1, utilisé par la Commission générale des élections en Indonésie. Le résultat du test est que la variance LBP peut reconnaître le caractère numérique de l'écriture manuscrite sur le jeu de données MNIST avec une précision de 89,81% et pour les données du formulaire C1, la précision est de 70,91%. [13]

II.5.2 Wu et Zhang (2010)

Wu et Zhang ont utilisé les caractéristiques de direction extraites pour la réduction de la dimensionnalité. K-ième voisin le plus proche, les modèles de mélanges gaussiens et SVM se révélant les meilleurs candidats pour les caractéristiques. Le taux de reconnaissance atteint 98.81% sur la base de donnée MNIST qui constituer 60 000 images d'apprentissage et 10 000 images de test. En utilisant 3-NN. [4]

II.5.3 Bernard, S., Adam, S., & Heutte, L. (2007)

Dans le domaine de la reconnaissance des formes, les chercheurs ont accordé plus d'attention, ces dernières années, aux systèmes multi-classificateurs, en particulier l'ensachage et le boosting. Bernard, Adam et Heutte ont étudié une technique d'extraction de caractéristiques conventionnelle basée sur une pyramide multi-résolution en niveaux de gris, pour découvrir l'effet des valeurs des paramètres sur les performances de la RF. Ils ont expérimenté l'algorithme Forest-RI, qui est considéré comme la méthode de référence de la forêt aléatoire, sur la base de données numérique manuscrite du MNIST et ils ont atteint un niveau de précision dans la reconnaissance des chiffres manuscrits supérieur à 93%. [14].

II.5.4 Swapna Prava Ekka Dr. Samit Ari(2014)

D'après « Recognition of Handwritten Digits using Proximal Support Vector Machine » le système basé sur la fonction Histogram of Oriented Gradient (HOG), et qui utilise la machine vectorielle de support proximal basé sur le classificateur SVM standard, prend moins de temps et les performances du classificateur PSVM sont meilleures que le réseau neuronal artificiel. Le total de 20 000 échantillons est prélevé pour les données d'apprentissage et de test (1 000 échantillons pour un chiffre). La PSVM linéaire de classe 10 a obtenu 98,65% avec un entraînement de 59 millisecondes. Le système a également maintenu une petite dimension pour le vecteur caractéristique, et moins de temps d'apprentissage.sans inclure une réduction dimensionnelle supplémentaire.

Le score de 98.65%avec PSVM avec une réduction de temps (de 109 secondes vers 59 millisecondes)pour une classification PSVM sur 10 000 échantillons pour chacun des ensembles d'entraînement et d'entraînement		
Paramètres	ANN (100 epoch)	PSVM
Sensitivité(%)	91.84	93.22
Prédictive positive(%)	91.87	93.27
Spécificité(%)	99.09	99.25
Le score(%)	98.37	98.65

Figure 19:Le résultat de l'étude PSVM

II.5.5 Yogish Naik G R, Amani Ali Ahmed Ali (2018)

Le rapport « Handwritten Digits Recognition » présente la mise en œuvre de l'analyse en composantes principales (PCA) combinée avec un voisin le k plus proche voisin en atteignant une précision de 78,4% sur l'ensemble de données MNIST.La raison de l'inexactitude était l'algorithme le plus élémentaire pour la sélection et la classification des caractéristiques. Par conséquent, il est très difficile d'obtenir de bons

résultats par rapport à un système plus complexe. Pour améliorer la précision et obtenir de meilleurs résultats, les exemples dans l'apprentissage et les tests doivent être augmentés. Il existe de tels échantillons où il est même difficile pour un être humain de les classer, les similitudes entre les chiffres étant beaucoup plus élevées sur certaines images. [15]

II.5.6 Etude comparative (2017)

L'article [11] présente une comparaison temporelle entre l'apprentissage automatique (RFC, KNN, SVM) et l'apprentissage profond (Multi layer CNN) sur l'ensemble de données MNIST. Plus précisément, pour une réduction du temps d'apprentissage et de test, le GPU peut être utile et peut aider à obtenir le parallélisme et à obtenir de meilleurs résultats. Ci-dessous les calculs sur le processeur :

Comparaison des résultats	RFC	KNN	SVM	CNN
Le score classificateur entraîné(%)	99.71	99.71	99.71	99.71
Le score sur les images de test (%)	96.86	96.67	97.91	98.72
La durée d'entrainement (min)	10	15	14	70
La durée de test (min)	6	9	10	20

Figure 20 :Analyse comparative des différentes techniques de classification

Conclusion

Au cours de ce chapitre, nous avons traité le domaine de la réduction de dimensionnalité. Dans un premier temps, une revue du domaine de la sélection de caractéristiques a été présentée. Nous avons détaillé dans la deuxième partie de ce chapitre, les techniques de réduction par une transformation de données, en présentant différentes approches linéaires et non linéaires.

Ce chapitre a passé aussi en revue la littérature existante pertinente pour la recherche. Il a notamment mis en évidence de nombreuses techniques opérationnelles à prendre en compte, à savoir, le prétraitement des images, l’extraction des caractéristiques et les classificateurs pertinents de l'apprentissage automatique. Ces facteurs doivent être pris en compte lors de l'acquisition des ressources de données et de la préparation du plan, pour obtenir la plus haute précision de reconnaissance des chiffres manuscrits.

La reconnaissance des chiffres manuscrits est l'une des applications les plus indispensables de la reconnaissance de formes. De nombreux chercheurs ont étudié et identifié différents ensembles de données. Par exemple, Les performances d'un classificateur peuvent dépendre de la qualité des caractéristiques du classificateur lui-même

Cependant, de nombreux classificateurs tels que SVM et RF et KNN ne peuvent pas traiter efficacement des images ou des données brutes, car l'extraction des caractéristiques structurelles appropriées à partir de formes complexes est un défi considérable. Par conséquent, la façon d'utiliser la combinaison d'extraction de caractéristiques sophistiquées et de classificateur est le principal problème de l'OCR dans la reconnaissance de chiffres manuscrits.

Chapitre III: Approche proposée

Introduction

Ces dernières années, en raison de l'augmentation marquée des dimensions des données, les chercheurs ont proposé de nombreuses méthodes et techniques, pour réduire les dimensions des données élevées. Dans un système de reconnaissance des chiffres manuscrits, Le but ultime d'extraction de caractéristiques c'est d'obtenir le volume d'informations le plus pertinent qui sera fourni au système. C'est une étape critique lors de la construction d'un système de reconnaissance. L'une des raisons pour laquelle cette phase pose un problème c'est qui il s'accompagne d'une perte d'information. De ce fait, il faut effectuer un compromis entre la quantité et la qualité de l'information.

Dans ce chapitre nous présentons l'organigramme qui décrit l'approche proposée de notre système de reconnaissance des chiffres manuscrits. Ainsi une description de la base de données MNIST choisie pour tester notre approche proposée. Nous allons détailler les trois étapes essentielles pour construire notre système qui sont comme suites : prétraitement, extraction des caractéristiques à l'aide de techniques de réduction de dimensionnalité PCA, enfin la classification en mettons l'accent sur les éléments de l'algorithme KNN et le choix de ces paramètres.

III.1 L'organigramme général

Lees Étapes de notre processus est comme suit de:

1) Fixer l'objectif de recherche : laisser l'ordinateur reconnaître les nombres à partir d'images.
2) Acquérir les données : nous utiliserons l'ensemble de données MNIST disponible sur Internet.
3) Préparer les données : standardiser les images pour qu'elles soient toutes de la même taille et puis faire une linéarisation avant de réduire l'ensemble des caractéristiques de la base MNIST.
4) Construire le modèle : créer le modèle de classification KNN.
5) Présentation des scores : rapporter les résultats

L'organigramme de cette combinaison est illustré dans la figure suivante.

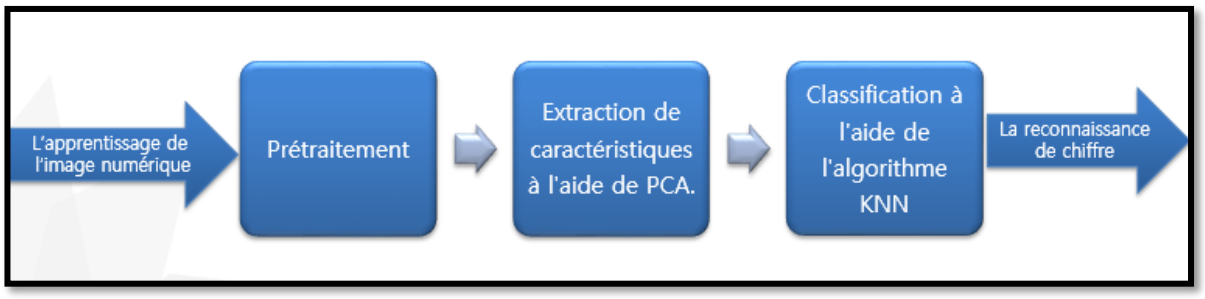


Figure 21 :L'organigramme pour la combinaison de prétraitement, et K-NN

Notre travail pour résoudre ce problème de reconnaissance numérique manuscrite peut être largement divisée en trois blocs:

- Prétraitement
- Extraction de caractéristiques à l'aide de PCA.
- Classification à l'aide de l'algorithme KNN.

III.1.1 Prétraitement

Le premier bloc est essentiellement composé de toute étapes de traitement impliquées directement à partir de l'image (standardisation, normalisation, linéarisation). Après le traitement de l'image, les caractéristiques de l'image sont extraites avec la technique d'extraction des caractéristiques proposée le classifieur facilite la prochaine étape de reconnaissance, c'est-à-dire la reconnaissance réelle des chiffres à l'aide de classifier KNN.

III.1.2 Extraction de caractéristiques à l'aide de PCA

Nous allons discuter le fonctionnement de l'analyse en composantes principales (ACP) et de la façon dont elle peut être utilisée comme technique de réduction de dimensionnalité pour les problèmes de classification. La question se pose alors de savoir quelles caractéristiques devraient être préférées et lesquelles devraient être supprimées d'un vecteur de caractéristiques de grande dimension.

Si toutes les caractéristiques étaient statistiquement indépendantes, on pourrait simplement éliminer les caractéristiques les moins discriminantes de ce vecteur. Les caractéristiques les moins discriminantes peuvent être trouvées par diverses approches de sélection de caractéristiques gourmandes. Cependant, dans la pratique, de nombreuses caractéristiques dépendent les unes des autres ou d'une variable inconnue sous-jacente. Une seule caractéristique pourrait donc représenter une combinaison de plusieurs types

d'informations par une seule valeur. La suppression d'une telle fonctionnalité supprimerait plus d'informations que nécessaire. Dans les paragraphes suivants, nous présentons ACP comme solution d'extraction de caractéristiques.

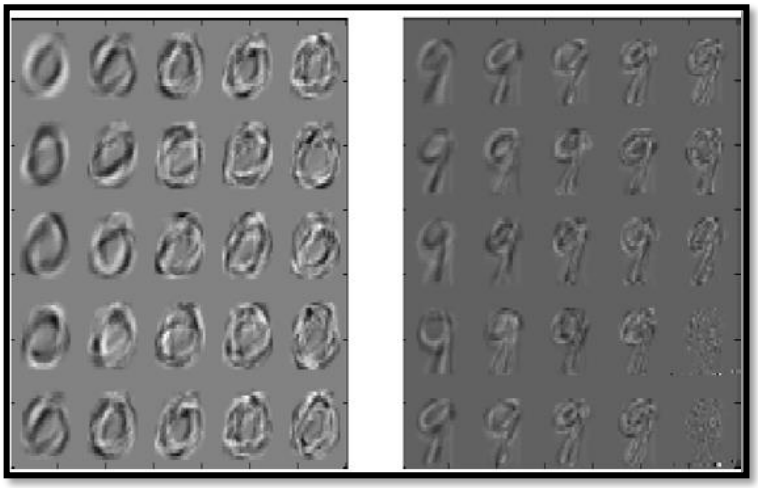


Figure 22 :Echantillon généré par PCA (Chiffres propres pour les chiffres 0 et 9)

L'ACP est une technique d'analyse multivariée basée sur des vecteurs propres qui extrait généralement la meilleure variance de données. L'autre avantage principal de l'ACP est qu'une fois les motifs détectés dans les données, puis les données sont réduites sans trop de perte d'informations. Dans cette expérience, l'ACP sera utilisé pour minimiser la proportion de l'ensemble de données MNIST de 784 à une valeur inférieure pour faciliter les calculs. Les équations mathématiques appliquées pour implémenter l'ACP sont décrites en détail ci-dessous. Considérons un groupe de n observations sur le vecteur de p variables formées dans une matrice X (n x p).

$$\{X_1, X_2, \dots, X_n \in R^p\}$$

L'approche ACP trouve p composantes principales, et chacune est une dé-corrélation linéaire des colonnes de la matrice X, dans laquelle les poids sont des facteurs d'un vecteur propre à la matrice de corrélation ou de covariance de données. La condition est que les données soient concentrées et normalisées. La première composante principale de la transformation linéaire est:

$$Z_1 = a_1^T x_j = \sum_{i=1}^{p} a_{i1} x_{ij} \, , j = 1, \dots, n$$

Où :

$$a_1 = (a_{11}, a_{21}, \dots, a_{p1})$$

Si il sont choisis comme tels, la variance de $z_1$est maximale. Chaque composante principale commence à partir de l'origine des axes des ordonnées. $a_1 et\ x_j$ [16] .

III.1.3 Réduction de dimensionnalité dans MNIST

La base MNIST (Modified National Institute of Standards and Technology database) est un ensemble de données constitué d'un jeu d'apprentissage de 60.000 chiffres décimaux manuscrits et d'un jeu de test de 10.000 chiffres de même nature, mais différents de ceux contenus dans le jeu d'apprentissage. [17]

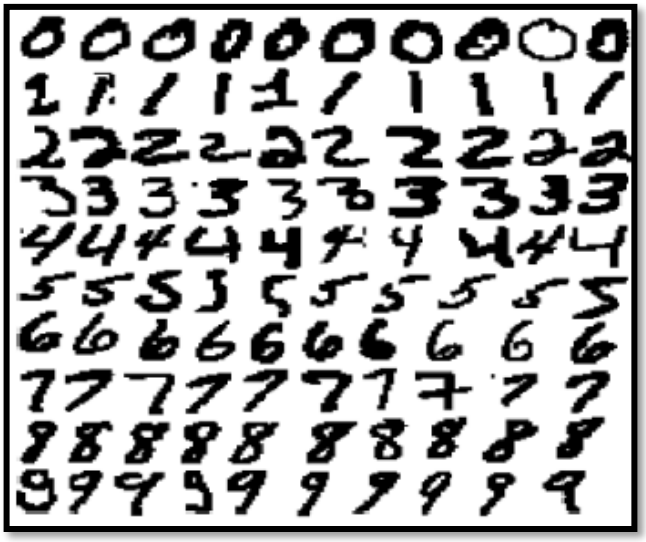


Figure 23 :Exemples d'images de la base de données de chiffres du MNIST [18]

Les images sources, sélectionnées par Chris Burges et Corinna Cortes, ont été initialement codées dans des matrices 20 x 20, puis converties en matrices de 28 lignes par 28 colonnes, chaque composante de la matrice étant codée sur un octet (donc avec 256 valeurs possibles, de 0 à 255) et représentant une quantité d'encre donnée, la valeur 0 indiquant qu'il n'y a pas d'encre dans la case (case blanche) et la valeur 255 indiquant que la case est complètement remplie d'encre (case noire). Les chiffres manuscrits ont été positionnés dans les matrices de façon à placer leur centre de masse au centre de la matrice. Bien évidemment,

la suite d'octets constituée par cet ensemble de chiffres manuscrits ne peut pas être visualisée sans l'aide des outils approprié.(l'interpréteur RPN). [19]

Les quatre fichiers sont disponibles dans le conteneur directement :

- train-images-idx3-ubyte.gz : images du jeu d'apprentissage (9912422 octets).
- train-labels-idx1-ubyte.gz : étiquettes du jeu d'apprentissage (28881 octets).
- t10k-images-idx3-ubyte.gz : images du jeu de tests (1648877 octets).
- t10k-labels-idx1-ubyte.gz : étiquettes du jeu de tests (4542 octets).
-

MNIST est un simple jeu de données de vision par ordinateur. Il se compose d'images de 28 x 28 pixels de chiffres manuscrits, tels que : Chaque point de données MNIST, chaque image, peut être considéré comme un tableau de nombres décrivant le degré d'obscurité de chaque pixel. Par exemple, nous pourrions penser à 1 comme quelque chose comme :

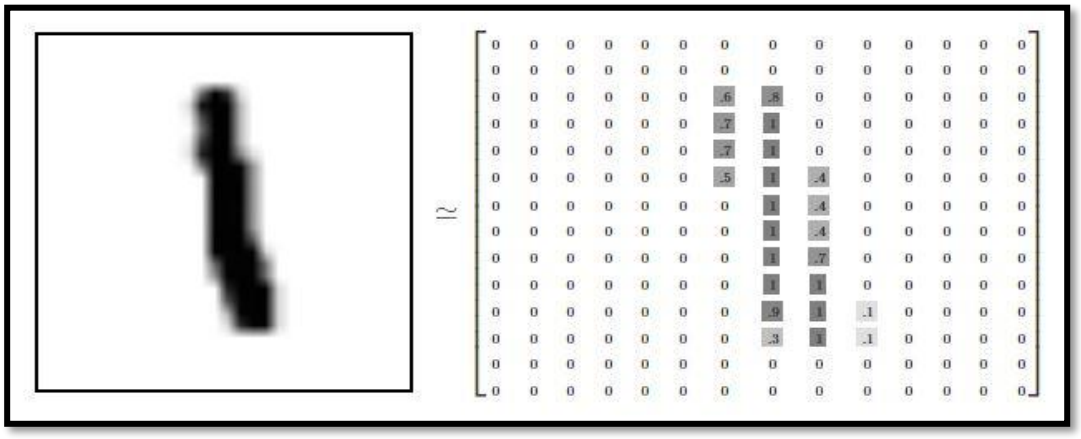


Figure 24 : La description de 1 dans la base MNIST [20]

Puisque chaque image a 28 par 28 pixels, nous obtenons un tableau de 28x28. Nous pouvons aplatir chaque tableau en un vecteur dimensionnel $28 * 28 = 784$. Chaque composante du vecteur est une valeur comprise entre zéro et un décrivant l'intensité du pixel. Ainsi, nous considérons généralement MNIST comme une collection de vecteurs de 784 dimensions.

La raison pour laquelle la réduction de dimensionnalité est utile, est que les données de plus petite dimension peuvent être traitées plus rapidement. Cette opération est cruciale en apprentissage automatique par exemple, pour lutter contre le fléau de la dimension.

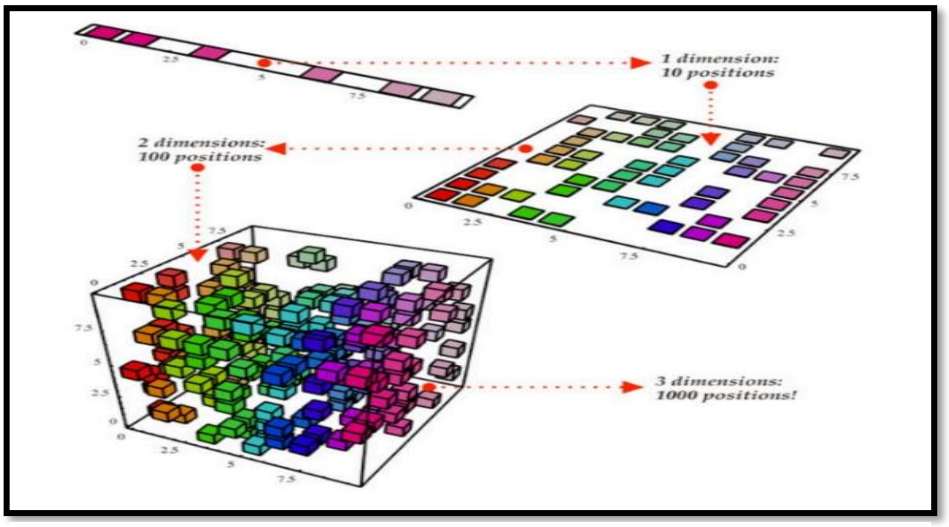


Figure 25: La réduction de dimensionnalité [20]

III.1.4 Choix de K plus proche voisin

.III.1.4.1 Propriétés de KNN

Les deux propriétés suivantes définiraient bien KNN :

- KNN est paresseux car il ne construit pas un modèle qui généralise un problème donné dans la phase d'apprentissage.
- Il se base uniquement sur les données d'entrainement « or K est un hyper paramètre n'est pas appris automatiquement à partir de jeu d'entrainement c'est à nous de l'optimiser à l'aide de jeu de donnés de test »

.III.1.4.2 Avantages de KNN

- C'est un algorithme très simple à comprendre et à interpréter.
- Il est très utile pour les données non linéaires car il n'y a pas d'hypothèse à propos des données dans cet algorithme.
- C'est un algorithme polyvalent car nous pouvons l'utiliser pour la classification ainsi que la régression.

.III.1.4.3 Inconvénients de KNN

- C'est un algorithme un peu coûteux en calcul car il stocke toutes les données d'entraînement.
- Stockage de mémoire élevé requis par rapport à d'autres algorithmes d'apprentissage supervisé.
- Pas de formule ou règle pour déterminer la valeur de K.

L'un des principaux avantages de l'algorithme k-NN est qu'il est extrêmement simple à mettre en œuvre et à comprendre. En outre, KNN ne prend absolument pas de temps à s'entrainer, car tout ce que nous devons faire est de stocker nos points de données pour les besoins de calcul ultérieur des distances à eux et d'obtenir notre classification finale.

III.2 KNN et le fléau de la dimensionnalité

L'algorithme des voisins les plus proches de k dépend de la proximité des points de données. Cela devient difficile, à mesure que le nombre de dimensions augmente, ce qu'on appelle le « fléau de la dimensionnalité », et l'ajout d'une nouvelle dimension crée une autre occasion pour les points d'être plus éloignés. Comme le nombre de dimensions augmente, la distance la plus proche entre deux points se rapproche de la distance moyenne entre les points, éradiquant la capacité de l'algorithme des voisins k-les plus proches de fournir des prédictions précieuses.

III.2.1 Stratégie d'amélioration des performances de KNN

- Fast Similarity Search (FSS) : Méthodes qui basent leur performance sur la création d'un modèle de recherche permettant la récupération rapide de prototype.
- **Data Reduction (DR)** : Stratégie de prétraitement de donnés visent à réduire la taille d'ensemble d'apprentissage en gardant la même performance de reconnaissance.
- Approximated Similarity Search (ASS) : rechercher des prototypes suffisamment similaires à une requête donnée dans l'ensemble d'apprentissage, au lieu de récupérer l'instance exacte la plus proche, au prix d'une légère diminution de la précision de la classification.

Ce projet présente une méthode de Data Réduction (DR).Le fléau de la dimensionnalité dans le contexte KNN signifie essentiellement, que la distance euclidienne est inutile dans les grandes dimensions, car tous les vecteurs sont presque équidistants au vecteur de requête de recherche (imaginons plusieurs points se trouvant plus ou moins sur un cercle avec le point

de requête au centre, la distance entre la requête et tous les points de données dans l'espace de recherche est presque la même).

Dans ce projet, KNN a été sélectionné pour effectuer des tâches de classification. KNN est très populaire car il a une bonne performance, il utilise peu de ressources et il est relativement simple. L'objectif de cette proposition est d'exploiter les avantages de KNN ainsi réduire la dimensionnalité des données de grande dimension et les visualiser en les projetant dans un espace de faible dimension.

III.2.2 Le flow chart de l'algorithme KNN

Pour la classification, l'algorithme KNN fonctionne comme suit:

- Charger les données MNIST (prétraite).
- Diviser et étiquetez les données comme image et étiquettes d'apprentissage et de test.
- Utiliser la validation croisée pour diviser les données en données d'entrainement et de test pour entrainer le classificateur.
- Entraîner le classificateur à l'aide de l'algorithme KNN. Fournir des données d'apprentissage et des étiquettes en entrée pour entrainer le classificateur. Le KNN utilise la différence de distance entre les points réels et les points fournis pour classer le chiffre dans l'image.
- Le chiffre reconnu à l'aide de KNN est ensuite apparié avec les étiquettes de l'apprentissage fournies pour obtenir le score / la précision du classificateur formé.
- Ce classificateur formé est décapé pour être utilisé à nouveau sur les données de test.
- Les données de l'image de test sont utilisées pour prédire les étiquettes des chiffres et ils ensuite sont comparées avec les étiquettes de test fournies pour voir la précision de l'algorithme.
- La matrice de confusion est imprimée et fournit le pourcentage de précision avec lequel chaque chiffre a été reconnu.

Pour les données de grande dimension (par exemple, avec un nombre de dimensions supérieur à 10), la réduction de dimension est généralement effectuée avant d'appliquer l'algorithme KNN, afin d'éviter les effets de fléau de la dimensionnalité. La figure suivante présente le flow charte de l'algorithme KNN :

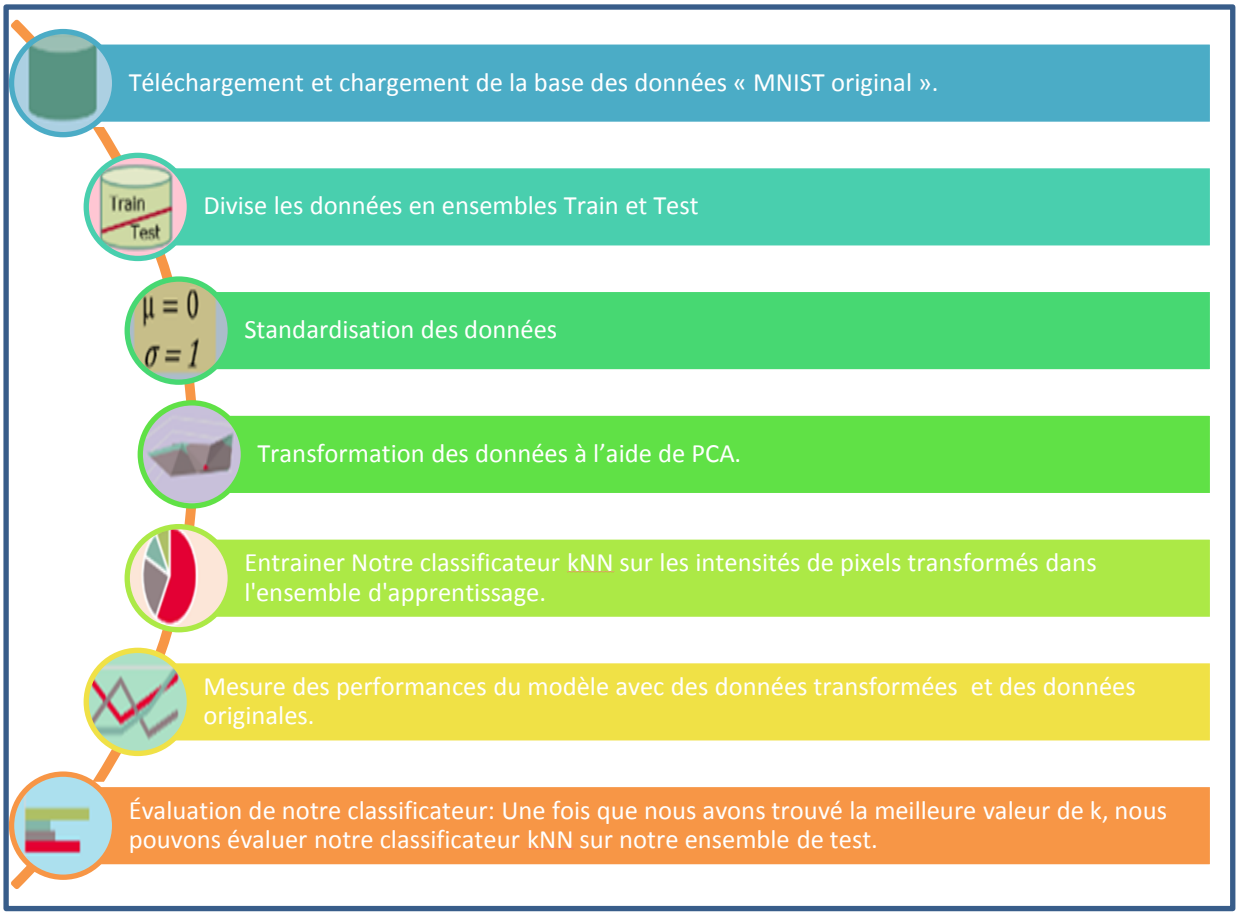


Figure 26: Le flow chart de l'algorithme KNN

III.2.3 Critère d'évaluation : Matrice de confusion

En 1998, la matrice de confusion a été proposée par Kohavi et Provost, qui contenait les informations sur les classifications réelles et prévues effectuées par un système de classification. Il se concentre sur la capacité prédictive d'un modèle plutôt que sur le temps et les vitesses que le modèle prend pour terminer la désignation.

La matrice de confusion est indiquée par une matrice où chaque ligne montre les exemples dans une classe prédite, tandis que chaque colonne les montre dans une classe réelle. Cet outil d'évaluation des performances peut non seulement déterminer si le modèle confond deux classes, mais également évaluer la précision globale ou moyenne du classificateur. Par conséquent, dans cette expérience, la matrice de confusion sera utilisée pour l'évaluation du modèle. [16]

Conclusion

Comme cette étude s'efforce d'améliorer la précision de reconnaissance de plus de 97% dans la reconnaissance des chiffres manuscrits, le prétraitement et l'extraction des caractéristiques ont les rôles cruciaux de cette expérience pour atteindre la plus grande précision.

Au stade d'extraction des caractéristiques de notre système. Le PCA est utilisé pour extraire la meilleure variance des données de la base MNIST, le chapitre suivant exposera le travail réel (la mise en œuvre du système et les résultats des expériences), ainsi que les effets des techniques de prétraitement d'image et la technique d'extraction des caractéristiques impliquant PCA sur la précision.

Chapitre IV: : Implémentation et résultats

Introduction

Dans ce chapitre, nous avons présenté les résultats obtenus pour la validation du système reconnaissance des chiffres manuscrite mais tous d'abord on va présenter une description de la base de données utilisée, ensuite plusieurs expériences sont effectués pour l'étude de classification des images avec les caractéristiques transformées avec les méthodes de réduction de dimensionnalité sur les résultats obtenus. Ces résultats permettent de mesurer les performances obtenues lors de l'utilisation de ces méthodes.

IV.1 Les outils utilisés

IV.1.1 Anaconda, la distribution open Source

Anaconda est une distribution libre et open source dédiée à la programmation Python et R. elle est très utilisée dans la science des données, Machine Learning et l'intelligence artificielle.

Cette distribution est devenue indispensable pour n'importe quel développeur dans le domaine de la data science.

Anaconda propose une variété d'outils de collecte et transformation de données à grande échelle. Elle est aussi connue pour sa richesse en modules et librairies de la data science

Plusieurs applications sont disponibles sur Anaconda Navigator :

- JupyterLab
- JupyterNotebook
- Spyder
- Pycharm
- VSCode
- Orane 3 APP
- RStudio
- Anaconda powerShell

IV.1.2 Jupyter Notebook

Jupyter Notebook est une plateforme web open-source qui permet de créer et de partager des documents qui contiennent du code. Il facilite l'organisation des fichiers, modules ainsi que la présentation du travail, sans oublier qu'il est très facile à déployer. Pour lancer Jupyter Notebook, deux manières sont possibles :

- Taper sur votre barre de recherche « Anaconda Navigator» puis lancer directement l'application
- Taper sur votre barre de recherche « **Anaconda prompt**» puis taper **« jupyter notebook »** dans la console comme sur l'image suivante. [20]

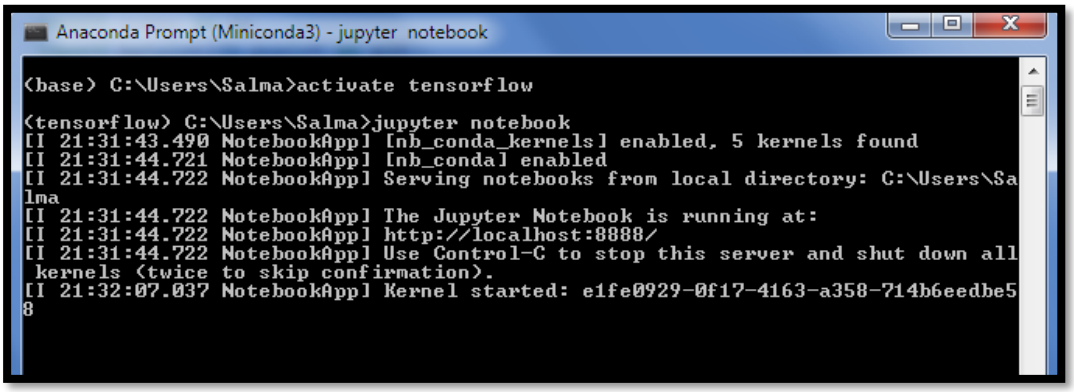


Figure 27: Activation l'environnement de travail Jupyter Notebook

IV.1.3 Bibliothèques

Nous commençons par importer nos bibliothèques et notre jeu de données. La première bibliothèque que nous importons est Tensorflow. Il s'agit d'une bibliothèque open source, développée par Google et publiée en 2015, qui est très populaire dans le Machine Learning e. Dans Tensorflow, nous importons Keras, qui est une interface de programmation d'application.

```
from tensorflow.keras.datasets import mnist
from time import time
from sklearn import metrics
from sklearn.metrics import confusion_matrix
from sklearn.preprocessing import StandardScaler
from sklearn.decomposition import PCA
from sklearn.pipeline import Pipeline
from sklearn.manifold import TSNE
from sklearn.neighbors import KNeighborsClassifier
import matplotlib.pyplot as plt
import numpy as np
```

Figure 28 : Importation des bibliothèques.

IV.2 Les résultats

IV.2.1 Lecture de MNIST

Bien que nous puissions trouver l'ensemble de données MNIST sur le site officiel de Yann LeCun, nous avons choisi un moyen plus pratique de trouver l'ensemble de données à partir de Keras.On a 60000 images dans le jeu d'apprentissage. Les images sont carrées (28*28)pixels et dans la phase de test on 10000 images. Ce qui est bien avec Keras : 70 000 images sont lues images sont réparties en training set et test set les labels sont lues.

Cependant, lors de la lecture directement à partir du Keras, il n'y a pas d'informations d'en-tête dans les 16 premiers octets par rapport à la lecture à partir du gzip, c'est pour cela que nous avons choisi cette méthode car c'est beaucoup plus rapide.

```
# Load mnist dataset
(x_train, y_train), (x_test, y_test) = mnist.load_data()
```

Figure 29 :Chargement de MNIST

IV.2.2 Prétraitement des données

Pour faciliter la phase d'apprentissage, une linéarisation est effectué en transformant chaque image de 28 × 28 en un tableau de taille 1 × 784, afin que le jeu de données d'apprentissage et de test soient convertis en vecteurs bidimensionnels de taille 60000 × 784 et 10000 × 784, respectivement.

```
print(x_train.shape, y_train.shape)
print(x_test.shape, y_test.shape)
```

```
(60000, 28, 28) (60000,)
(10000, 28, 28) (10000,)
```

Figure 30 : La forme de jeu d'apprentissage et de jeu de test avant la linéarisation

```
x_train = np.reshape(x_train, (len(x_train), -1))/255
x_test = np.reshape(x_test, (len(x_test), -1))/255

print(x_train.shape, y_train.shape)
print(x_test.shape, y_test.shape)
```

```
(60000, 784) (60000,)
(10000, 784) (10000,)
```

Figure 31 : La forme de jeu d'apprentissage et de jeu de test après la linéarisation

Affichons la premiere image :

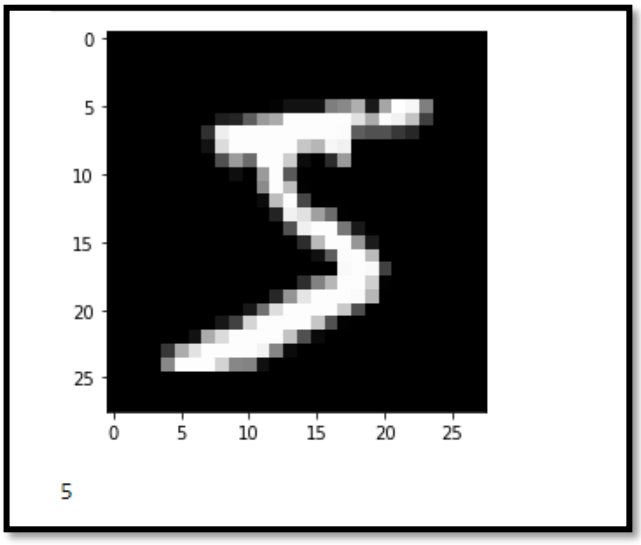


Figure 32: Affichage de la première image 5.

Maintenant, effectuons la classification et voyons les résultats. Nous utiliserons 3 algorithmes différents

- Le k plus proche voisin.
- Les réseaux bayésiens.
- La régression linéaire.

```
from sklearn import neighbors, linear_model, naive_bayes

knn = neighbors.KNeighborsClassifier()
nb = naive_bayes.GaussianNB()
logistic = linear_model.LogisticRegression(solver='liblinear', max_iter=1000, multi_class='ovr', verbose=0)

print('KNN score: %f' % knn.fit(X_train, y_train).score(X_test, y_test))
print('Naive Bayes score: %f' % nb.fit(X_train, y_train).score(X_test, y_test))
print('LogisticRegression score: %f' % logistic.fit(X_train, y_train).score(X_test, y_test))
```

```
KNN score: 0.936364
Naive Bayes score: 0.633838
LogisticRegression score: 0.834848
```

Figure 33 : Les scores des différentes techniques de classification

Nous avons obtenu une grande précision en utilisant KNN. Cependant, la résolution de 28x28 pixels est très élevée avec une taille de 784. Mais avons-nous vraiment besoin de toutes les 784 caractéristiques.

Peut-être que nous pouvons faire la même chose sinon mieux en utilisant moins de caractéristiques. Nous pouvons essayer de redimensionner l'image.

Nous allons voir comment nous pouvons changer la résolution des images numériques.

IV.2.3 Réduction de dimension avec PCA

Dans ce cas, les pixels de l'image seront les caractéristiques utilisées pour construire notre modèle prédictif. De cette manière, la mise en œuvre du KNN consiste à calculer les normes dans un espace de 784 dimensions.

Cependant, le fait calculer les normes dans cet espace de 784 dimensions est loin d'être simple et efficace. Intuitivement, nous pouvons effectuer une réduction de dimension avant d'aller à KNN et calculer ces normes, afin de devenir plus efficace.

La manière de réduire les dimensions ici est l'PCA mentionnée dans le chapitre précédent. Je ne creuse pas profondément dans PCA ici, et nous utilisons plutôt les API de sklearn pour implémenter PCA. Nous réduisons l'espace des caractéristiques de 784 dimensions à 100 dimensions.

Nous établissons un pipeline où nous procédons d'abord à l'échelle, puis nous appliquons la PCA. Il est toujours important de mettre à l'échelle les données avant d'appliquer la PCA.

Le paramètre n_components de la classe PCA peut être défini de deux façons :

- le nombre de composants principaux lorsque n_components > 1
- le nombre de composantes qui explique ce pourcentage de la variance totale des données, lorsque 0 <n_components <1.

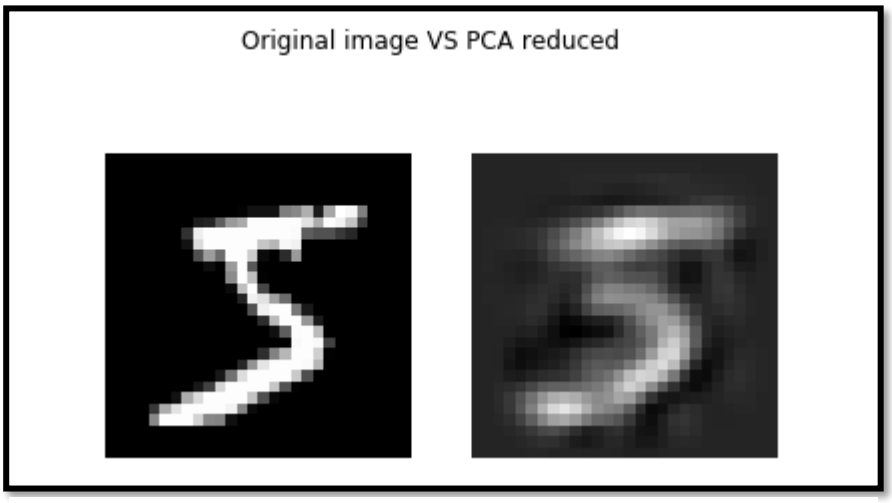


Figure 34 : Affichage de nombre 5 avec n_components=0.50

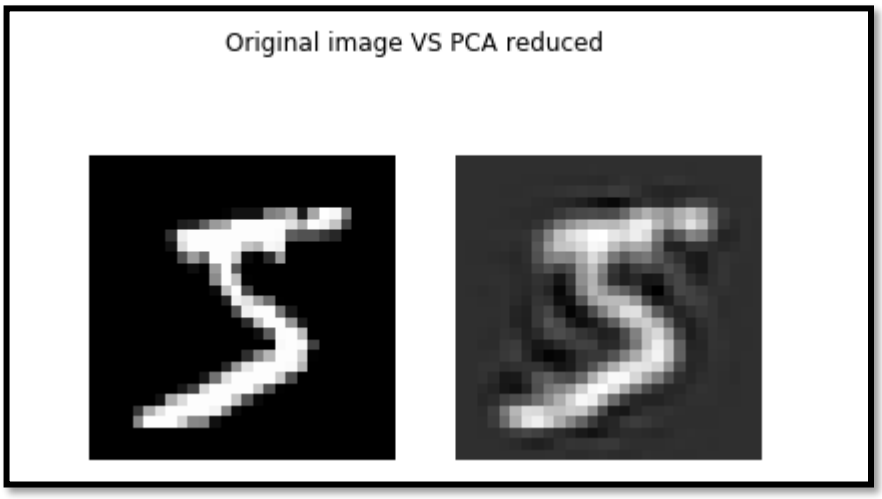


Figure 35 :Le chiffre 5 avec n _components=0.85

Dans la figure suivante, le pourcentage de variance des données expliqué par chaque composante par l'ensemble des 200 composantes

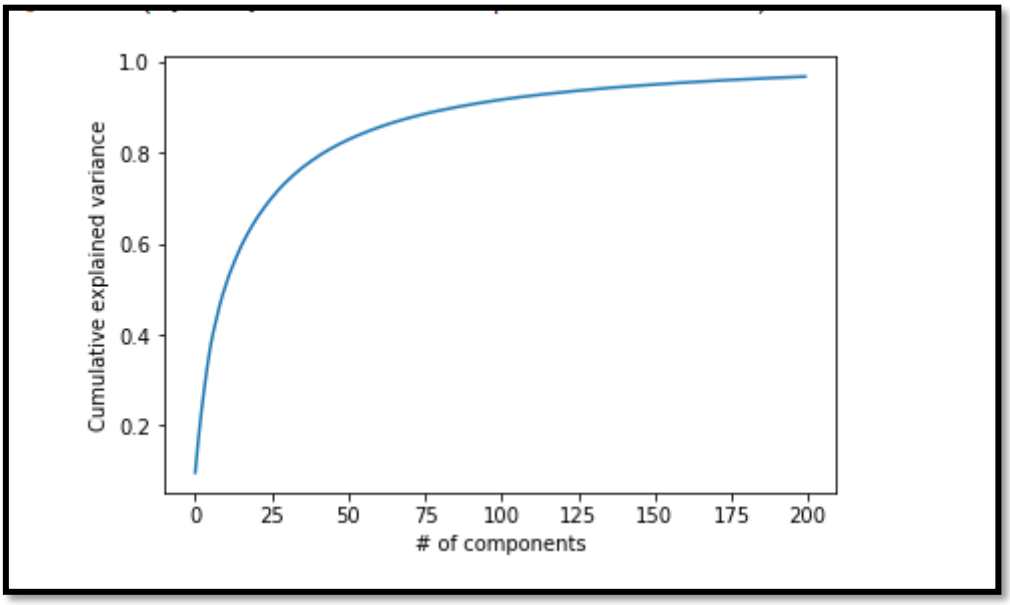


Figure 36: Pourcentage de variance des données expliqué par chaque composante

Dans notre cas avec n_components=100, nous sélectionnons le nombre de composantes principales qui représentent 91.4% de la variance.

D'après le résultat ci-dessus, nous pouvons savoir que l'ensemble de données d'apprentissage et de test deviennent deux vecteurs de taille 60000 × 100 et 10000 × 100, respectivement. A ce stade, les ensembles de données sont prêts.

IV.2.4 Entrainement et Classification

Comparons maintenant la classification des images en utilisant le nombre original de caractéristiques aux caractéristiques réduites en utilisant PCA. Nous allons comparer le temps qu'il faut pour exécuter les deux modèles, ainsi que la différence dans le score de précision.

- Entrainement sur toutes les caractéristiques :

Tout d'abord, nous entraînons une machine k plus proche voisin (KNN) avec tous les 784 pixels des images MNIST.

```
accuracy: 0.9443

[[ 963    0    1    3    1    5    6    1    0    0]
 [   0 1129    3    0    0    0    3    0    0    0]
 [  14    6  960   20    5    0    7    9   10    1]
 [   0    3    5  962    3   13    0   10   10    4]
 [   1   10    5    3  922    3    6    4    2   26]
 [   5    1    3   23    8  824   13    2    6    7]
 [  10    4    2    1    3    6  929    0    3    0]
 [   0   21   12    4    8    2    0  949    1   31]
 [  13    3    6   18    8   30    3    6  880    7]
 [   6    5    5   10   18    6    0   31    3  925]]
Training and classification done in 975.2628877162933s
```

Figure 37 : La matrice de confusion de modèle KNN sur 784 dimensions

Nous avons entrainé notre modèle et nous avons obtenu une précision de94 .44 %en environ 975.26 secondes,

- Entrainement sur les caractéristiques réduites (exemple de 50 dimensions)

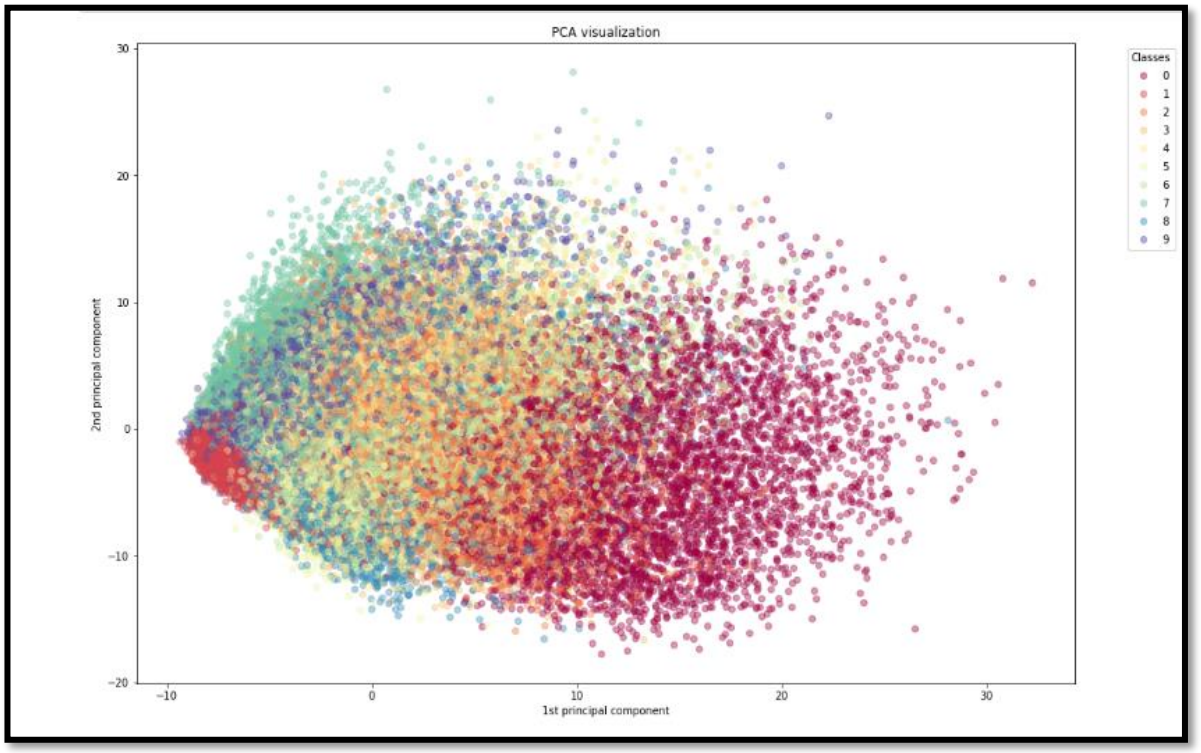
```
accuracy: 0.9579

[[ 968    0    1    1    2    2    5    1    0    0]
 [   0 1128    3    0    0    1    3    0    0    0]
 [   8    2  988    5    4    1    4    5   14    1]
 [   0    1    7  962    0   11    1    9   14    5]
 [   1    1    4    1  938    0    5    3    4   25]
 [   4    1    2   18    3  840   11    1    5    7]
 [   6    3    2    0    2    8  935    0    2    0]
 [   1   13    9    3    8    0    0  964    2   28]
 [   7    0    6   13    5   16    4    5  912    6]
 [   4    6    4   12   18    4    0   12    5  944]]
Training and classification done in 75.64980483055115s
Speedup 12.408132037173766x
```

Figure 38:La matrice de confusion sur 50 dimensions

Nous obtenons une accélération> 10x lors du prétraitement avec PCA et un score de précision assez comparable à celui de l'ensemble de données. Dans notre cas, KNN est un algorithme qui peut gérer la haute dimensionnalité, il n'y a donc une différence en termes de précision du modèle entre l'utilisation de la pleine et de l'ensemble de données réduits.

- Visualisation des donnés MNIST avec PCA sur 2D :



Nous avons atteint une précision encore meilleure en utilisant seulement 100 caractéristiques. Ceci est important car nous pouvons réduire considérablement le coût de calcul de notre classification et on peut essayer des algorithmes complexes qui prennent beaucoup de temps à s'exécuter.

Maintenant, la prochaine étape consiste à utiliser l'ensemble de données pour créer et tester notre modèle de classification final. Nous allons utiliser le classificateur KNN. Tout d'abord, en utilisant notre petit ensemble d'échantillons, décidons quelle valeur k fonctionne le mieux.

```
for k in range(1,21):
    knn = neighbors.KNeighborsClassifier(n_neighbors=k)
    acc = knn.fit(X_train, y_train).score(X_test, y_test)
    print(acc)
    plt.scatter(k, acc, c='blue')
```

```
0.945959595959596
0.9444444444444444
0.9489898989898989
0.946969696969697
0.9505050505050505
0.9494949494949495
0.9464646464646465
0.9404040404040404
0.9388888888888889
0.9393939393939394
0.9363636363636364
0.9363636363636364
0.9368686868686869
0.9378787878787879
0.9323232323232323
0.9318181818181818
0.9282828282828283
0.9282828282828283
0.9262626262626262
0.9262626262626262
```

Figure 39: Les meilleurs scores de KNN avec différents k.

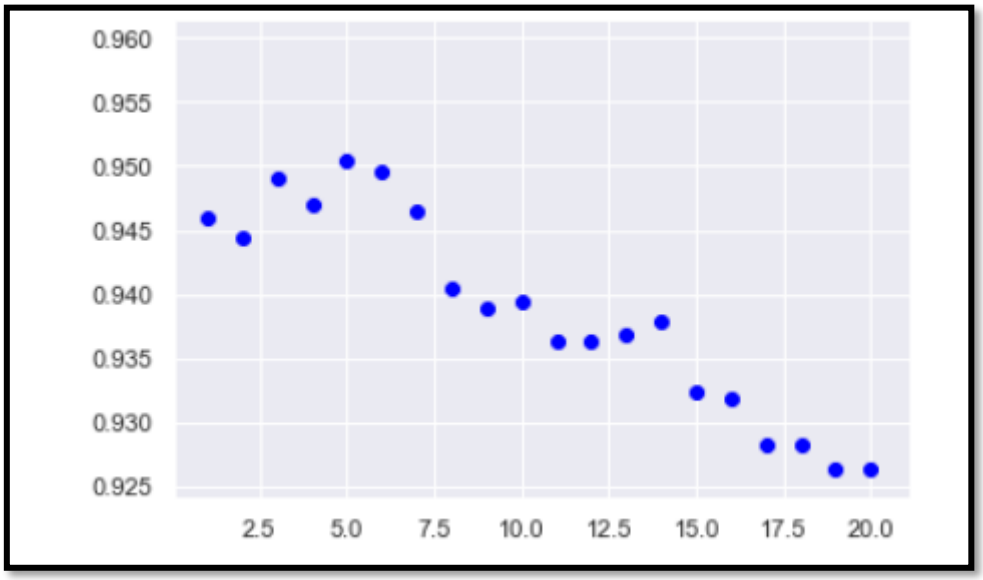


Figure 40: Les scores des différents k.

Nous concluons que k=5 fonctionne légèrement mieux.

```
print('Test accuracy: ', accuracy_score(y_test, predicted))

Test accuracy:  0.9676

print(confusion_matrix(y_test, predicted))

[[ 973    1    1    0    0    1    3    1    0    0]
 [   0 1128    3    1    0    0    1    1    1    0]
 [   8    2  997    4    0    0    2   14    5    0]
 [   1    5    7  965    1   12    0    5   12    2]
 [   1    3    3    0  939    0    2    1    1   32]
 [   4    2    0   14    2  855    6    2    4    3]
 [   5    4    0    0    2    2  944    0    1    0]
 [   0   20    7    0    2    0    0  985    1   13]
 [   3    2    4    9    8    6    4    5  926    7]
 [   2    9    3    6   12    3    2    4    4  964]]
```

Figure 41:La matrice de confusion de 100 dimensions.

IV.2.5 Tests

Enfin, terminons cet exemple de code en examinant certaines des prédictions individuelles de notre classificateur k-NN:

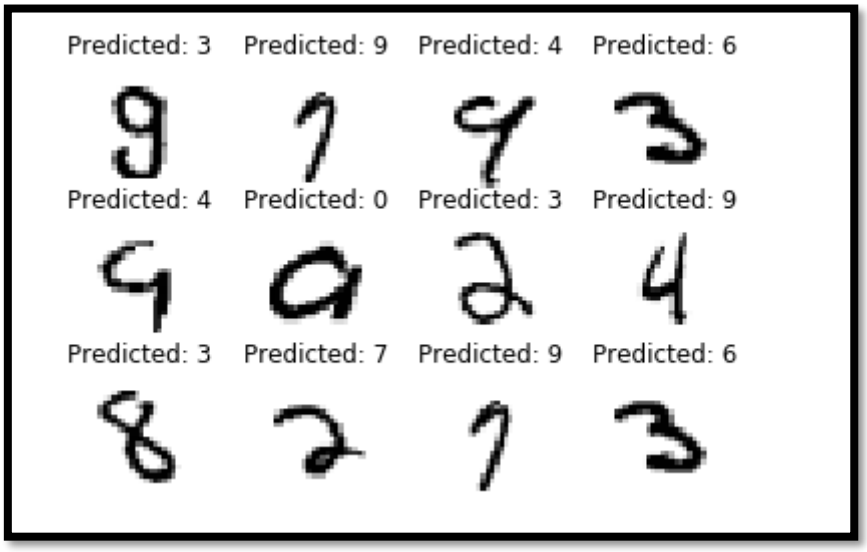


Figure 43:Les fausses prédictions

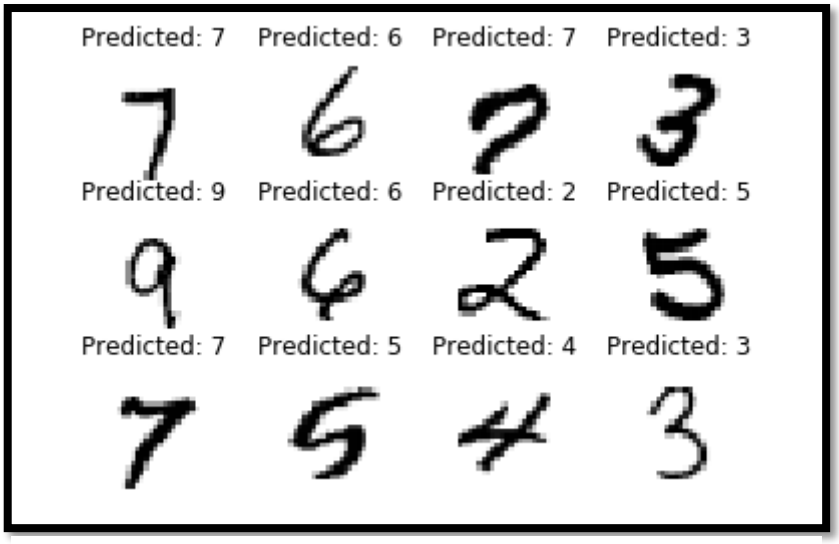


Figure 42:Les vraies prédictions

Conclusion

D'après notre modèle expérimental nous avons conclu que l'application PCA sur le jeu des données MNIST réduit considérablement le temps de calcul et l'espace requis et donne également une meilleure précision de généralisation et aide à éviter le surajustement.Une précision de 97% est obtenue sur l'ensemble de données de test de 100 dimensions.

Conclusion et perspectives

L'objectif de ce travail est la mise en œuvre et l'évaluation de méthodes d'extraction de caractéristiques pour la reconnaissance des chiffres manuscrits en utilisant le classifieur KNN.

Le travail présenté dans ce mémoire décrit les différentes étapes nécessaires à la construction d'un système de reconnaissance de chiffres manuscrits isolés, pour chacune de ces étapes à savoir : l'extraction des caractéristiques et la classification.

Dans le domaine de la reconnaissance de chiffres manuscrits, les caractéristiques peuvent être décrites comme un moyen permettant de distinguer un caractère d'une classe d'un autre caractère d'une autre classe. Dès lors, il est nécessaire de définir des caractéristiques significatives lors du développement d'un système de reconnaissance. Nous avons présenté plusieurs méthodes d'extraction des caractéristiques pertinentes dans la littérature.

Notre principale contribution était d'améliorer le système par l'utilisation le module de l'extraction de caractéristiques. Nous avons proposé, au cours de ce projet un système de reconnaissance de chiffres qui a permis de réduire la dimension de la base des données MNIST d'images de chiffres écrits à la main (0-9) et extraire les informations pertinentes. Afin de gérer la complexité des ensembles de données. Nous avons choisi la PCA comme technique de réduction de dimensionnalité dans l'étape d'extraction des caractéristiques. Les vecteurs de dimension inférieurs obtenus sont ensuite utilisés pour classer les chiffres numériques en utilisant la méthode de classification KNN. Note modèle expérimental nous a permis d'atteindre la plus grande précision de 97% sur l'ensemble des données MNIST avec 100 dimensions.

Avant d'aborder les nouvelles perspectives, ce système est considéré comme une étape préliminaire pour réaliser un système de reconnaissance des chiffres isolés puissant et performant. Au-delà un certain nombre d'amélioration peut être envisagé :

- Essayer d'agrandir le nombre des échantillons de bases de données pour tester l'efficacité de l'approche proposée.
- Améliorer le système par combinaison des plusieurs méthodes de classification.
- Tester d'autres méthodes de sélection de caractéristiques pour une meilleure caractérisation des chiffres.

.

Bibliographie

[1] P. Y. Nidhika Yadav, «Handwriting Recognition System- A Review,» *International Journal of Computer Applications* , vol. 114, p. (0975 – 8887), 2015.

[2] [En ligne]. Available: https://fr.wikipedia.org/wiki/Reconnaissance_de_l%27%C3%A9criture_manuscrite.

[3] H. T. a. G. W. B. Du~R~ W. HAETTICH, «A COMBINATION OF STATISTICAL AND SYNTACTICAL PATTERN RECOGNITION APPLIED,» *Pergamon Press Ltd. 1980. Printed in Great Britain,* vol. 12, pp. 189-199.

[4] Ming Wu Zhen Zhang, «Handwritten Digit Classification using the MNIST Data Set,» *ResearchGate publications,* 2010.

[5] Z. S. MENASRIA Abdelaali, «Reconnaissance hors ligne des chiffres manuscrite isolé (Base de donnée M.N.I.S.T),» *MEMOIRE DE MASTER Domaine: Mathématiques et Informatique,* 2015/2016.

[6] ⇑. R. R. a. P. D. a. C. P. b. G. S. K. a. D. Prabha Devi a, «Design and simulation of handwritten recognition system,» *Materials Today: Proceedings,* 2019.

[7] Z. ismaili, «datascientist.fr,» 28 janvier 2019. [En ligne]. Available: https://le-datascientist.fr/apprentissage-supervise-vs-non-supervise.

[8] A. M. A. D. Q. Z. D. A. Z. J. N. S. Rizgar R. Zebari, «A Comprehensive Review of Dimensionality Reduction Techniques for Feature Selection and Feature Extraction,» *JOURNAL OF APPLIED SCIENCE AND TECHNOLOGY TRENDS,* vol. 01, n° %12708-0757, p. 56 –70, 2020.

[9] F. C. J. R. M. J. d. j. Francisco J. Pulgar1, «AEkNN: An AutoEncoder kNN-based classifier with,» 2018.

[10] H. CHOUAIB, «séléction de caractéristiques.».

[11] A. D. Anuj Dutt, « 'Handwritten Digit Recognition Using Deep Learning',» *IJARCET,* vol. 6, n° %12278 – 1323, pp. 990-997, 2017.

[12] M. R. H. J. &. T. D. F. Phangtriastu, «Comparison Between Neural Network and Support Vector Machine in Optical Character Recognition.,» *Procedia Computer,* 2017.

[13] T. A. B. W. K. N. R. Nurul Ilmi, «Handwriting Digit Recognition using Local Binary Pattern Variance and K-Nearest Neighbor Classification,» *Fourth International Conference on Information and Communication Technologies (ICoICT),* 2016.

[14] S. A. S. &. H. L. Bernard, « Using Random Forests for Handwritten Digit Recognition.,» *Ninth International Conference on Document Analysis and Recognition,* Vols. %1 sur %22,1043-1047, 2007.

[15] A. A. A. A. Yogish Naik G R, «PCA Based English Handwritten Digit Recognition,» *International Journal of Advanced Research in Computer Science,* vol. 8, n° %10976-5697, p. 5, 2017.

[16] K. Zhao, «Handwritten Digit Recognition and Classification,» [En ligne].

[17] [En ligne]. Available: http://yann.lecun.com/exdb/mnist/.

[18] S. C. A. G. L. wells, «Offline Handwritten Digits Recognition Using Machine learning,»

Conference: Proceedings of the International Conference on Industrial Engineering and Operations Management Washington DC, USA, vol. 133, 2018.

[19] [En ligne]. Available: https://connect.ed-diamond.com/GNU-Linux-Magazine/GLMFHS-102/Retour-d-experience-sur-l-etude-de-la-base-MNIST-pour-la-reconnaissance-de-chiffres-manuscrits.

[20] [En ligne]. Available: https://www.cours-gratuit.com/tutoriel-python/tutoriel-python-comment-programmer-en-python-avec-anaconda.

[21] O. R. M. M. Hanmandlu, «Fuzzy model based recognition of handwritten numerals,» *Pattern Recognition,* vol. 40, p. 1840 – 1854, 2007.

[22] [En ligne]. Available: https://openclassrooms.com/fr/courses/4379436-explorez-vos-donnees-avec-des-algorithmes-non-supervises/4379443-comprenez-pourquoi-reduire-la-dimension-de-vos-donnees.

[23] [En ligne]. Available: https://fr.wikipedia.org/wiki/R%C3%A9duction_de_la_dimensionnalit%C3%A9.

[24] [En ligne]. Available: https://www.datacamp.com/community/tutorials/introduction-t-sne.

[25] [En ligne]. Available: https://colah.github.io/posts/2014-10-Visualizing-MNIST/.

[26] [En ligne]. Available: https://github.com/akhilgadi/Dimension-Reduction-on-MNIST.

[27] Y. Benzaki, 8 October 2018. [En ligne]. Available: https://mrmint.fr/introduction-k-nearest-neighbors.

[28] [En ligne]. Available: https://towardsdatascience.com/feature-extraction-using-principal-component-analysis-a-simplified-visual-demo-e5592ced100a.

[29] 2. 3. W. 1RAHUL R. TIWARI, «HANDWRITTEN DIGIT RECOGNITION USING BACK PROPAGATION NEURAL NETWORK& K-NEAREST NEIGHBOUR CLASSIFIER,» *International Journal of Electrical, Electronics and Data Communication,* vol. 1, pp. 2320-2084.

[30] [En ligne]. Available: https://medium.com/@redouanechafi/data-science-0-0-quest-ce-que-le-machine-learning-fde2b3c5f19f.