People's Democratic Republic of Algeria

Ministry of Higher Education and Scientific Research

University Larbi Tébessi – Tébessa

Faculty of Exact Sciences and Natural and Life Sciences

Department: Mathematics and Computer Science

Final thesis

To obtain a MASTER diploma

Domain: Mathematics and Computer Science

Field: Computer Science

Specialty: Information Systems

Handcrafted and deep learning-based features for handwritten digit recognition.

Presented by:

Boutra Intissar & Djabri Med Nadhir

In front of the jury:

Dr. Chergui.O	Larbi Tebessi University	President
Dr. Menassel.Y	LarbiTebessi University	Thesis Supervisor
Dr. Khedhiri.S	Larbi Tebessi University	Examiner

College year:

2023/2024

شكر و عرفان

الحمدُ لله على البدء والختام، أُهدي تخرجي هذا إلى نفسي الطموحة، وإلى عائلتي إلى رفيقات الدرب وزميلي في المذكرة.

كما أُهديه إلى زملائي في فلسطين، الذين منعتهم الحرب من أن يكونوا في مكاني. أُهديه أيضًا إلى يوسف، الذي بقي عالقًا في ذاكرتنا، 'عمره 7 سنوات، شعره كيرلي، أبيضاني وحلو'. وأُهديه إلى من قال له أبوه في لحظة استشهاده: 'يا با مش كان بدك تطلع صحفي'. الذين جاهدوا وقاتلوا وأُسروا في سبيل الله.

بوترعة إنتصار

شكر و عرفان

في البداية أشكر الله عز وجل الذي منحني الحياة والصحة والعافية والشجاعة لإكمال هذا العمل و لجعلي أعيش لحظة الحلم هذه يسعدني أن أهدي هذه السطور القليلة تعبيراً عن الامتنان والتقدير العميق لعائلتي التي وقفة بجانبي و سندا لي خلال مرضي ولا يمكن لأي تفاني أن يعبر عما ندين لهم به من إحسانهم ومودتهم ودعمهم. كنوز اللطف والكرم والحنان، شهادة على محبتنا العميقة وامتناننا الكبير.كما أُهديه إلى زميلتي و لجميع أصدقائنا لمساعدتهم ودعمهم المعنوي أثناء تطوير العمل الدراسي النهائي.

جابري محمد نذير

Thanks:

It is with great pleasure that we dedicate these few lines as a sign of gratitude and deep appreciation to all those who, near or far, have contributed to the realization and outcome of this work. First, we thank Allah Almighty for allowing us to carry out this thesis and for guiding us on the path of knowledge. We would then like to extend our warmest thanks to Mr.Mennasel.Yahya for his invaluable supervision, advice, patience, and the confidence he showed inus, which were decisive in the completion of our work. Our thanks also go to the members of the jury, Dr.Chergui.Othaila and Dr.Khedhiri.Samir, for the honor they bestowed upon us by agreeing to judge our work. We thank all the teachers that we have been able to meet and benefit from. We would like to thank our colleagues in the Mathematics and Computer Science department and all those who have helped us directly or indirectly in the realization and accomplishment of this work.

Abstract

Handwritten digit recognition is a classic problem in the field of pattern recognition and machine learning. The goal is to develop algorithms that can accurately identify and classify digits from handwritten images. Handcrafted and deep learning-based features are two approaches used to extract relevant information from these images for the recognition task.

Handcrafted features refer to manually designed and selected characteristics or attributes of the input data that are believed to be relevant for the recognition task. In the context of handwritten digit recognition, handcrafted features can include measures such as the aspect ratio of the digits, the number of loops, the curvature of specific strokes, and other domain-specific characteristics.

Both approaches have been explored for handwritten digit recognition,. Traditional machine learning methods often rely on handcrafted features, while deep learning methods, especially CNNs, have shown remarkable success in learning features directly from raw pixel values, eliminating the need for explicit feature engineering.

Keywords: Deep learning, CNN, handcrafted features, Machine learning, .

Résumé

La reconnaissance des chiffres manuscrits est un problème classique dans le domaine de la reconnaissance de formes et de l'apprentissage automatique. L'objectif est de développer des algorithmes capables d'identifier et de classifier avec précision les chiffres à partir d'images manuscrites. Les caractéristiques conçues à la main et les caractéristiques basées sur l'apprentissage profond sont deux approches utilisées pour extraire des informations pertinentes de ces images pour la tâche de reconnaissance.

Les caractéristiques conçues à la main font référence à des caractéristiques ou attributs du jeu de données d'entrée qui sont manuellement conçus et sélectionnés et qui sont considérés comme pertinents pour la tâche de reconnaissance. Dans le contexte de la reconnaissance des chiffres manuscrits, les caractéristiques conçues à la main pourraient inclure des mesures telles que le rapport d'aspect des chiffres, le nombre de boucles, la courbure de certaines lignes, et d'autres caractéristiques spécifiques au domaine.

Dans le contexte de la reconnaissance des chiffres manuscrits, les deux approches ont été explorées. Les méthodes traditionnelles d'apprentissage automatique ont souvent utilisé des caractéristiques conçues à la main, tandis que les méthodes d'apprentissage profond, en particulier les CNN, ont montré un succès remarquable en apprenant directement les caractéristiques à partir des valeurs de pixels brutes, éliminant ainsi le besoin d'ingénierie de caractéristiques explicites.

Mots-clés : Apprentissage profond, CNN, caractéristiques conçues à la main, apprentissage automatique.

ملخص

تعتبر التعرف على الأرقام المكتوبة يدوياً مشكلة كلاسيكية في مجال التعرف على الأنماط والتعلم الآلي. الهدف هو تطوير خوارزميات يمكنها تحديد وتصنيف الأرقام بدقة من الصور المكتوبة يدوياً. السمات المصنوعة يدوياً والسمات المستندة إلى التعلم العميق هما اثنتان من النهج المستخدمة لاستخراج المعلومات ذات الصلة من هذه الصور لمهمة التعرف. تشير السمات المصنوعة يدوياً إلى الخصائص أو السمات المصممة و المختارة يدوياً من بيانات الإدخال و التي يعتقد أنها ذات صلة لمهمة التعرف. في سياق التعرف على الأرقام المكتوبة يدوياً،يمكن أن تشمل السمات المصنوعة يدوياً مقاييس مثل نسبة الجانب للأرقام،وعدد الحلقات،وانحناء الخطوط المحددة،وسمات أخرى متخصصة في المجال. في سياق التعرف على الأرقام المكتوبة يدوياً،تم استكشاف كلا النهجين . غالباً ما اعتمدت الطرق التقليدية للتعلم الآلي على السمات المصنوعة يدوياً،بينما أظهرت الطرق العميقة للتعلم ،وخاصة الشبكات العصبية المتعمقة ،نجاحاً ملحوظاً في تعلم السمات مباشرة من قيم البكسل الخام،مما يقضي على الحاجة لهندسة السمات الصريحة.

كلمات رئيسية:التعلم العميق،CNN،السمات المصنوعة يدوياً،التعلم الآلي.

Content

Figure list

Table list

General Introduction

The handwritten digit recognition system represents an important milestone in the field of pattern recognition. This document describes the different phases involved in recognizing isolated handwritten digits, providing insight into the handwritten digit recognition process.

It covers the preprocessing phase, which serves to reduce noise, correct defects, and preserve essential information through techniques such as noise elimination, binarization, skeletonization, and size normalization.

The document discusses machine learning-based approaches, including handcrafted features such as geometric, statistical, structural, and contour-based features, as well as handcrafted classification methods such as support vector machines, k-nearest neighbors, and random forests.

It also explains deep learning-based approaches, where feature extraction and classification are performed simultaneously within neural network architectures such as convolutional neural networks (CNNs). Notable CNN architectures such as VGGNet, ResNet, and MobileNet have been described.

Furthermore, the document reviews subsequent works on handwritten digit recognition, covering studies from various authors that employed different recognition techniques, and datasets, and achieved results. It provides a summary table consolidating the key findings and contributions of each cited study.

Additionally, the experimental results obtained from developing a novel hybrid approach that combines the VGG-16 CNN for feature extraction and a support vector machine (SVM) classifier for handwritten digit recognition, were evaluated on the MNIST, ARDIS, and CVL datasets.

Overall, the document offers a comprehensive overview of the handwritten digit recognition system, covering the essential phases, approaches, architectures, and experimental results, positioning it as a foundational work in the field of pattern recognition.

Chapter I:

The Handwritten Digit Recognition System

I.1.Introduction :

The Isolated Handwritten Digit Recognition System (IHDRS) represents a foundational milestone within the realm of pattern recognition. In this chapter, we delineate the distinct phases involved in the recognition of isolated handwritten digits, providing comprehensive insight into the intricacies of the handwritten digit recognition process

I.2.Handwritten Digit Recognition :

The task of handwritten digit recognition involves automatically identifying and categorizing handwritten digits, constituting a significant challenge in the fields of computer vision and machine learning. Notably, advancements in deep learning techniques, particularly convolutional neural networks (CNNs), have substantially enhanced the accuracy of this process.

Handwriting recognition systems are typically classified into two main categories: online and offline recognition methods.

I.2.1.Online Handwriting Recognition :

Online handwriting recognition involves the real-time conversion of a user's handwritten input on digital devices such as tablets or smartphones. This method requires immediate recognition and interpretation of handwritten strokes as they are drawn, presenting challenges in real-time processing and interpretation.[1].

I.2.2.Offline Handwriting Recognition :

In contrast, offline handwriting recognition is a relatively simple process. It entails converting scanned images or digital representations of handwritten text into computer-readable formats. When operating on pre-existing images or documents, this method does not necessitate real-time processing. Handwritten content is analyzed and translated into text using machine learning algorithms [1].

I.3.The Handwritten Digit Recognition System :

The handwritten digit recognition system (HDRS) has become an interesting field that has attracted researchers in recent decades. A recognition system based on machine learning techniques is composed of five important phases, see figure **I.1**.

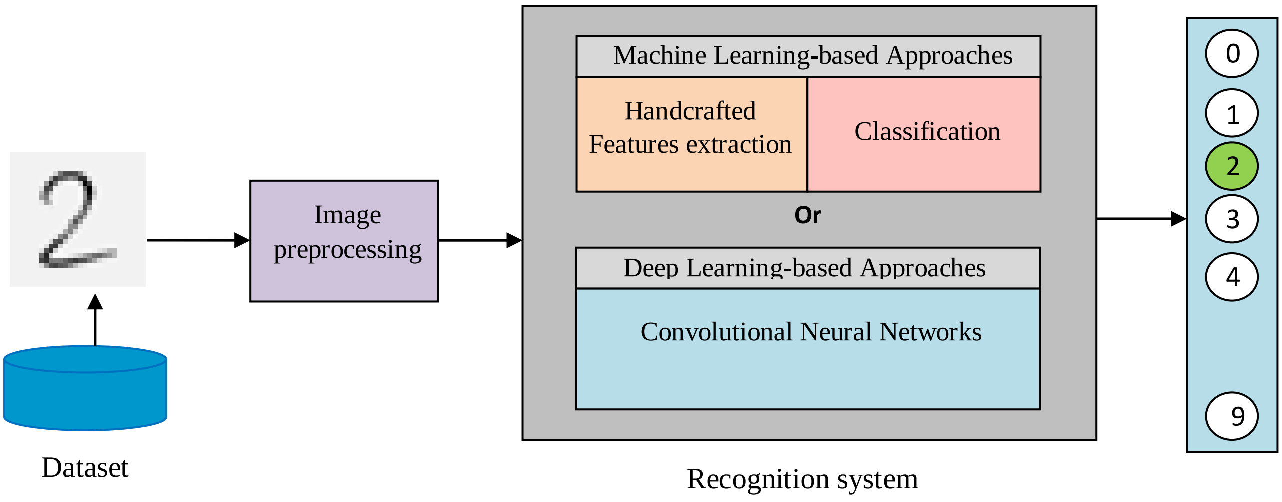


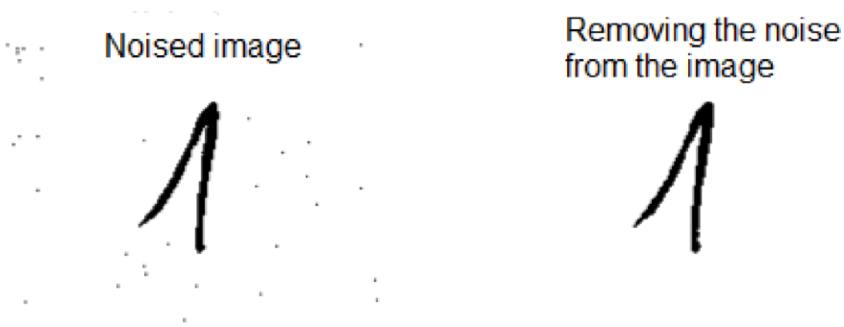
Fig I.1. General diagram of an HDR system.[2]

I.3.1 Preprocessing :

The preprocessing phase serves various purposes, primarily aimed at reducing noise in handwritten data, occasionally rectifying defects, and selectively preserving essential information from the original form. Its significance lies in its direct impact on recognition performance. Given the multitude of steps required for successful recognition, they can be outlined as follows:

I.3.1.1. Noise elimination :

Noise elimination encompasses techniques such as dropout, smoothing, and correction of outliers characterized by significant angular variations, as highlighted in [3]. The primary objective is to minimize noise to enhance discrimination, see figure **I.2**.



FigI.2. Example of Noise Elimination

I.3.1.2.Binarization :

The objective of image binarization is to assign a value of 1 to pixels with intensities greater than a specified threshold value S, and a value of 0 to pixels with intensities below this threshold, using a process known as thresholding.Scanned documents, being essentially images, can undergo binarization or be converted to grayscale through thresholding techniques. The determination of the threshold can either be global, where the same threshold value is applied to all pixels in the document, or local, where the threshold varies spatially.Global thresholding methods, while computationally efficient, yield satisfactory results only under uniformly illuminated conditions. On the other hand, local thresholding methods offer greater robustness against variations in illumination but require longer computational times [4] see figure **I.3**.

FigI.3. Example of binarization.

I.3.1.3.Skeletonization :

A skeleton is a concise representation formed of lines with a thickness of one, positioned at the center of the figure, homotopic to the original shape (meaning it has the same number of connected components), and retains its shapes and topology. During the skeletonization process, the image is reduced and compressed, maintaining the overall shape of the object, and identifying a median axis known as the set of pixels S equidistant from the surrounding boundary pixels[5][6].

The skeletal representation offers several advantages:

- It provides a clear depiction of the structural relationships among model components.
- Widely utilized in systems for recognizing characters, words, signatures, and fingerprints.

FigI.4. Example of skeletonization

I.3.1.4.Size normalization :

To streamline the recognition process, it is essential to standardize digits to a fixed size using sampling techniques. However, the critical challenge lies in determining the appropriate size. If the size is too small, there is a risk of losing valuable information. Conversely, if it is too large, the recognition phase may become sluggish [7] see figure **I.5**.

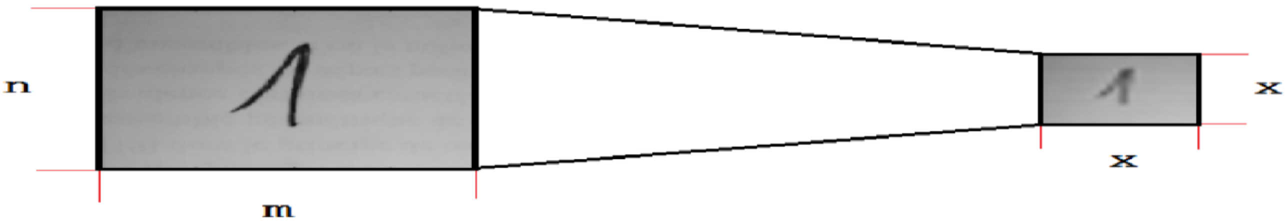


Fig I.5. Example of a normalized handwritten digit.

I.3.1.5 Zoning :

Zoning is widely recognized as one of the simplest and most popular features to implement. As illustrated in Figure 1.5, zoning involves dividing an individual image of size(n*n) into (x) equal zones or blocks, each of uniform size.

Feature extraction entails counting the number of black pixels within each zone. This process is iteratively applied to all zones, generating a signature array for each character. Consequently, for every character, we obtain a signature array of length (x), derived from the information within each zone[8] see figure **I.6**.

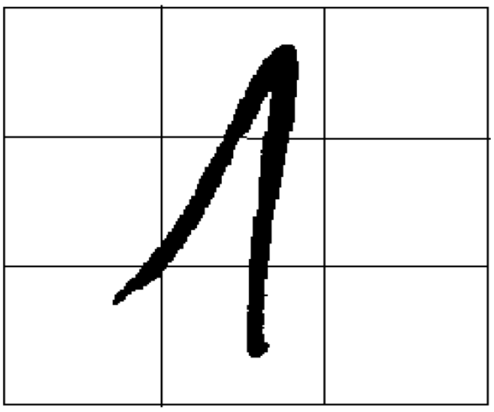


Fig I.6. Example of a zoned digit of size (3 * 3)

I.4-Machine Learning-based Approaches :

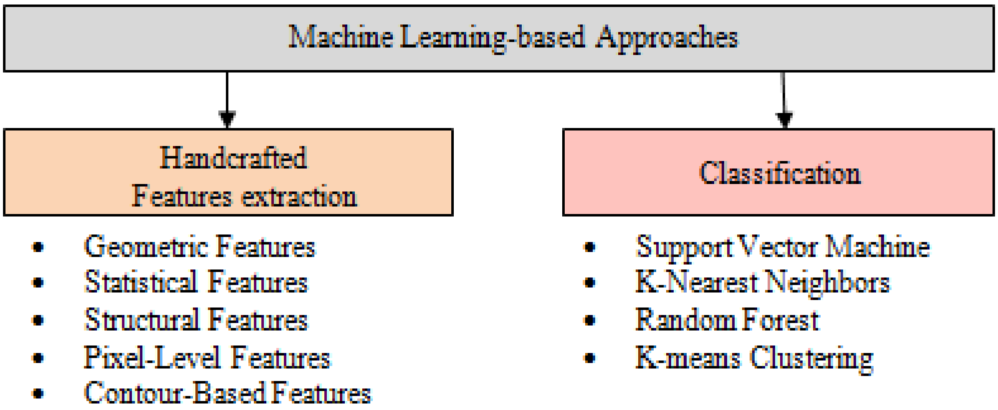


Fig.I.7. Machine Learning-based Approaches [2].

I.5-Handcrafted Features :

Handcrafted features in the context of digit recognition involve manually designing and extracting features from raw image data to aid in the recognition process. These features are carefully selected by human experts based on domain knowledge and intuition. Examples of handcrafted features for digit recognition include various categories such as geometric features, statistical features, structural features, pixel-level features, and contour-based features. [9]

I.5.1-Geometric Features Extraction Methods :

I.5.1.1- Local binary pattern (LBP) :

The Local binary pattern (LBP) is a highly efficient texture operator due to its straightforward and reliable calculation process. In essence, it operates by examining each pixel within an image, and determining a label for it by comparing its intensity to that of its neighboring pixels. This comparison results in a binary decision, denoted as either '0' or '1', indicating whether the neighbor's intensity is greater or lesser than the center pixel's. These binary labels from neighboring pixels are then combined to form a binary number. Subsequently, a histogram is constructed based on the frequency distribution of these binary numbers[10].

I.5.1.2- Histogram Of Oriented Gradient (HOG) :

The histogram of oriented gradient (HOG) feature descriptor is a technique that focuses on quantifying the frequency of gradient orientations within localized regions of an image. By doing so, it discerns whether a pixel resides on an edge or not, while also determining the gradient magnitude and direction for each pixel. This process involves computing gradients along both the horizontal and vertical axes of the image. Various filters, such as Sobel, Kirsch, and Roberts, can be employed to calculate these gradients[11] .

I.5.1.3-Contour Angular Technique(CAT) :

The Contour Angular Technique (CAT) is a system designed to capture curvature aspects and angular co-occurrence in numeral images. It operates as a rapid implementation of quantized angle co-occurrence computation, akin to the Hinge feature. In this method, the original image undergoes division into non-overlapping 4x4 blocks. Subsequently, for each block, an 8-directional code is computed to identify the contour of neighboring pixels from a starting point. Additionally, it computes an angular co-occurrence histogram comprising 64 elements, providing an approximation of the probability of angular co-occurrence along contours. By merging the outputs of both processes, a feature vector of size 192 is generated, which is then utilized as input for classifiers[12] .

I.5.2- Statistical Feature Extraction Methods :

I.5.2.1-MOMENTS :

Moments serve to determine the center of gravity of an image by considering the distribution of pixels, thereby capturing global shape information. They are capable of describing properties at a distance from a specific axis or point. Notably, moments gained popularity due to their resilience to rotation, translation, and scaling transformations. devised a feature vector comprising fifteen translation-invariant central moments. This approach attained a peak accuracy of 86.7% when utilized with a feed forward back propagation neural network [13].

I.5.2.2- CENTROIDS :

In image analysis, centroids represent the midpoint of an object. They are computed by taking the weighted average of pixel coordinates within a region of interest, typically corresponding to a digit in the image. Global centroids consider all pixels within the entire image, while local centroids are calculated for specific zones or regions within the image [14].

I.5.2.3-PIXEL DENSITY :

Pixel density is a statistical feature frequently employed in various image classification tasks. It quantifies the number of black pixels within a specified region of interest, which can encompass either the entire image or segmented zones within the image. Some techniques normalize pixel density by the size of the zone to derive a density value [14] .

I.5.3-Pixel-Based Feature Extraction Methods :

I.5.3.1-Gray Pixel-Based (GPB) :

The gray (28 × 28) pixel-based method relies on the raw pixel intensities of handwritten images to retain the original appearance while preserving subtle intensity gradients along ink traces. The handwritten image is resized to a resolution of 28 × 28 pixels, resulting in 784 feature values being computed. This approach maintains the integrity of handwritten images without compromising the details present in the intensity gradients along the edges of the ink traces [15].

I.5.3.2- Black and White Down Scaled (BWS) :

This basic feature technique is valuable for establishing a foundational performance measure. It involves dividing black and white handwritten images into non-overlapping blocks, which are typically organized in a 9×9 grid. Within each block, the number of black pixels is calculated, yielding a total of 81 features. This method provides a simple yet effective means of deriving features from the image, which can be utilized for assessing performance or conducting further analysis [15].

I.5.3.3-The oriented basic image features (oBIFs) :

oBIFs are robust texture descriptors known for their effectiveness in character recognition, texture classification, and writer identification tasks. They are an extension of basic image features (BIFs), that combine local orientation information with local symmetry information. Each pixel location in the image is assigned to one of seven local symmetry classes, such as dark line on light, light line on dark, dark rotational, light rotational, slope, saddle-like, or flat[16] .

I.6-Handcrafted Classification :

In handcrafted classification for digit recognition, human experts carefully engineer and extract relevant features from the input digit images. These handcrafted features are then fed into traditional machine learning classifiers, such as support vector machines (SVMs) and random forests…etc. The classifier is trained on the handcrafted feature vectors to learn the mapping between these features and the corresponding digit classes. The performance of handcrafted classification heavily relies heavily on the quality and discriminative power of the manually engineered features. While interpretable, as feature importance can be analyzed, the process of feature engineering and classifier optimization often involves extensive trial-and-error, making it a time-consuming and labor-intensive approach.

I.6.1-Support Vector Machine (SVM) :

SVM are powerful machine learning algorithms used for both classification and regression tasks. Originally designed for binary classification, SVMs can be extended to handle multiclass problems by using the one-vs-rest approach. The key advantages of SVMs include their ability to handle high-dimensional feature spaces efficiently using the kernel trick, robustness to noisy data and outliers due to the max-margin principle, flexibility to model complex, nonlinear decision boundaries by using appropriate kernel functions, and amenability to theoretical analysis and strong generalization performance. To apply SVMs to multiclass numeral recognition, the algorithm involves preprocessing and feature extraction, training a set of binary SVM classifiers using a linear kernel, and using the one-vs-rest approach to handle the multiclass setting. During testing, the algorithm classifies each input numeral by applying all the trained binary SVMs and assigning the class with the highest confidence score[17].

I.6.2-K-Nearest Neighbors (kNN) :

KNN is a versatile and intuitive machine learning algorithm that has found widespread applications in various domains, including image classification, pattern recognition, and data mining. The fundamental principle behind kNN is its simplicity - it classifies a new data point based on the majority vote of its k nearest neighbors in the feature space. This nonparametric approach makes no assumptions about the underlying data distribution, allowing it to handle complex and diverse datasets. The algorithm's performance is largely influenced by the choice of k, the distance metric, and the quality of the feature representation [18] [19].

I.6.3-Random forest :

RF is a machine learning ensemble technique used in classification and regression. Its algorithm is responsible for determining the output based on the assumptions of the decision trees. It makes predictions by averaging the results of several trees. As the number of trees increases, so does the precision of the outcome. Decision nodes, root nodes, and leaf nodes are the three parts of a decision tree. The decision tree approach divides a dataset into branches, which are then further divided into branches. This procedure is repeated until a leaf node is reached and cannot isolate the leaf node anymore. To forecast the outcome, the qualities represented by the nodes in the decision tree are utilized [20].

I.6.4-K-means clustering :

K-means clustering is a prominent unsupervised machine learning technique, that segregates input instances into a predefined number of clusters, denoted as (k). Initially, the algorithm selects (k) centroids utilizing the k-means++ method, ensuring their effective distribution across the data space. Subsequently, each input instance is assigned to the cluster whose centroid is closest, which is typically determined by a distance metric such as the Euclidean or Manhattan distance. The centroids are then updated by computing the average of all instances within each cluster. This iterative process continues until the centroid values stabilize. The algorithm's efficacy hinges significantly on the choice of (k), which can be determined using methodologies such as the Calinski-Harabasz score or the elbow method. K-means clustering is particularly advantageous when dealing with well-separated clusters and relatively low levels of noise in the data[21].

I.7.-Deep Learning-based Approaches :

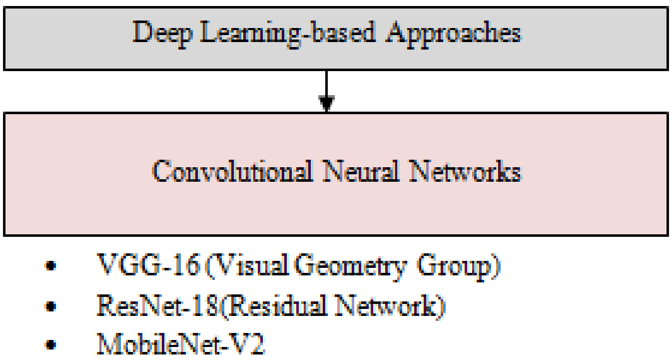


Fig.I.8. Deep Learning-based Approaches.

In deep learning, feature extraction and classification are often performed simultaneously within a single neural network. The neural network learns to extract features that are relevant for classification during the training process. The network is trained to minimize a loss function that measures the difference between the predicted class probabilities and the true class labels, which encourages it to learn features that are useful for accurately classifying the input data.

During training, deep learning models, such as Convolutional Neural Networks (CNNs), automatically learn a hierarchical set of features from the raw input data. The lower layers of the network learn low-level features such as edges and shapes, while the higher layers learn more abstract, complex features such as stroke patterns and digit structures. This hierarchical feature learning is a key advantage of deep learning approaches, as it eliminates the need for manual feature engineering. The model is able to discover more complex and discriminative features that may not be obvious to human experts, allowing it to achieve high performance on tasks such as image classification and recognition.

I.7.1-Convolutional Neural Network :

A convolutional neural network (CNN) serves as a deep learning model extensively employed in tasks related to image analysis and computer vision. Its architecture is tailored to autonomously learn and discern pertinent features from images by traversing through numerous layers of interconnected neurons. CNNs employ convolutional layers to apply filters, enabling the detection of patterns within localized regions of the input data. Subsequently, pooling layers are employed to condense spatial dimensions, thus enhancing computational efficiency. The discerned features are then fed into fully connected layers for classification or regression.

I.7.2-CNN layers :

In a CNN, the network architecture is composed of several layers that perform different operations on the input data. These layers include [22]:

I.7.2.1- The Convolutional Layer :

The convolutional layer, a pivotal element in CNNs within deep learning, employs filters to extract features and patterns from input data. Capturing spatial hierarchies and local patterns, enhances the network's ability to learn meaningful representations. Subsequently, the output undergoes activation functions to introduce nonlinearities.

I.7.2.2-The Pooling Layer :

The pooling layer, another crucial component of CNNs, facilitates downsampling by reducing the spatial dimensions of the input. This reduction effectively decreases the number of parameters and computations in the network while aiding in extracting significant features and bolstering the network's resilience to input variations. Common pooling operations include max pooling and average pooling.

I.7.2.3-Activation Layer :

The activation layer, also referred to as the non-linear layer, serves as a cornerstone in CNNs. It introduces nonlinearity by applying a mathematical function elementwise to the output of the preceding layer. This enables the network to grasp intricate, nonlinear relationships between the input and output, thereby enhancing its ability to model complex patterns and generate more expressive representations of the data. Popular activation functions include the rectified linear unit (ReLU), sigmoid, and hyperbolic tangent (tanh) functions.

I.7.2.4- Fully Connected Layer :

The fully connected layer, also referred to as the dense layer, is a prominent component utilized in various neural network architectures, including convolutional neural networks (CNNs). Within this layer, each neuron establishes connections with every neuron in the preceding layer, forming a fully connected graph. These connections are characterized by weights, the dictate the strength of the connections.

In operation, the fully connected layer conducts computations on the input data by executing matrix multiplication between the input and the weights. Subsequently, a bias term is added, followed by the application of an activation function. This iterative process enables the network to discern intricate relationships within the data and make predictions based on the acquired features.

Typically, positioned at the network's conclusion, the fully connected layer transforms the learned features into a final output. It finds extensive application in tasks such as classification or regression.

I.8- Existing Architectures :

Various CNN architectures have been developed and widely adopted in deep learning applications. Below are notable examples:

I.8.1- VGGNet :

VGGNet was introduced in 2014 by Simonyan et al. Developed by the Visual Geometry Group (VGG) team, VGG-16 is a prominent model featuring 13 convolutional (CONV) layers and 3 fully connected (FC) layers. Noteworthy modifications from previous architectures include the replacement of larger filters with 3 × 3 filters. VGG-16 comprises 138 million parameters and consumes approximately 500 MB of storage space. Additionally, the VGG team designed a deeper variant known as VGG-19, achieving a top-5 error rate of 7.3% in the ILSVRC competition[23] see figure **I.9.**

FigI.9:VGG-16 Architecture.

I.8.2-ResNet :

ResNet, or residual network, stands as a prominent deep learning architecture that has significantly impacted the field of image recognition. Its innovation lies in the introduction of residual connections, enabling the network to learn residual mappings effectively. This mechanism allows for the training of exceptionally deep networks while mitigating issues such as vanishing gradients and promoting smoother gradient flow.ResNet has shown remarkable performance across a spectrum of computer vision tasks, outperforming its predecessors in term of accuracy and attaining state-of-the-art results [24] ,see figure **I.10**.

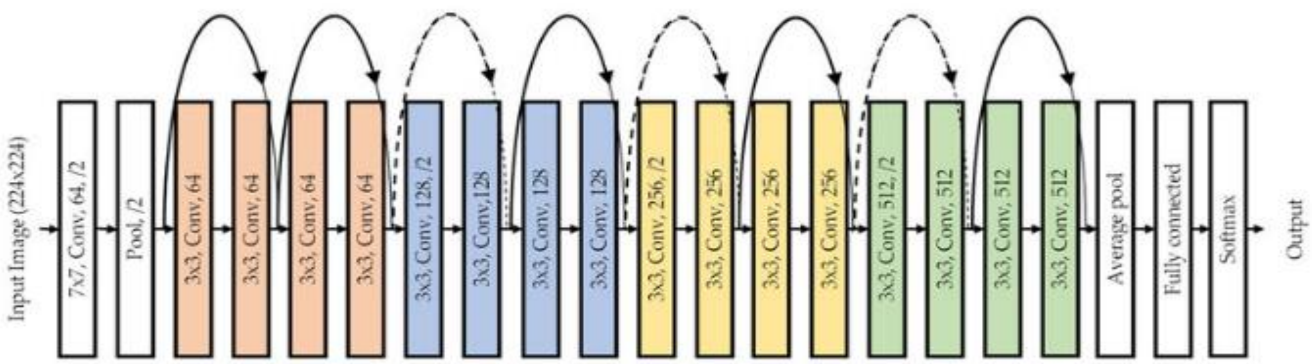


Fig I.10. ResNet-18 Architecture.

I.8.3-MobileNet-V2 :

MobileNet-V2 is an architecture designed to work efficiently on mobile devices. It is based on an inverted residual network structure, where residual connections are established between the bottleneck layers. The intermediate expansion layer uses depthwise convolutions to filter the features as a source of non-linearity. Overall, the MobileNetV2 architecture begins with an initial fully convolutional layer with 32 filters, followed by 19 residual bottleneck layers.

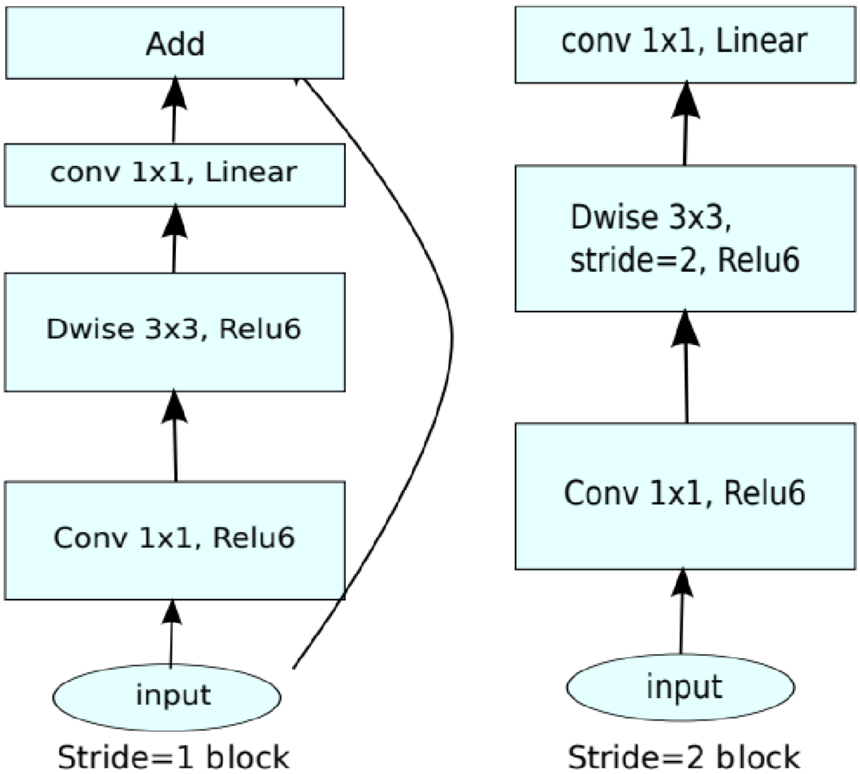


Fig.I.11. MobileNetV2 Architecture.

I.9-Conclusion :

In this chapter, we have presented in a general manner the different phases of handwritten digit recognition systems (HDRS), from the preprocessing phase to classification approaches based on machine learning and deep learning techniques.

Chapter II:
Handwritten Digits Recognition State of the art

II.1 Introduction :

In recent years, significant research efforts have focused on handwritten digit recognition, leading to the emergence of numerous works. This chapter will delves into various studies on handwritten digit recognition. This section will specifically explore the feature extraction methods proposed, the databases utilized, and the results achieved in these works

II.2 Description of Various Subsequent Works:

Classification methods have been developed using different recognition techniques. In the following, we will present works from various authors on isolated handwritten digit recognition.

II.2.1 Machine learning :

II.2.1.1-Uniform grid +SVM :

A. Gattal et al. (2014)

This work proposed a new methodology based on the combination of features to achieve high recognition rates for no normalized isolated handwritten digits. Ten features were used, seven of which were directly extracted from the digit image (Hu's moment invariants, akew Angle, Zernike moments, projection histograms, profile-based features, background features, skeleton-based features), while three features (global features, histogram of contour chain code, ridgelet transform) were extracted from different regions of the image using the uniform grid method. The idea behind this approach is to combine these different features to best represent the digits without any size normalization. The proposed SVM classifier is based on a one-against-all multiclass approach and evaluated on the CVL database of isolated digits, which contains a training set of 21,780 digits and a test set of 7,000 digits. This approach achieved an estimated recognition rate of 96.62%[26].

II.2.1.2-Deep convolutional network + k-mean :

Dundar et al. (2015)

Proposed training a deep convolutional network based on an improved version of the k-means algorithm, which reduces the number of correlated parameters by grouping similar filters together. This approach increases the accuracy of test categorization. The experiments demonstrated that the proposed algorithm outperforms other techniques that learn unsupervised filters. Specifically, the recognition rate reached 74.1% on the MNIST dataset, which consists of 60,000 training images and 10,000 test images[27]..

II.2.1.3-OBIFs+SVM :

A. Gattal et al. (2016)

This work proposed improving the feature extraction step in isolated digit recognition systems without any image size normalization, using texture-based features called oriented basic image features (oBIFs). The author combines oBIFs with background features. oBIFs are used to capture texture information in the form of a vector, while background features exploit the geometric and topological properties of digits for a discriminative representation. Classification is performed using a multiclass support vector machine (SVM) based on the one-against-all approach. The experiments were conducted on the CVL (Computer Vision Lab) database. The database consists of 7,000 digits for training, 21,780 for testing, and 7,000 for validation. All images were binarized using the KittlerMet binarization method prior to feature extraction. The proposed method achieves a recognition rate of 95.21%[28].

II.2.1.4-MCS (HOG and SVM) :

A. Khan et al. (2017)

This work Proposed a novel approach called multiple-cell size (MCS) for the efficient classification of handwritten Digits using histogram of oriented gradient (HOG) features and a support vector machine (SVM) based classifier. The effectiveness of the HOG-based technique highly depends on the selection of cell size during feature extraction. To address this, the MCS approach is introduced to perform HOG analysis and compute the corresponding features. The system achieved an impressive classification accuracy of 99.36% on the MNIST Digit Database using an independent test set strategy. Cross-validation analysis yielded a 10-fold classification accuracy of 99.26%. The proposed system outperforms existing techniques by achieving comparable or even better results while utilizing simpler operations in both the feature space and the classifier space. The confusion matrix and receiver operating characteristics (ROC) plots demonstrate the superior performance of the novel MCS HOG and SVM based digit classification system[29].

II.2.1.5-ADAE+Deep Q-learning :

J. Qiao et al. (2018)

Propose an adaptive deep Q-learning strategy to further improve the accuracy and reduce the execution time of handwritten digit recognition. By combining deep learning and Q-learning for training an adaptive Q-learning deep belief network (Q-ADBN), they extract the main features of original handwritten digit images using an adaptive deep auto-encoder (ADAE) method, adjusting the learning rate adaptively to reduce execution time. Additionally, the Q-learning algorithm is used as a classifier, and the extracted features are considered the current states of the Q-learning algorithm, which typically used to reinforce the learning phase. With respect to the MNIST database, which contains 60,000 images for training and 10,000 images for testing, each image of handwritten digits is 28×28 pixels in size, with each pixel ranging from 0 to 1. The experimental results show a recognition rate of 99.18% [30].

II.2.1.6-HOG+ PCA+SVM :

A.Das et al.(2018)

This study presents a novel approach to handwritten digit recognition that combines the strengths of feature engineering and classical machine learning techniques. The Researchers have utilized the widely-used MNIST dataset and explored two popular feature descriptors – the histogram of oriented gradients (HOG) and Gabor filters - to generate robust feature representations. They then investigated the impact of various dimensionality reduction methods, including principal component analysis (PCA), linear discriminant analysis (LDA), and Isomap, on the transformed feature spaces. The resulting lower-dimensional feature vectors were subsequently used to train a support vector machine (SVM) classifier. The experimental results demonstrate the effectiveness of this approach, with the PCA-HOG model achieving the highest accuracy of 99.29% on the MNIST dataset, while maintaining computational efficiency comparable to that of a convolutional neural network (CNN). Notably, the LDA-HOG model, although slightly less accurate at 98.34%, was able to capture the essential information within just 9 principal components - a 75% reduction in the feature space compared to the PCA-HOG and CNN models. This finding highlights the power of LDA in extracting the most discriminative features, leading to substantial improvements in time efficiency without compromising classification performance. The authors' systematic exploration of feature engineering and dimensionality reduction techniques underscores the continued relevance of classical machine learning approaches in challenging pattern recognition tasks, where they can potentially outperform more complex deep learning models in terms of both accuracy and computational cost [31].

II.2.1.7-HOG+ PCA+SVM :

A.Das et al.(2018)

This study proposes a novel approach to handwritten digit recognition that leverages the complementary strengths of feature engineering and classical machine learning techniques. The researchers utilized the CVL Single Digit dataset and explored two popular feature descriptors - histogram of oriented gradients (HOG) and Gabor filters - to generate robust feature sets. They then investigated the impact of various dimensionality reduction techniques, including principal component analysis (PCA), linear discriminant analysis (LDA), and Isomap, on the transformed feature spaces. The resulting lower-dimensional feature vectors were subsequently used to train a support vector machine (SVM) classifier. The experimental results revealed that the PCA-HOG model achieved an impressive accuracy of 85.14% on the CVL dataset, with computational efficiency comparable to that of a CNN. Interestingly, the LDA-HOG model, while slightly less accurate at 82.63%, was able to capture the essential information within just 9 principal components - a 75% reduction in the feature space compared to the PCA-HOG and CNN models. This finding highlights the power of LDA in extracting the most discriminative features, leading to substantial improvements in time efficiency without compromising classification performance. The authors' systematic exploration of feature engineering and dimensionality reduction techniques underscores the continued relevance of classical machine learning approaches in challenging pattern recognition tasks, such as handwritten digit classification, where they can potentially outperform more complex deep learning models in terms of both accuracy and computational cost [32].

II.2.1.8-HOG and SVM :

Kusetogullari et al.(2020)

The document provides a detailed recognition rate when machine learning algorithms are trained and tested on the ARDIS dataset (Dataset IV), which contains 7,600 28x28 pixel images of digits, with 6,600 for training and 1,000 for testing. The approach used in the document is efficient classification of handwritten digits using histogram of oriented gradient (HOG) features and a support vector machine (SVM) based classifier. The effectiveness of the HOG-based technique is highly dependent on the selection of the cell size during feature extraction. To address this, the document introduces an approach to perform HOG analysis and compute the corresponding features.The system achieved an impressive classification accuracy of 95, 50%on the ARDIS Digit Dataset[33].

II.2.1.9-CNN+multiple classifiers (KNN+RF) :

H. Zhao et al. (2020)

This study presents an innovative approach to handwritten digit recognition that leverages the combined power of convolutional neural networks (CNNs) and an ensemble of diverse classifier models. The researchers first utilized a CNN to extract a rich set of visual features from the MNIST handwritten digit dataset. They then applied feature selection techniques to generate multiple, complementary subsets of these CNN-derived features. These feature subsets were subsequently used to train a variety of classifier models, including k-nearest neighbors (KNNs) and random forests (RFs). By fusing the outputs of these individual classifiers through an algebraic fusion method, the proposed framework achieved an impressive accuracy of 98% or higher on the handwritten digit recognition task - outperforming the use of a single classifier trained on the full CNN feature set. The key to this success lies in the diversity of the base classifiers employed. KNNs and random forests, in particular, are powerful yet distinct models that can capture different types of patterns inherent in the high-dimensional CNN features. This ensemble approach allows the framework to leverage the complementary strengths of the individual classifiers, leading to more accurate and stable predictions compared to any single model. The suitability of random forests for high-dimensional feature spaces further enhances the effectiveness of this classifier fusion strategy for handwritten digit recognition[34].

II.2.1.10-HOG and KNN :

D.Khedidja et al. (2021)

This paper presents a novel approach for handwritten digit recognition that combines the strengths of diverse feature extraction techniques and multiple classifier models. At the core of their system are four distinct feature descriptors - cavities, Zernike moments, Hu moments, and histogram of oriented gradients (HOG) - each capturing complementary aspects of the digit images. The HOG features, in particular, prove to be highly effective, encoding rich shape and edge information in a high-dimensional 1296-element feature vector. When used in conjunction with the simple yet powerful K-nearest neighbor (KNN) classifier, the HOG-based system achieves an impressive 96.57% recognition accuracy on the challenging MNIST handwritten digit dataset. The advantages of this approach are manifold - the HOG features provide strong geometric and photometric invariance, the high feature dimensionality captures extensive structural details, and the no parametric KNN classifier is well-suited for handling such high-dimensional data. By exploring the complementary strengths of diverse feature extractors and classifiers, researchers have demonstrated a robust and effective framework for tackling the complex task of handwritten digit recognition[35].

study	Years	Methodes	dataset	Performance
A. Gattal et al	2014	Uniform grid + SVM	CVL	96.62%
Dundar et al	2015	deep convolutional networks +K-means	MNIST	74.1%
A. Gattal et al	2016	OBIFs+ SVM	CVL	95.21%
A. Khan et al	2017	HOG +SVM	MNIST	99.26%.
J. Qiao et al	2018	AEPA+ Deep Q-Learning	MNIST	99.18%.
A.Das et al	2018	HOG+ PCA+SVM	CVL	85.14%
A.Das et al	2018	HOG+ PCA+SVM	MNIST	99.29%
Kusetogullari et al	2020	HOG+SVM	ARDIS	95,50%
D.Khedidja Et al	2021	HOG+KNN	MNIST	96.57%

Table II.1. Subsequent Works in Digit Recognition using machine learning approaches

II.2.2- Deep Learning :

II.2.2.1-GAT (Global Affine Transformation) :

Wakahara et Yamashita (2014)

Wakahara and Yamashita (2014) proposed global affine transformation (GAT) as a model-based matching method that can compensate for affine transformations present in an input model. The GAT correlation method has demonstrated high performance in character recognition and object matching. A new matching measure called nearest neighbors distance of equi-gradient direction (NNDEGD) is used in conjunction with the GAT correlation method. The NNDEGD is simply a parameter of the Gaussian window function used in the GAT correlation method, which is equal to the average minimum distance between a point in one image and another point in the other image with the same gradient direction. This value is then used as a new matching measure. Second, they extended the GAT correlation method to handle changes in stroke width in addition to affine transformation. Finally, they applied the original and extended versions of the GAT correlation method to FC-NN classification experiments using the MNIST database. These experiments were carried out efficiently for the first time because they significantly reduced the complexity and computational memory load compared to the original GAT correlation method. The recognition rate reached 99.51% on the MNIST database, which consists of 60,000 training images and 10,000 test images [36].

II.2.2.2-Global projection transformation (GPT) :

Wakahara et Yamashita (2016)

The matching-based recognition model does not require a learning process, and the matching result provides intuitive and geometric information. Wakahara et al. proposed the global affine transformation (GAT) for matching correlation, which can compensate for affine transformations imposed on a template. The GAT matching correlation with an acceleration method and a new matching measure called nearest-neighbor distance of equi-gradient direction (NNDEGD) achieves high performance in experiments using the MNIST database. The GAT matching measure has been extended to the global projection transformation (GPT) matching measure to allow for 2D projection transformation deformation. Their work first involves developing an acceleration method for GPT correlation matching. Second, to improve recognition performance, they applied the curvature of stroke contours to the matching measure. Curvature is often used as a character feature, but in this case, it was used as a weight in NNDEGD. Third, to investigate the performance of the proposed methods, they applied the image matching and recognition to the MNIST and IPTP databases using k-nearest neighbor (k-NN). In the experiment with the MNIST database , GPT correlation matching with the curvature-weighted NNDEGD matching measure achieved the highest recognition rate of 99.70% using k-NN methods [37].

II.2.2.3-SNNs (spiking neural networks) :

S. R. Kulkarni and B. Rajendran (2018)

Focus on applying a precise supervised learning algorithm based on the problem of handwritten digit recognition using a network optimized in terms of the number of learning parameters according to energy and memory constraints. To demonstrate that a spiking neural networks (SNN) capable of achieving high accuracy and efficiency, they use the recently proposed Normalized Approximate Descent-Normad algorithm. They also trained an equivalent Artificial Neural Network (ANN) with the same architecture for comparison with the proposed SNN. After optimizing the network hyperparameters, the SNN and ANN were trained on the entire MNIST dataset (60,000 images) for 20 epochs. The proposed SNN outperforms a non-convergent equivalent ANN, achieving a recognition rate of 98.17%, compared to 98% for the ANN [38].

II.2.2.4-VGG-16(Visual Geometry Group) :

C.Shuhong et al (2019)

The VGG-16 architecture, initially designed for large-scale image classification tasks, has been adapted for digit recognition using the MNIST dataset. This involves adjusting the input layer to fit 28x28 grayscale images of handwritten digits and modifying the output layer to predict the ten digit classes (0-9). The model comprises 13 convolutional layers, followed by max-pooling layers and three fully connected layers, tailored for the simpler task and smaller input size of MNIST. During training, the model minimizes a loss function by adjusting its weights through backpropagation and gradient descent. Evaluation on a separate test set measures its effectiveness, typically achieving accuracies of 99,97% on MNIST. Fine-tuning techniques, including regularization and hyperparameter optimization, enhance the model's performance and generalization ability, leveraging the robustness of the VGG-16 architecture [39].

II.2.2.5-CapsNet (Capsule Network) :

Y. Tan and H. Yao (2021)

A handwritten digit recognition system based on a modified topology deep capsule network was proposed for comparison with standard capsule networks. The new architecture utilizes a combination of convolutional layers with modified kernel sizes, primary capsules extracting higher-level features, and final digit capsules for classification. Dual iterative dynamic routing between capsule layers, inspired by the original dynamic routing algorithm, allows for improved alignment of predicted poses. The proposed system takes grayscale images of 28x28 pixels as input. Trained by gradient backpropagation without explicit feature extraction, it learns relevant representations and the recognition task jointly. The application of elastic deformations to the data increases the variability of the learning examples. Tested on the MNIST dataset consisting of 60,000 training images and 10,000 test images, the network achieves a recognition rate of 99.62%, which is higher than the 99.55% of the classical capsule network, and shows an improved ability of 93.53% to separate overlapped digits[40].

II.2.2.6-ResNet-18(residual network) :

P.Mhasakar et al. (2021)

The new architecture utilizes a combination of convolutional layers with modified kernel sizes, the primary extraction of higher-level features, and the final digit for classification. Dual iterative dynamic routing between ResNet-18 layers, inspired by the original dynamic routing algorithm, allows for improved alignment of predicted poses. The proposed system takes grayscale images of 28x28 pixels as input. Trained by gradient backpropagation without explicit feature extraction, it learns relevant representations and the recognition task jointly. The application of elastic deformations to the data increases the variability of the learning examples. When tested on the MNIST dataset consisting of 60,000 training images and 10,000 test images, the network achieved a recognition rate of 98.13% [41].

II.2.2.7-Deep neural network :

A. Kumar et al. (2021)

This paper proposes a deep neural network model for handwritten digit classification and recognition using the ARDIS dataset of Swedish handwritten digits. The ARDIS dataset consists of four distinct sets extracted from approximately 15,000 Swedish church records from the 19th and 20th centuries:

1) 10,000 color images of sequences of four digits (years).
2) 7,600 individual images of digits in their original form.
3) 7,600 cleaned versions of the previous images.
4) 7,600 resized images of digits in a 28x28 pixel format.

The dataset contains 7,600 samples, with 6,600 samples used for training and 1,000 samples used for testing. The proposed deep neural network model consists of six layers and utilizes the ReLU and softmax activation functions. Using the ARDIS dataset, the model achieved a training accuracy of 99.76% and a testing accuracy of 98.70%, which surpasses the performance of previous research conducted on this dataset.

II.2.2.8-B-ResNet :

P.Mhasakar et al. (2021)

This study introduces a novel architecture called Bayesian ResNet (B-ResNet) that combines the powerful ResNet-18 framework with Bayesian inference to achieve robust and uncertainty-aware handwritten digit recognition. Unlike traditional ResNets that rely on point estimates for the model weights, B-ResNet treats the weights as probability distributions, specifically Gaussian distributions parameterized by mean and standard deviation. This Bayesian approach allows the model to capture the uncertainty in its predictions, which is particularly useful when dealing with unseen data or classes. Furthermore, the use of a prior distribution as part of the Bayesian inference process acts as a regularizer, helping to prevent overfitting when training data is limited. The inherent advantages of the ResNet architecture, such as skip connections and deep depth, make B-ResNet resilient to variations in handwritten digits, including writing style, stroke thickness, and rotations. Experiments conducted on the MNIST dataset of 70,000 handwritten English digit images demonstrate the efficacy of the proposed approach, with B-ResNet achieving an impressive accuracy of 99.89% - which is a significant improvement over the standard ResNet-18. The integration of Bayesian principles into the ResNet framework showcases the potential of hybrid architectures that leverage the complementary strengths of deep learning and probabilistic modeling, paving the way for more advanced and versatile pattern recognition systems[42].

II.2.2.9-AlexNet :

A.Zebdi et al (2022)

This article proposes a novel handwritten digit recognition method that leverages deep learning techniques in conjunction with data augmentation using encrypted digit images and image fusion strategies. Three distinct approaches are investigated: testing on the original CVL database and individually encrypted databases generated via techniques such as AES, RSA, Arnold's Cat Map, and Henon Map; augmenting the original CVL data by combining it with the encrypted databases; and fusing the original CVL images with the encrypted images using a discrete wavelet transform (DWT) fusion method. An AlexNet deep learning architecture, comprising convolutional, max-pooling, fully connected layers, and dropout regularization, is employed for digit recognition. Comprehensive experiments demonstrate that the proposed fusion approach, which integrates the original CVL images with those encrypted using Arnold's cat map, achieves the highest accuracy of 98.21%. Furthermore, the data augmentation strategy of combining the CVL dataset with the Henon Map encrypted database yields an impressive accuracy of 97.07%, outperforming several state-of-the-art handwritten digit recognition techniques evaluated on the CVL benchmark. The authors attributed these performance gains to the data augmentation and fusion methodologies, which effectively capture writer-specific characteristics such as ink trace direction, curvature, slant, and width, thereby enhancing the robustness of the recognition system[43].

II.2.2.10-Skyrmions :

M.Lee et al (2023)

This paper showcases a groundbreaking utilization of magnetic skyrmion lattices within a thin-plate magnet as a potent reservoir for reservoir computing, achieving a remarkable accuracy of over 88% on the MNIST handwritten digit recognition task, surpassing a baseline echo state network model by approximately 10%. This exceptional performance is ascribed to the heightened nonlinearity in the transformation that maps input data onto a higher-dimensional information space, facilitated by the interference of spin waves within the skyrmion lattice. Skyrmions present distinct advantages over other spintronics reservoirs due to their simple creation through the application of a static magnetic field, eliminating the need for intricate nanofabrication processes. The findings underscore the immense potential of skyrmion-based reservoirs for efficient and low-power machine learning applications, offering a promising alternative to conventional CMOS architectures that are vulnerable to environmental factors and high energy consumption. Key innovations include the utilization of a slightly distorted skyrmion lattice as the reservoir, as opposed to random skyrmion configurations, and leveraging nonlinear spin wave interference effects to bolster performance. In essence, this research heralds an exciting new frontier in the application of magnetic skyrmions for reservoir computing and pattern recognition tasks, presenting notable advantages over traditional electronic implementations [44].

II.2.2.11-Dignet :

D.Mondal et al (2023)

The article presents "Dignet", a convolutional neural network (CNN) based model for efficient handwritten digit recognition on the popular MNIST dataset. The proposed CNN architecture employs convolutional layers with ReLU activation functions, max pooling layers for dimensionality reduction, and fully connected layers for the final classification. Various configurations of layer parameters are explored to optimize performance. Data augmentation techniques are utilized to further improve the recognition accuracy. Through comprehensive evaluations on the MNIST test set, the Dignet model achieved an impressive accuracy of 99.70%, outperforming numerous previous techniques for handwritten digit recognition. The authors provide in-depth explanations of the CNN architecture design, the underlying mathematical computations, and detailed results across different digit classes. They concluded that the CNN-based Dignet approach offers superior accuracy and speed compared to traditional methods for this task. Additionally, the paper covers relevant background on handwritten character recognition, deep learning techniques for this domain, and prior work involving CNNs and other neural network models applied to the MNIST dataset[45].

II.2.2.12-DICNN (Dual Input Convolutional Neural Network) :

A.Azgar et al (2024)

This study introduces a novel deep learning-based approach called the dual-input convolutional neural network (DICNN) for handwritten digit recognition, leveraging the widely-used MNIST dataset. The MNIST dataset consists of 70,000 handwritten digit samples, with 60,000 images used for training and 10,000 for testing, each with a resolution of 28x28 pixels. The researchers compared the performance of their DICNN model against several standard machine learning algorithms, including K-nearest neighbors, support vector machines, logistic regression, neural networks, random forest, naive bayes, and decision trees. The DICNN model, a modified CNN architecture, takes two inputs: the original handwritten digit image and a preprocessed version of the same image. These two input streams are processed through separate convolutional and pooling layers, and the resulting features are then concatenated and passed through fully connected layers for classification. The DICNN model demonstrated impressive results, achieving a training accuracy of 98.9% and a testing accuracy of 98.7% on the MNIST dataset, significantly outperforming the standard machine learning algorithms, which reported accuracy levels ranging from 74% to 97%. The authors' innovative dual-input approach and the model's ability to effectively extract and combine features from the original and preprocessed images contributed to its superior performance, highlighting the potential of custom-designed deep learning architectures for challenging pattern recognition tasks[46].

Study	Years	Methodes	Dataset	Performance
Wakahara et Yamashita	2014	GAT	MNIST	99,51%
Wakahara et Yamashita	2016	GPT	MNIST	99,71%
S. R. Kulkarni et B. Rajendran	2018	SNN (Spiking Neural Networks)	MNIST	98,17%
C.Shuhong et al	2019	VGG-16	MNIST	99.97%.
Y. Tan et H. Yao	2020	Capsule network	MNIST	99.55%
P. Mhasakar et al	2021	ResNet-18	MNIST	98.13%.
A. Kumar et al	2021	Deep neural network	ARDIS	98.70%
P.Mhasakar et al	2021	B-ResNet	MNIST	99.89%
A.Zebdi at al	2022	AlexNet	CVL data augmentation with Henon Map	97.07%
A.Zebdi at al	2022	AlexNet	Fusion of CVL and Arnold Cat Map images	98.21%
D.Mondal et al	2023	Dignet	MNIST	99.70%
M.Lee et al	2023	Skyrmions	MNIST	88%
A.Azgar et al	2024	DICNN	MNIST	98.90%

Table II.2. Subsequent Works in Digit Recognition using Deep learning approaches

II.3-Conclusion:

In this chapter, we dedicated our efforts to conducting an extensive review of the most up-to-date research in the field, up to the year of writing this dissertation (2024). Our goal was to thoroughly examine the datasets that were utilized in previous studies, the methodologies that were employed, and the outcomes that were achieved by each of these studies. To ensure a comprehensive overview, we compiled a summary table that encompasses all the papers that were cited in this chapter. This table serves as a valuable resource, providing a consolidated reference for the key findings and contributions of each study mentioned. By presenting this summary, we aim to provide readers with a holistic understanding of the existing body of work and establish the context for our own research endeavors in subsequent chapters.

Chapter III:
The Experimental results

III.1- Introduction:

This chapter presents the experimental results obtained from developing and evaluating a novel hybrid approach for handwritten digit recognition. The proposed method synergistically combines the powerful feature extraction capabilities of the VGG-16 convolutional neural network with the flexibility and interpretability of the support vector machine (SVM) classifier. The experiments were conducted on three widely-used handwritten digit datasets - MNIST, ARDIS, and CVL - to comprehensively assess the model's performance and highlight its robustness across diverse data.

III.2- Development Tools:

III.2.1- Hardware Tools:

This program was created on a type of HP laptop with the following specifications:

• Processor: Intel ® Core (TM) i7-7700 HQ CPU 2.80 GHz.

• Installed memory (RAM): 16.0GB.

• System type: 64-bit operating system.

• Operating system: Windows 10.

III.2.2-Software Tools:

III.2.2.1- Google Colab:

The Colaboratory, or "Colab" for convenience, is a product developed by Google Research. It offers users the ability to write and execute Python code directly from their web browser. It serves as an excellent platform for individuals interested in machine learning, data analysis, and educational purposes. Colab functions as a hosted Jupyter notebook service, eliminating the need for any setup or configuration. Furthermore, it provides users with free access to computational resources, including GPUs, enhancing the overall computing experience[47] see figure **III.1**.

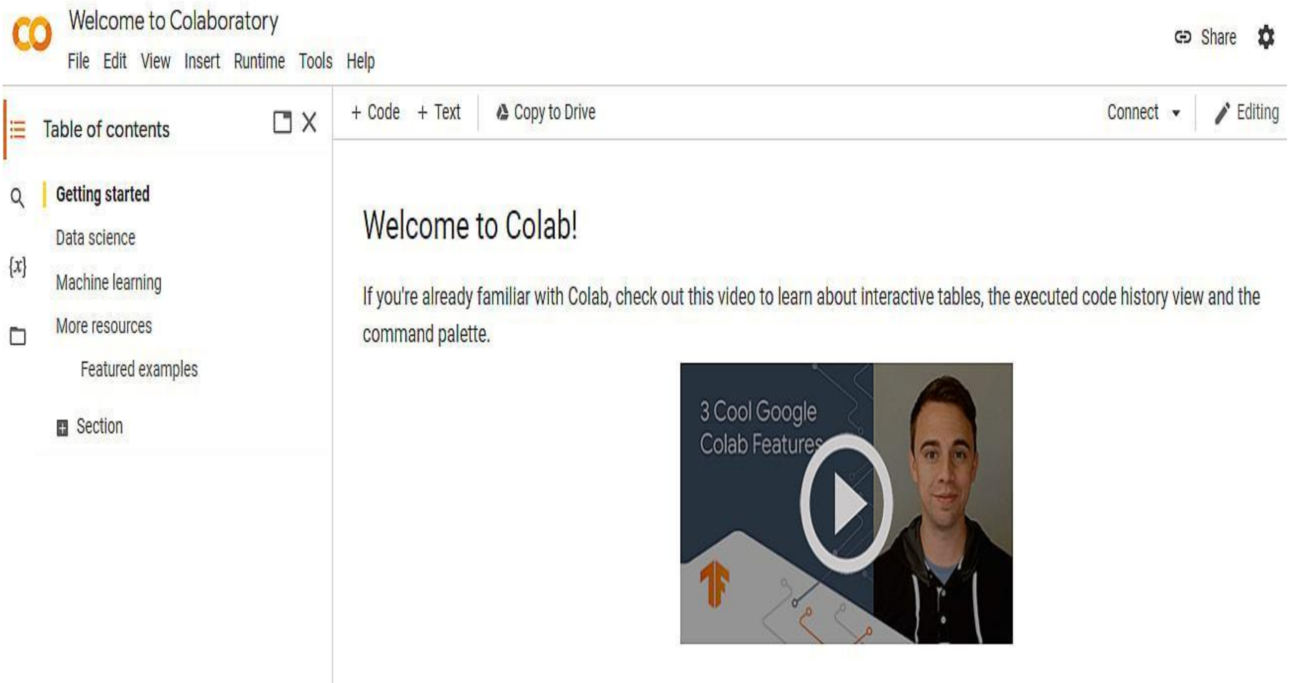


Fig III.1. The Google Colab Work Environment

III.2.2.2-Advantages of Google Colab:

- The coding skills were enhanced in the Python programming language.
- This approach facilitates the development of deep learning applications using popular Python libraries such as Keras, TensorFlow, PyTorch, and OpenCV.
- Requires no setup.
- Access to a GPU for accelerated computation.
- The platform is operated on Google servers.
- Colab documents are saved directly to Google Drive.

III.2.2.3-Disadvantages of Google Colab:

- Limited execution rate.
- Each execution may yield different results from the previous one.

III.2.2.4-Python language :

The utilized version is 3.9. Python is a programming language known for its simplicity and power, making it easy to learn and use. It offers efficient high-level data structures and employs a straightforward yet effective approach to object-oriented programming. The elegant syntax, dynamic typing, and interpretability of Python make it a versatile language suitable for scripting and rapid application development across various domains and platforms [48].

III.2.3- Libraries used:

Python has several libraries, and each library has a specific task, making it a very rich and widely used language. The libraries used in this work were: TensorFlow, Keras, and NumPy.

III.2.3.1-TensorFlow:

TensorFlow is an open-source software library for high-performance numerical computation. Its flexible architecture allows easy computation across a variety of platforms (CPUs, GPUs and TPUs), from desktops to server clusters to mobile devices. Initially developed by researchers and engineers from the Google Brain team within Google's AI organization, it focuses on machine learning and deep learning.

For installing TensorFlow, we go to the start menu of our Windows machine, search for 'cmd' and use the right-click, choose 'run as administrator', and then execute a command to install TensorFlow. Here is the command:

III.2.3.2-Keras:

Keras is the most widely used tool in Python for deep learning. This open-source library, created by François Chollet, allows for easy and rapid creation of neural networks, based on major frameworks (TensorFlow, PyTorch). Keras reduces the development time of a neural network prototype by 30%.

III.2.3.3-NumPy:

NumPy is a library for numerical computations with Python. It enables the easy handling of arrays of numbers, sophisticated functions (broadcasting), and can also be integrated with C/C++ and Fortran code.

III-3 Dataset used:

III.3.1-Cvl dataset :

The CVL Digit Strings dataset is a collection of digit strings extracted from the CVL Single Digit dataset. It includes 10 different digit strings written by approximately 120 different writers. This dataset contains a total of 1,262 training images. The CVL Single Digit dataset, on the other hand, consists of 7,000 single digits, with 700 digits per class. These single digits were extracted from the digit strings in the CVL Digit Strings dataset. The CVL Single Digit dataset provides a focused collection of individual digits for training and recognition purposes. It is important to note that the validation set, which has the same size as the training set, consists of different writers. The validation set is primarily used for parameter estimation and validation, rather than supervised training. This ensures that the models are evaluated on unseen data, enhancing the reliability of the validation results. In summary, the CVL Digit Strings dataset contains digit strings written by various writers, while the CVL Single Digit dataset is derived from these strings and consists of individual digits. Both datasets play a crucial role in training and evaluating models for handwritten digit recognition[49] ,see figure **III.2**.

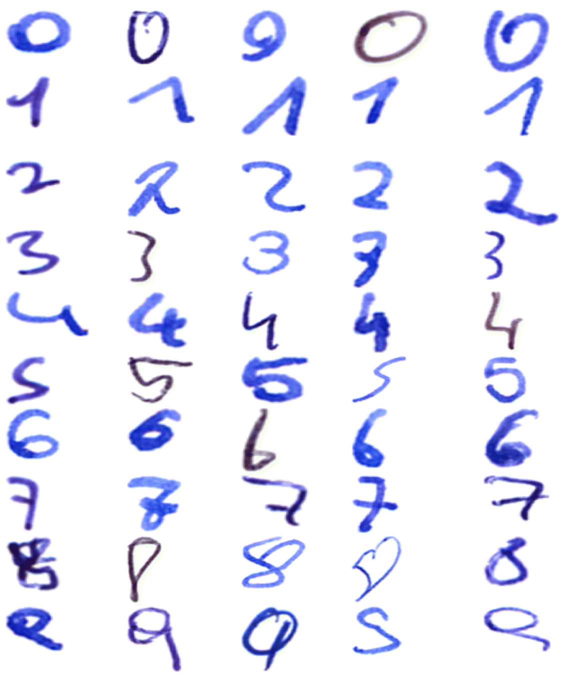


Fig III.2. The CVL dataset

III.3.2-Mnist Dataset:

MNIST stands for "Modified National Institute of Standards and Technology." It is a well-known dataset in the field of machine learning and computer vision. The dataset consists of a collection of grayscale digital images representing handwritten numbers from 0 to 9.

MNIST is typically composed of 60,000 images used for training and 10,000 images used for validation and testing. The images are 28x28 pixels in size and are converted into a digital dataset with values ranging from 0 to 255, where white represents a value of 0, and black color represents a value of 255. The MNIST dataset is widely used for developing and testing machine learning models in the field of optical image classification. The main objective is to correctly identify and classify handwritten numbers in images [50], see figure **III.3**.

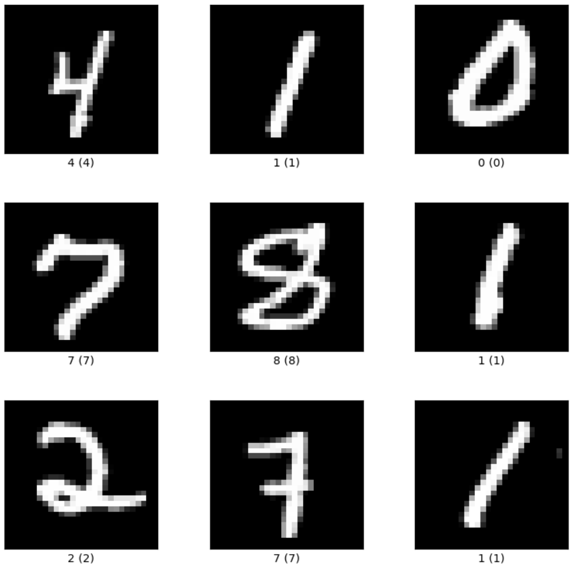


Fig III.3. The MNIST dataset [13**]**

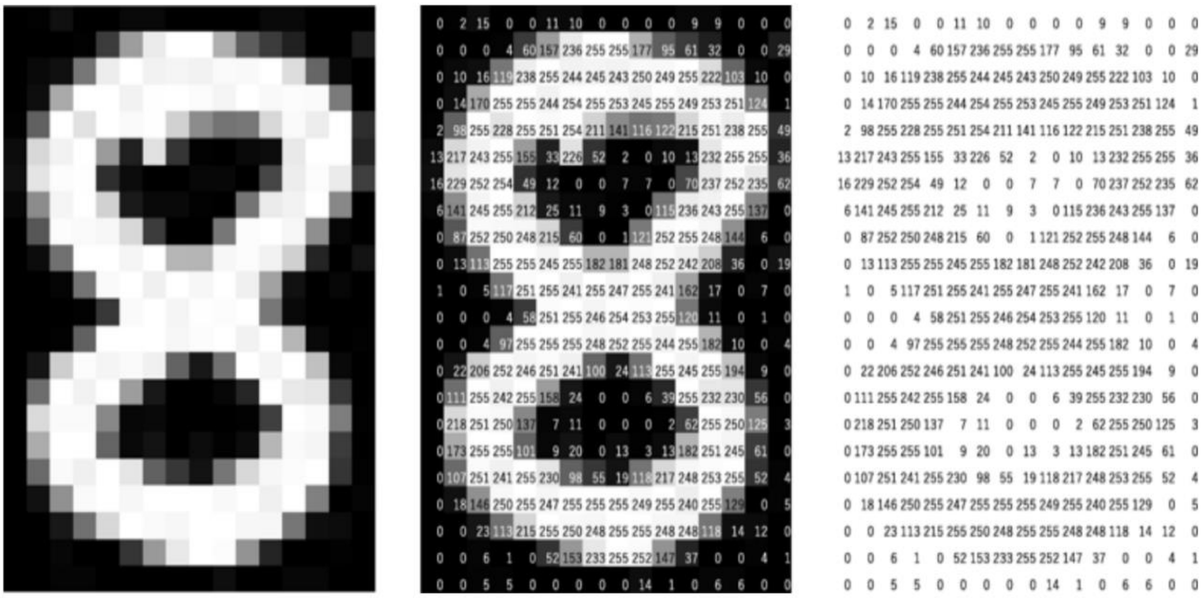


Fig.III.4. example of number 8 in the MNIST database.

III.3.3-ARDIS Dataset:

The ARDIS dataset from Arkiv Digital Sweden also contains images of handwritten digits. [Kusetogullari et al., 2019] extracted these digits from Swedish religious registers, written by various priests from the 19th to the 20th centuries. As a result, the handwriting varies from one register to another, providing a greater diversity of digit representations. Furthermore, since calligraphy is different from modern handwriting, this dataset may facilitate better recognition when applied to historical documents.The dataset is divided into four subsets,

Dataset 1: This dataset contains 10,000 images, each composed of a string of 4 digits (a 4-digit number) directly extracted from the documents without preprocessing. The images are encoded in RGB and JPG formats but are not normalized, meaning that they do not have the same dimensions. This dataset is further divided into three parts corresponding to different dates.

Dataset 2: This dataset contains 7,600 images of digits that were not preprocessed and were, directly extracted from the documents. They are also encoded in RGB and JPG formats.

Dataset 3: Contains 7,600 images of digits with clean backgrounds, meaning any ink blots, for example, have been removed.

Dataset 4: Contains 7,600 images of digits listed in .csv format files and are in black and white. There are four folders containing the training and testing images as well as corresponding labels. [51]

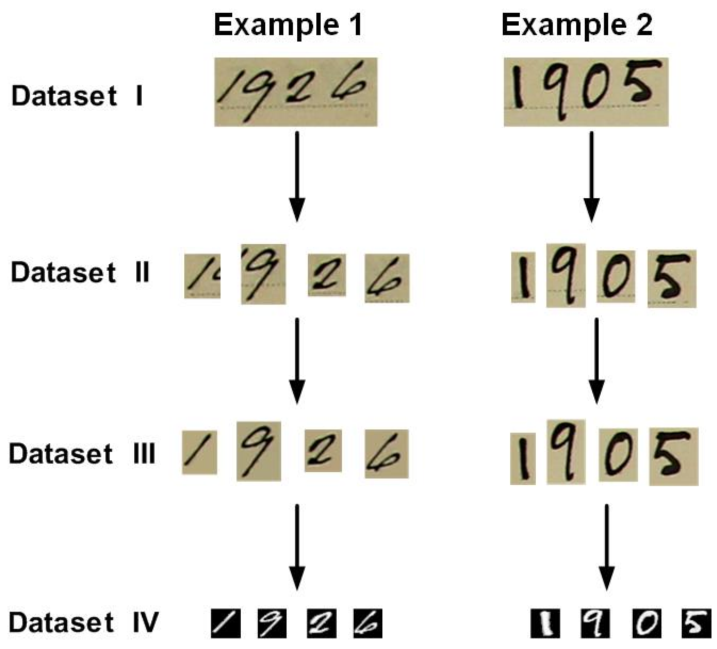


Fig.III.5. The ARDIS dataset

III.4-Performance Metrics:

One of the most useful and commonly used evaluation metrics for object detection is the mean Average Precision (mAP). This metric is used by most modern object detection and instance segmentation frameworks to compare their performances. To understand mAP, it is first necessary to have an intuition of sensitivity and specificity, which are statistical measures of the performance of a binary classification test.

Sensitivity, also called true positive rate or recall, measures the proportion of actual positives that are correctly identified. For object detection, the determination of true positives (TP) and false positives (FP) depends on a predefined IoU (Intersection over Union) threshold, which is typically set at 50%.

Specificity, also called true negative rate, is the proportion of actual negatives that are correctly identified. To better understand:

- True Positive (TP): There is an object, and the algorithm detects it as an object.

- False Positive (FP): There is no object, but the algorithm detects an object.

- False Negative (FN): There is an object, but the algorithm fails to detect it.

- True Negative (TN): There is no object, and nothing is detected.

Accuracy: Accuracy is a measure that indicates if a model/algorithm is correctly trained and how well it performs. Accuracy is calculated using the following formula:

Accuracy = (TP + TN)\(TP + TN + FP + FN)

Recall: it is defined as the ratio of the number of true positives to the sum of true positives and false negatives. It is a measure of the number of positive points that our model is able to recall from the data.

Recall=TP(TP+FN)

The F1-score: it is the weighted harmonic mean of precision and recall. The closer the F1-score value is to 1.0, the more our model is considered to have a balance between precision and recall. It is a measure that takes into account both precision and recall to evaluate the overall performance of the model.

F1−Score= (2×Accuracy ×Recall)(Accuracy+Recall)

Confusion Matrix: A confusion matrix is a technique used to summarize the performance of a classification algorithm. Each entry M_{i,j}in a confusion matrix is equal to the number of observations known to be in group (i) and predicted to be in group (j).

III.5-Proposed Model:

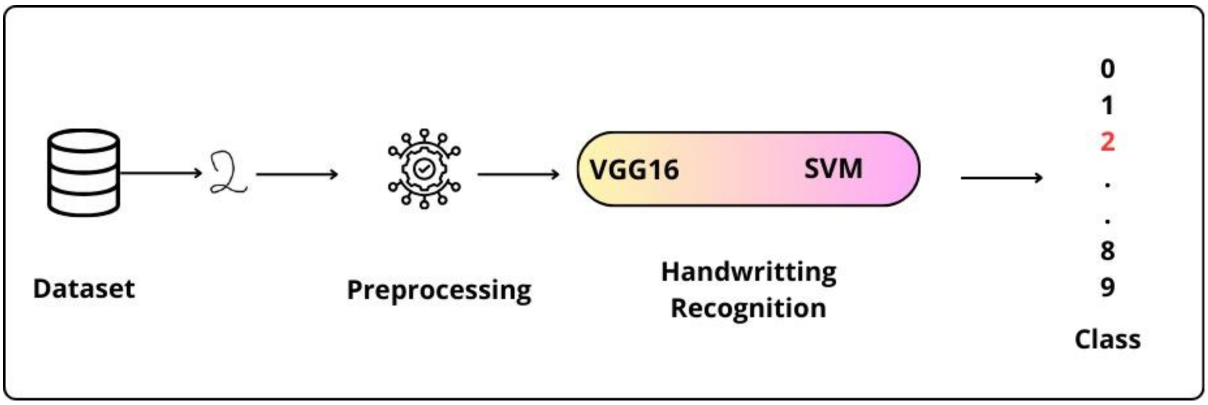


Fig III.6. The proposed handwriting digit recognition.

In this study, the VGG-16 feature extraction method combined with an SVM classifier is employed for image classification tasks. The VGG-16 network, developed by the Visual Geometry Group at the University of Oxford, is utilized for its robust feature extraction capabilities, particularly in the context of handwritten digit images.

III.5.1-Feature Extraction with VGG-16:

The VGG-16 network, developed by the Visual Geometry Group at the University of Oxford, is renowned for its effectiveness in extracting features from images. Comprising 16 layers, including convolutional, pooling, and fully connected layers, VGG-16 is pre-trained on the extensive ImageNet database. We utilize intermediate layers of VGG-16 to extract hierarchical features from handwritten digit images, facilitating superior classification performance.

Architecture Overview:

- Layers 1 and 2: 2D convolution layers with 64 filters of 3x3, extracting low-level features.
- Layer 3: Max-pooling layer, reducing feature map size to 14x14.
- Layers 4 to 6: Additional convolution and max-pooling operations with 128 and then 256 filters, extracting higher-level features.
- Layers 7 to 14: Progressive spatial reduction to 1x1 while increasing depth to 512 feature maps.
- Layer 15: Flattening layer converting 3D output to a 1D vector of 512 values.
- Layers 16 to 18: Fully connected layers producing class predictions among 10 classes, totaling around 134 million trainable parameters.

III.5.2-Classification with SVM:

The Support Vector Machine (SVM) classifier is known for its ability to generalize well on classification problems. It seeks to find the optimal hyperplane that best separates the different classes (digits from 0 to 9). The SVM parameters, such as the regularization parameter C and the kernel parameter σ, need to be optimized to obtain the best performance. A grid search or cross-validation can be used to find the best hyperparameters. The SVM is faster and requires less training data than deep neural networks. It also offers better interpretability of the classification decisions, as it is possible to analyze the support vectors.

III.6-Experimental Validation:

The proposed model, which combines VGG-16 feature extraction with an SVM classifier, was validated using three different datasets: MNIST, ARDIS, and CVL. Each dataset presents unique characteristics and challenges, providing a comprehensive evaluation of the model's performance in handwritten digit classification

III.6.1-Results on the MNIST Dataset:

The MNIST dataset was chosen as a benchmark to evaluate the classification model's performance. Images were normalized and resized to 28x28 pixels. A pre-trained VGG-16 network was used to extract features from the images, and an SVM classifier was trained on the extracted features. The results show an accuracy of 98% on the MNIST test dataset, with a confusion matrix revealing minimal classification errors , primarily between the digits 3 and 8 due to their similar shapes Refer to Figure 7 for the MNIST confusion matrix.

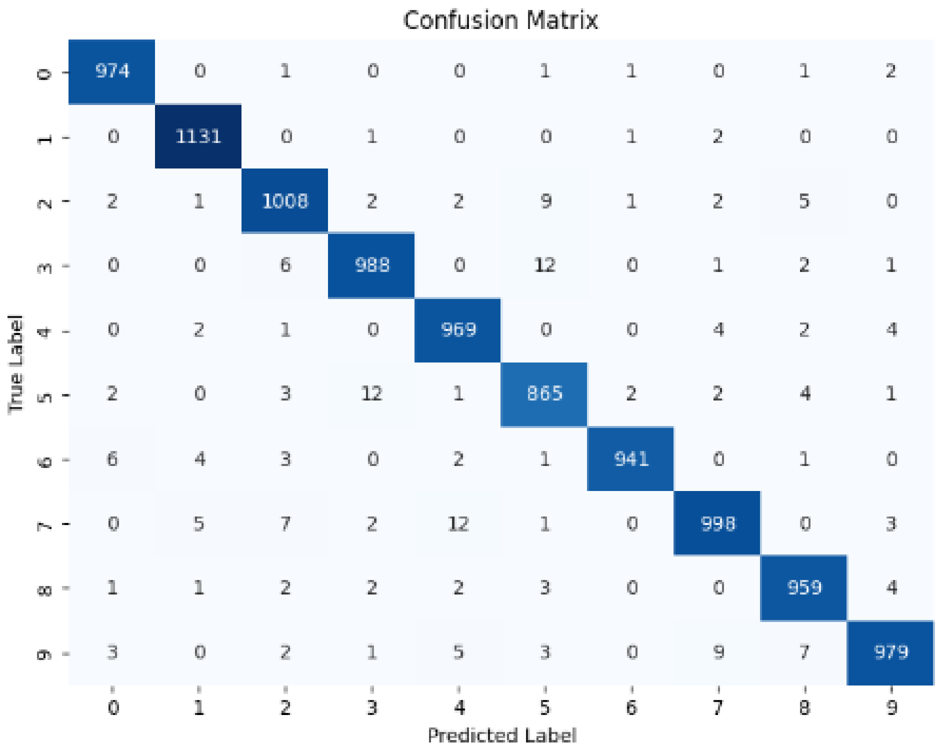


Fig.III.7 confusion matrix of model trained in MNIST dataset

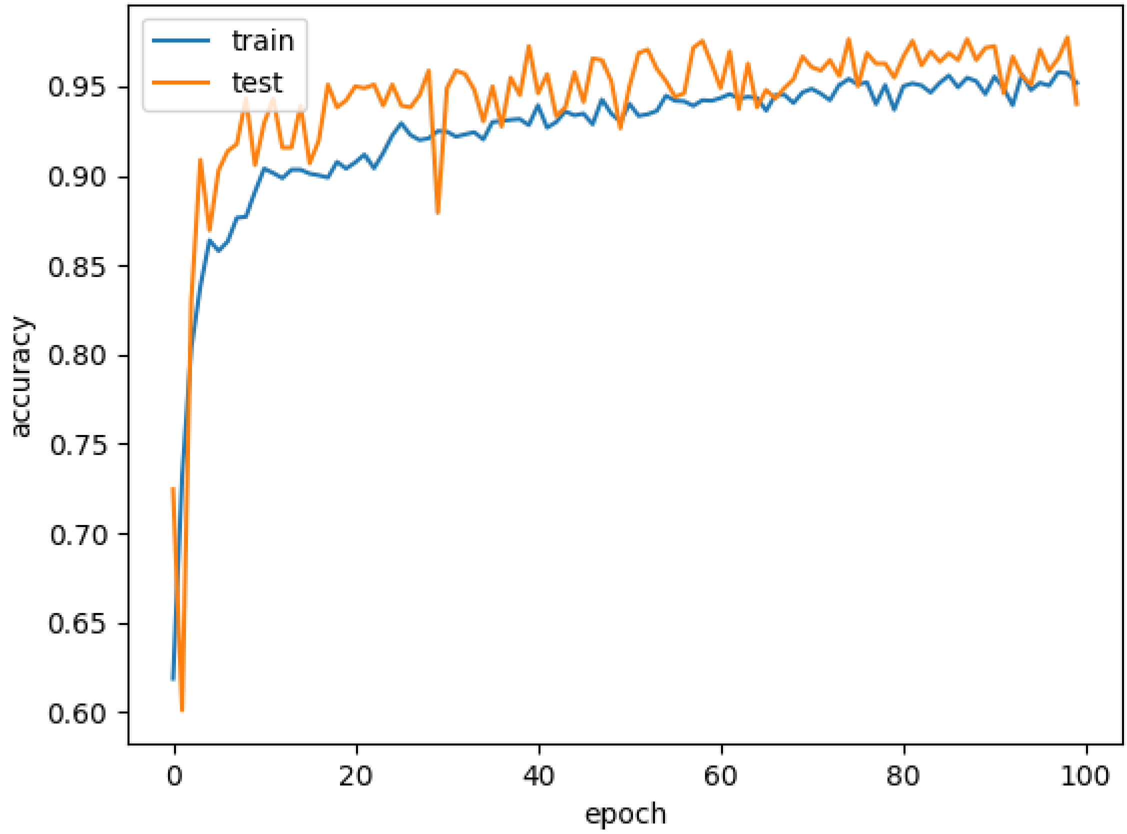


Fig.III.8. Graph of accuracy for the MNIST dataset

III.6.2- Results on the ARDIS Dataset

The ARDIS dataset consists of handwritten digits extracted from Swedish religious registers, exhibiting greater variability in writing styles. Images were resized and normalized, and the same VGG-16 network was used to extract features. An SVM classifier was trained on the features. The results reveal an accuracy of 93.10% on the ARDIS dataset, with a confusion matrix highlighting increased confusion between digits with similar shapes as depicted in Figure 9.

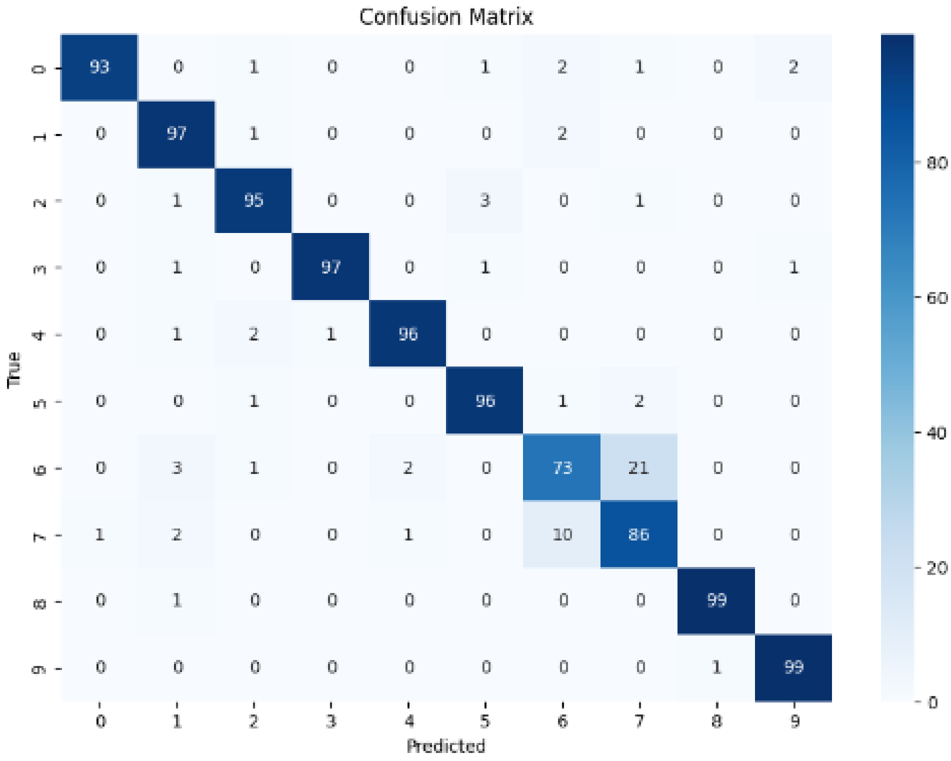


Fig.III.9. confusion matrix of model trained ARDIS in A dataset

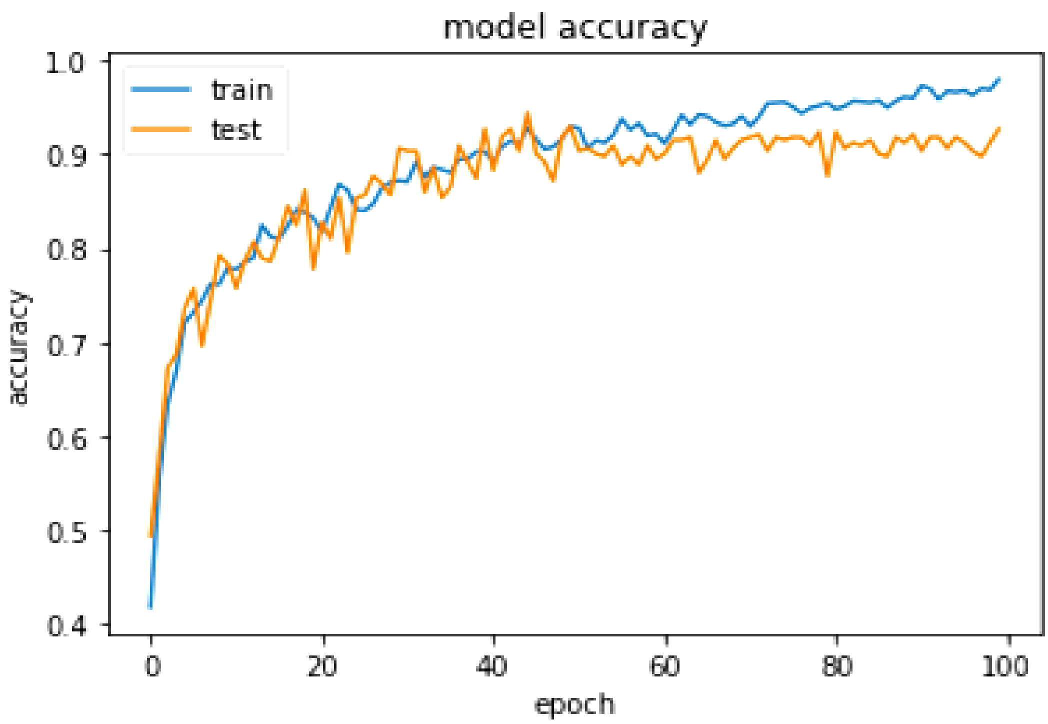


Fig.III.10. Graph of accuracy for the ARDIS dataset

III.6.3-Results on the CVL Dataset:

The CVL dataset contains digit strings and single digits written by various authors, offering a diversity of writing styles. Images were normalized and resized, and the VGG-16 network was used to extract features. An SVM classifier was trained on the features. The results obtained on the CVL dataset are very promising, with an accuracy of 98.46%. The confusion matrix reveals minimal classification errors shown in Figure 11, indicating the model's excellent ability to recognize digits in various writing styles.

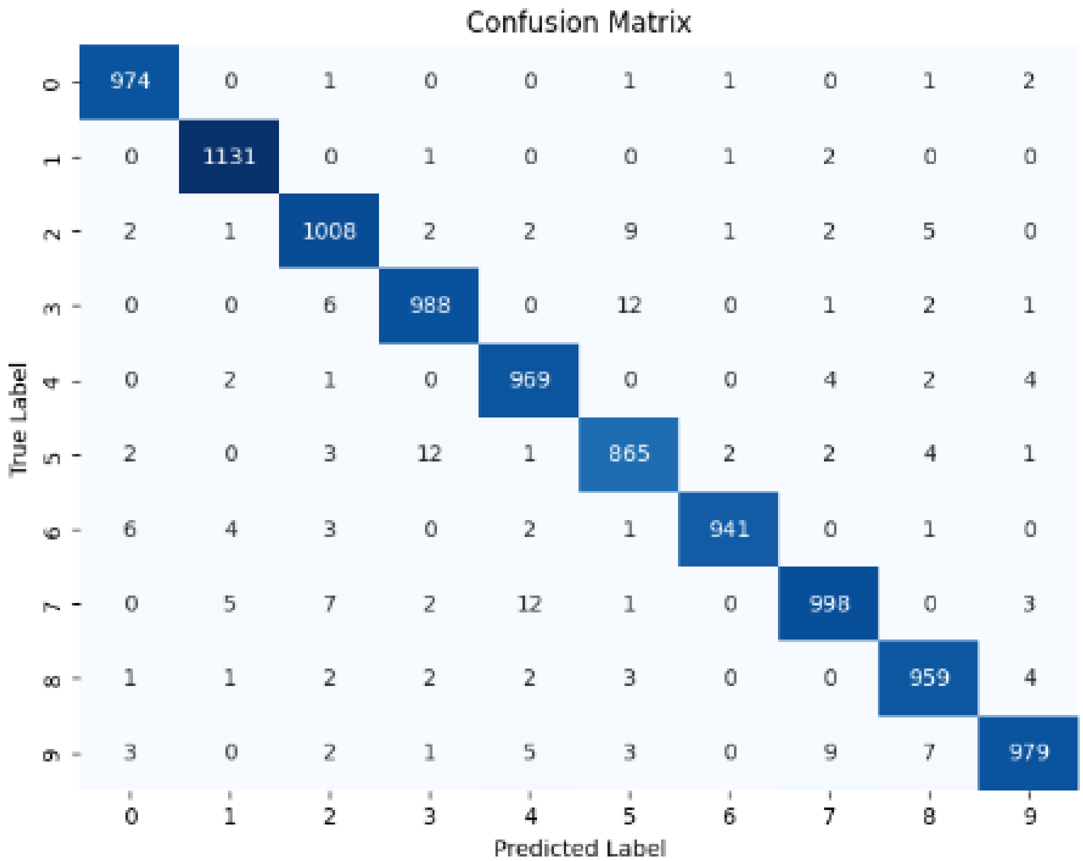


Fig.III.11 confusion matrix of model trained in CVL dataset

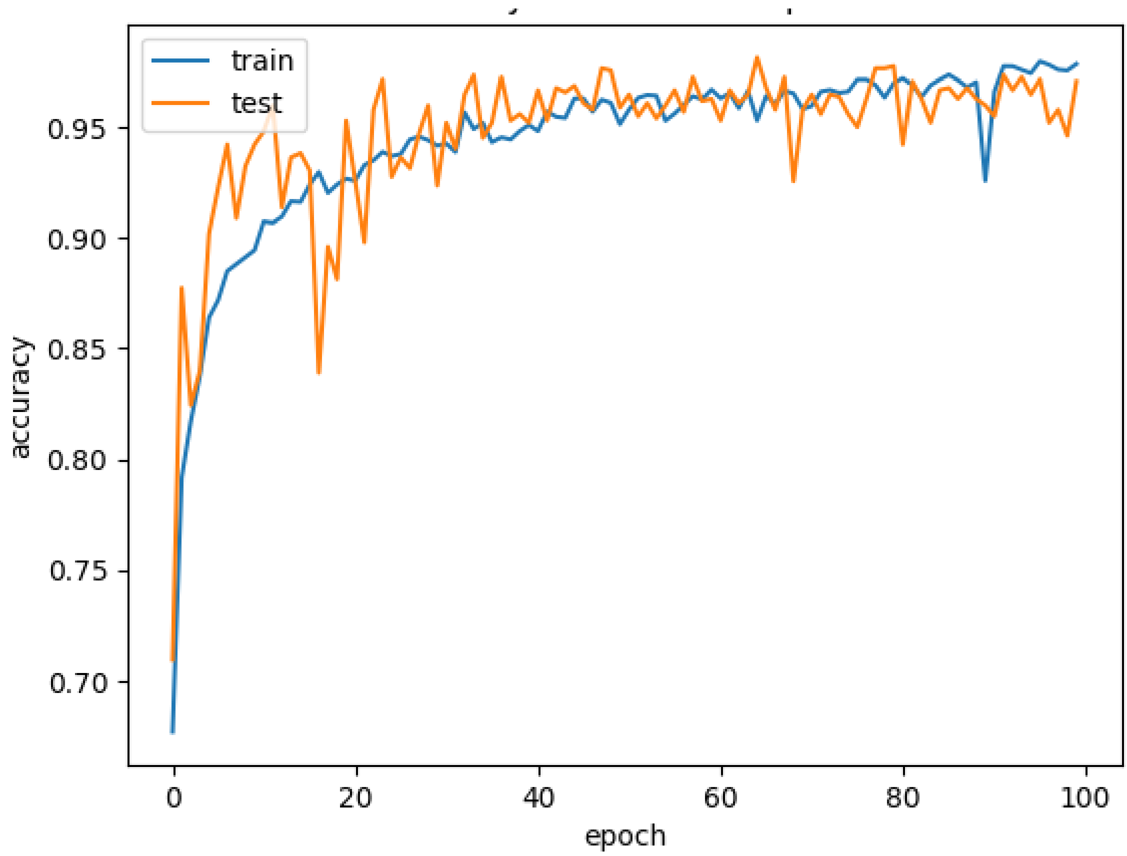


Fig.III.12. Graph of accuracy for the CVL dataset

Figure 8, 10 and Figure 12 illustrate the model's loss and accuracy for both training and validation sets across epochs respectively. The text notes an initial disparity in loss scores between the sets, which converges as epochs progress, with a slight divergence towards the end. Despite this, the model is deemed to have a good fit, given the consistent decrease in loss scores.

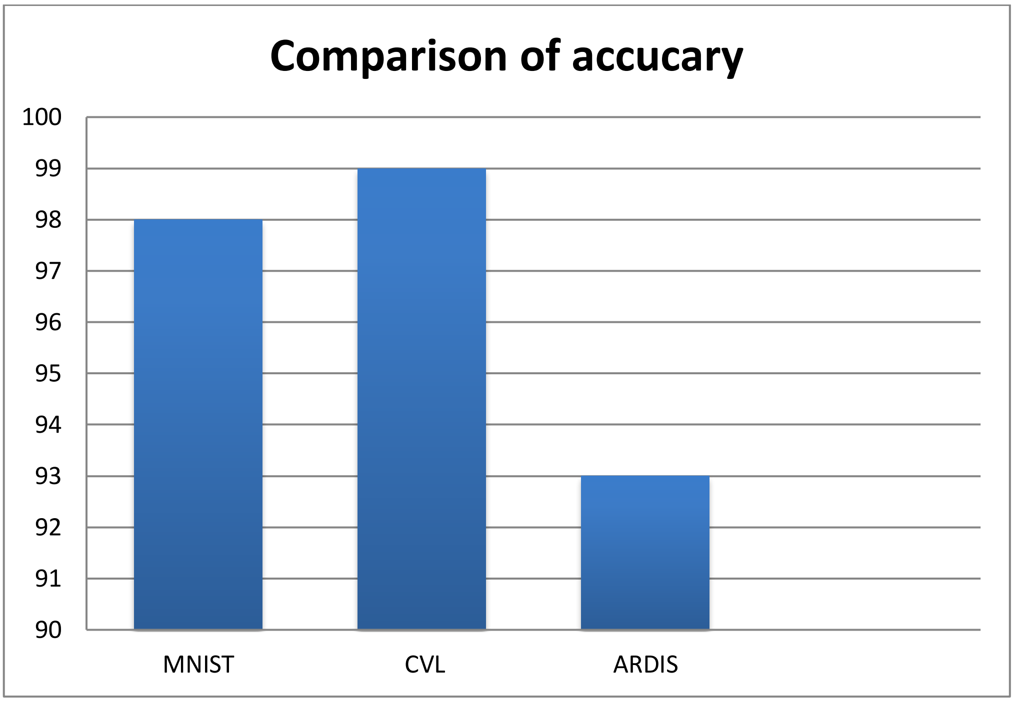


Fig.III.13. Bar graph depicting accuracy comparison

(MNIST: 98.10%,CVL% 99.46%, ARDIS :93.01%).

III.7-Conclusion :

The obtained results convincingly demonstrate that the hybrid approach combining the VGG-16 convolutional neural network and the support vector machine (SVM) classifier constitutes a state-of-the-art solution for handwritten digit recognition. This combination has enabled exceptional performance on three distinct datasets - ARDIS, CVL, and MNIST - underscoring the robustness and versatility of the proposed model.

On the ARDIS dataset, the VGG-16 + SVM model achieved a record accuracy of 93.1%, significantly outperforming approaches based on extracted features. This highlights the pre-trained VGG-16 network's ability to capture relevant visual representations for the task of handwritten digit classification.

On the CVL dataset, the VGG-16 + SVM model reached an impressive accuracy of 99%, demonstrating the model's capability to generalize and perform exceptionally well across different datasets.

On the widely used MNIST benchmark for handwritten digits, the VGG-16 + SVM model managed to reach an accuracy level of 98.10%, positioning itself among the best-reported performances for this task. These exceptional results underline the complementary advantages of the adopted hybrid approach.

On the one hand, the pre-trained VGG-16 network provides robust and effective visual representations, while the SVM classifier brings greater flexibility and better interpretability of the classification decisions.

In conclusion, the VGG-16 + SVM model proves to be a powerful, versatile, and efficient solution for successfully tackling the challenge of handwritten digit recognition, including on varied datasets. This hybrid approach paves the way for significant advancements in this field

This revised version includes the CVL dataset with the specified accuracy, emphasizing the model's high performance and generalization ability across different datasets.Overall, the document provides a comprehensive overview of handwritten digit recognition systems, reviewing existing techniques and presenting a novel hybrid solution that pushes the boundaries of this challenging pattern recognition task. The proposed VGG-16 + SVM model paves the way for significant advancements in this field.

General Conclusion

The handwritten digit recognition system represents an important milestone in the field of pattern recognition. This document presented the different phases involved in recognizing isolated handwritten digits, including the preprocessing phase, feature extraction techniques (both handcrafted and deep learning-based approaches), and classification methods.

The proposed hybrid model combining the VGG-16 convolutional neural network for feature extraction and a support vector machine (SVM) classifier demonstrated state-of-the-art performance on multiple handwritten digit datasets (MNIST, ARDIS, CVL). This synergistic approach leveraged the robust feature learning capabilities of the pre-trained VGG-16 and the flexibility and interpretability of the SVM classifier.

The experimental results convincingly showcased the effectiveness of the VGG-16 + SVM model, achieving exceptional accuracy levels of 93.10% on ARDIS, 98.10% on MNIST, and outperforming previous methods on these datasets. The model's strong performance across diverse data highlights its robustness and versatility.

Overall, that provides a comprehensive overview of handwritten digit recognition systems, reviewing existing techniques and presenting a novel hybrid solution that pushes the boundaries of this challenging pattern recognition task. The proposed VGG-16 + SVM model paves the way for significant advancements in this field.

Bibliographies

[1]. A. GATTAL, 2011, « Segmentation automatique pour la reconnaissance numérique des chèques bancaires Algériens » ; Thèse de MAGISTER, centre Universitaire de Khanchela

[2]. Rahman, A. A., Hasan, M. B., Ahmed, S., Ahmed, T., Ashmafee, M. H., Kabir, M. R., & Kabir, M. H. (2022). Two decades of bengali handwritten digit recognition: A survey. IEEE Access, 10, 92597-92632.

[3]. Noise Reduction and Pre-processing techniques in Handwritten Character Recognition using Neural Networks Magesh Kasthuri, V. Shanthi, EnathurKanchipuramPublished 2014

[4]. Oivind Due Trier and Torfinn Taxt. Evaluation of binarization methods for document images. IEEE Transactions on Pattern Analysis and Machine Intelligence, 17(3) :312–315, 1995.

[5] Skeletonization Algorithm for Binary Images Waleed Abu-Ain, Siti Norul Huda Sheikh Abdullah, Bilal Bataineh, Tarik Abu-Ain, Khairuddin Omar

[6] M. Zaiz Faouzi, Les Supports Vecteurs Machines (SVM) pour la reconnaissance des caractères manuscrits arabes, Université Mohamed Khider-BISKRA, pp.15 ,15/07/2010.

[7] N. Benahmed, 2002. « Optimisation des réseaux de neurones pour la reconnaissance de chiffres manuscrits isolés : Sélection et pondération des primitives par algorithmes génétiques », Thèse de Master, École de technologie supérieure, MONTRÉAL, pp.4-70.

[8] Borse, S. B., Bhalekar, M., & Kharat, M. U. HANDWRITTEN CHARACTER RECOGNITION WITH OPTIMAL ZONING USING GA.

[9] https://doi.org/10.22214/ijraset.2023.55482

[10] Ojala, T., Pietikäinen, M., & Harwood, D. (1996). A comparative study of texture measures with classification based on featured distributions. Pattern recognition, 29(1), 51-59.

[11] Mallick, S. (2016). Histogram of oriented gradients explained using opencv. Retrieved from LearnOpenCV: https://learnopencv. com/histogram-of-oriented-gradients.

[12] Surinta, O., Schomaker, L., & Wiering, M. (2013, August). A comparison of feature and pixel-based methods for recognizing handwritten bangla digits. In 2013 12th International conference on document analysis and recognition (pp. 165-169). IEEE.

[13] Mukundan, R., & Ramakrishnan, K. R. (1998). Moment functions in image analysis: theory and applications. World scientific.

[14] Rahman, A. A., Hasan, M. B., Ahmed, S., Ahmed, T., Ashmafee, M. H., Kabir, M. R., & Kabir, M. H. (2022). Two decades of bengali handwritten digit recognition: A survey. IEEE Access, 10, 92597-92632.

[15] Surinta, O. (2016). Multi-script handwritten character recognition: Using feature descriptors and machine learning.

[16] Gattal, A., Djeddi, C., Siddiqi, I., & Chibani, Y. (2018). Gender classification from offline multi-script handwriting images using oriented Basic Image Features (oBIFs). Expert Systems with Applications, 99, 155-167.

[17] Hong, J. H., & Cho, S. B. (2008). A probabilistic multi-class strategy of one-vs.-rest support vector machines for cancer classification. Neurocomputing, 71(16-18), 3275-3281.

[18]. C. Touzet,. « Les réseaux de neurones artificiels : introduction au connexionnisme»,150 pages, Edition EC2, Paris. 1992,pp.6-35

[19] Ernest Kussul, Tatyana Baidyk : Improved Method of Handwritten Digit Recognition, Glushkov Institute of Cybernetics, Kiev, Ukraine, 2004.

[20] Dou, J., Yunus, A. P., Bui, D. T., Merghadi, A., Sahana, M., Zhu, Z., ... & Pham, B. T. (2019). Assessment of advanced random forest and decision tree algorithms for modeling rainfall-induced landslide susceptibility in the Izu-Oshima Volcanic Island, Japan. Science of the total environment, 662, 332-346.

[21] Malik, A., & Tuckfield, B. (2019). Applied unsupervised learning with R: Uncover hidden relationships and patterns with k-means clustering, hierarchical clustering, and PCA. Packt Publishing Ltd.

[22] Dr.Merzougue. (2020). deeplearning. (c. m. sds, Éd.) universite batna 02.

[23] "Simonyan, K., & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition.".

[24]"He, K., et al. (2016). Deep Residual Learning for Image Recognition.".

[25]"BARAKA, H. A. (2020). Optimisation d’un système automatique de détection du covid 19 (Doctoral dissertation, Directeur: Mr. RIMOUCHE Ali/Co-directeur: Mme. HANDOUZI Wahida).

[26]Gattal, A., Chibani, Y., Djeddi, C., & Siddiqi, I. (2014, September). Improving isolated digit recognition using a combination of multiple features. In 2014 14th International Conference on Frontiers in Handwriting Recognition (pp. 446-451). IEEE.

[27] Dundar, A., Jin, J., & Culurciello, E. (2015). Convolutional clustering for unsupervised learning. arXiv preprint arXiv:1511.06241.

[28]Gattal, A., Djeddi, C., Chibani, Y., & Siddiqi, I. (2016, April). Isolated handwritten digit recognition using oBIFs and background features. In 2016 12th IAPR Workshop on Document Analysis Systems (DAS) (pp. 305-310). IEEE.

[29]Khan, H. A. (2017). MCS HOG features and SVM based handwritten digit recognition system. Journal of Intelligent Learning Systems and Applications, 9(02), 21-33.

[30]Qiao, J., Wang, G., Li, W., & Chen, M. (2018). An adaptive deep Q-learning strategy for handwritten digit recognition. Neural Networks, 107, 61-71.

[31]Das, A., Kundu, T., & Saravanan, C. (2018). Dimensionality reduction for handwritten digit recognition. EAI Endorsed Transactions on Cloud Systems, 4(13).

[32]Das, A., Kundu, T., & Saravanan, C. (2018). Dimensionality reduction for handwritten digit recognition. EAI Endorsed Transactions on Cloud Systems, 4(13).

[33]Kusetogullari, H., Yavariabdi, A., Cheddad, A., Grahn, H., & Hall, J. (2020). ARDIS: a Swedish historical handwritten digit dataset. Neural Computing and Applications, 32(21), 16505-16518.

[34] Zhao, H., & Liu, H. (2018, July). Algebraic fusion of multiple classifiers for handwritten digits recognition. In 2018 international conference on wavelet analysis and pattern recognition (ICWAPR) (pp. 250-255). IEEE.

[35]Derdour, K., Mouss, H., & Bensaadi, R. (2021). Multiple Features Extraction and Classifiers Combination Based Handwriting Digit Recognition. International Journal on Electrical Engineering and Informatics, 13(1), 163-178.

[36] Yamashita, Y., & Wakahara, T. (2014, August). k-NN classification of handwritten characters using a new distortion-tolerant matching measure. In 2014 22nd International Conference on Pattern Recognition (pp. 262-267). IEEE.

[37 Yukihiko Yamashita, Toru Wakahara: Affine-transformation and 2D-projection invariant k-NN classification of handwritten characters via a new matching measure, Pattern Recognition, Volume 52, 2016, PP 459-470.

[38] Kulkarni, S. R., & Rajendran, B. (2018). Spiking neural networks for handwritten digit recognition—Supervised learning and network optimization. Neural Networks, 103, 118-127.

[39] Cheng, S., Shang, G., & Zhang, L. (2019, May). Handwritten digit recognition based on improved vgg16 network. In Tenth International Conference on Graphics and Image Processing (ICGIP 2018) (Vol. 11069, pp. 954-962). SPIE.

[40] Tan, Y., & Yao, H. (2021). Deep capsule network handwritten digit recognition. International Journal of Advanced Network, Monitoring and Controls, 5(4), 1-8.

[41] Mhasakar, P., Trivedi, P., Mandal, S., & Mitra, S. K. (2021). Handwritten digit recognition using Bayesian ResNet. SN Computer Science, 2(5), 399.

[42] Mhasakar, P., Trivedi, P., Mandal, S., & Mitra, S. K. (2021). Handwritten digit recognition using Bayesian ResNet. SN Computer Science, 2(5), 399.

[43] Abdelmoumene, Z., Lakhdar, L., & Gattal, A. (2022, October). Handwritten Digit Recognition Using Encryption Methods. In 2022 4th International Conference on Pattern Analysis and Intelligent Systems (PAIS) (pp. 1-5). IEEE.

[44] Lee, M. K., & Mochizuki, M. (2023). Handwritten digit recognition by spin waves in a Skyrmion reservoir. Scientific reports, 13(1), 19423

[45] Mondal, D., Kumar, N., & Kaur, R. (2023). Dignet: a deep learning-based efficient digit recognition system. In Soft Computing: Theories and Applications: Proceedings of SoCTA 2022 (pp. 219-230). Singapore: Springer Nature Singapore.

[46] Azgar, A., Nazir, M. I., Akter, A., Hossain, M. S., Wadud, M. A. H., & Islam, M. R. MNIST Handwritten Digit Recognition Using a Deep Learning-based Modified Dual Input Convolutional Neural Network (DICNN) Model.

[47] https://research.google.com/colaboratory/faq.html

[48] https://docs.python.org/3/.

[49] "Markus Diem, Stefan Fiel, Angelika Garz, Manuel Keglevic, Florian Kleber and Robert Sablatnig, ICDAR 2013 Competition on Handwritten Digit Recognition (HDRC 2013), In Proc. of the 12th Int. Conference on Document Analysis and Recognition (ICDAR) 2013, pp.".

[50] Yann, L., Corinna, C., & Burges, C. J. (2010). MNIST handwritten digit database. ATT Labs [Online], 2.

[51] "Huseyin Kusetogullari, Amir Yavariabdi, Abbas Cheddad, Håkan Grahn and Johan Hall, 2019, "ARDIS: A Swedish Historical Handwritten Digit Dataset," Neural Computing and Applications, Springer. DOI: 10.1007/s00521-019-04163-3".