الجـــــــــمهوريـــــــة الجـــــــزائريـــــة الديمـــــــــقراطية الشعبيـــــة

PEOPLE’S DEMOCRATIC REPUBLIC OF ALGERIA

وزارة التـــــــعليم العـــــــالي والبــــــــحث العـــــــلمي

Ministry of Higher Education and Scientific Research

جـــــــامعة الــــــــوادي

University of El Oued

كليـــــة التكنـــــــولوجيا

Faculty of Technology

قسم الهنــــــــدسة الكهربــــــــائيــة

Department of Electrical Engineering

Dissertation

Presented for the degree of **Master**

In: Electrical engineering **Specialty**: Telecommunication Systems

By:

BAATOUT Ali Awab
BENABDALLAH Abderrahmen Taha
BOUAFIA Djamel

THEME

Image Classification using Deep Learning

Jury members

Mr. KHALIL Abdelatif	MCA	Univ. El Oued	President
Mr. GHENDIR Said	MCA	Univ. El Oued	Examiner
Mrs. TIDJANI Amina	MCB	Univ. El Oued	Supervisor

College year: 2022/2023

بسم الله الرحمن الرحيم

اهداء

الى من أُفضِّلها على نفسي، ولِمَ لا؛ فلقد ضحَّت من أجلي

ولم تدَّخر جُهدًا في سبيل إسعادي على الدَّوام (أُمِّي الحبيبة).

نسير في دروب الحياة، ويبقى من يُسيطر على أذهاننا في كل مسلك نسلكه

صاحب الوجه الطيب، والأفعال الحسنة .فلم يبخل عليَّ طيلة حياته (والدي العزيز)

إلى أصدقائي، وجميع من وقفوا بجواري وساعدوني بكل ما يملكون، وفي أصعدة كثيرة

الى زملائي في هذا البحث شكرا لعملكم الشاق طوال فترة التحضير

أُقدِّم لكم هذا البحث، وأتمنَّى أن يحوز على رضاكم

BENABDALLAH Abderrahmen Taha

Dedication

I dedicate my thesis to my family. A special feeling of gratitude to my loving parents whose words of encouragement and push for tenacity ring in my ears. My sisters and brother have never left my side and are very special.

I also dedicate this dissertation to my many friends who have supported me throughout the process. I will always appreciate all they have done.

BAATOUT Ali Awab

اهداء

الحمد لله رب العالمين والصلاة والسلام على أشرف المرسلين وخاتم النبيئين محمد رسول الله صل الله عليه وسلم وأما بعد:

اهدي هذا العمل المتواضع الى من أحمل اسمه بكل فخر واعتزاز ابي الغالي رحمة الله عليه. الى نبع الحنان الغالية أمي أطال الله في عمرك الى جميع الأهل والأقارب والأحباب.

الى جميع الأساتذة جزاهم الله عني خير الجزاء. الى جميع الأصدقاء في الدراسة من المرحلة الابتدائية الى المرحلة الجامعية وخاصة دفعة 2023 شعبة أنظمة اتصالات.

الى كل انسان طيب قابلته في حياتي الى من نسيه القلم ولم ينسه القلب

اهدي هذا العمل المتواضع.

BOUAFIA Djamel

Acknoledgment

"Words spoken disappear into the air like a mist, but words written leave footprints on the page "

TIDJANI Amina *has been an ideal teacher, mentor, and supervisor, offering advice and encouragement with a perfect blend of insight and humor. We're proud of, and grateful for, our time working with her.*

Abstract:

Deep learning (DL) is a sub-domain of machine learning (ML), it consists of artificial neural networks. The word (deep) refers to the number of layers in the network, the more layers there are, the deeper the network. Advanced tools and techniques have significantly improved Deep Learning algorithms to the point where they can outperform humans in classifying images. Among artificial neural network architectures there are convolutional neural networks (CNN). Image classification is a classic problem in the fields of image processing and convolutional neural networks. Our work studies the images classification by using an architecture of a convolutional neural network. Intelligence and has the potential to revolutionize many industries.

Key Words: Image Classification, Deep Learning (DL), Machine Learning (ML), Convolutional Neural Network (CNN), Webcam, Transfer Learning.

Résumé :

L'apprentissage profond (DL) est un sous-domaine de l'apprentissage automatique (ML). Il se compose de réseaux de neurones artificiels. Le mot "profond" fait référence au nombre de couches dans le réseau, plus il y a de couches, plus le réseau est profond. Des outils et des techniques avancés ont considérablement amélioré les algorithmes d'apprentissage profond au point qu'ils peuvent surpasser les humains dans la classification des images. Parmi les architectures de réseaux de neurones artificiels, on trouve les réseaux de neurones convolutifs (CNN). La classification d'images est un problème classique dans les domaines du traitement d'images et des réseaux de neurones convolutifs. Notre travail étudie la classification d'images en utilisant une architecture de réseau de neurones convolutifs. L'intelligence artificielle a le potentiel de révolutionner de nombreuses industries.

Mots clés : Classification d'images, Deep Learning (DL), Machine Learning (ML), réseau neuronal convolutif (CNN), webcam,Tranfer Learning

الملخص:

التعلم العميق (DL) هو مجال فرعي للتعلم الآلي(ML) ، ويتكون من شبكات عصبية اصطناعية. تشير كلمة (عميق) إلى عدد الطبقات في الشبكة، وكلما زاد عدد الطبقات، زادت عمق الشبكة. لقد حسنت الأدوات والتقنيات المتقدمة بشكل كبير خوارزميات التعلم العميق لدرجة أنها يمكن أن تتفوق على البشر في تصنيف الصور. من بين معماريات الشبكات العصبية الاصطناعية هناك شبكات عصبية ملتفة .(CNN) تصنيف الصور هو مشكلة كلاسيكية في مجالات معالجة الصور والشبكات العصبية الملتفة. يدرس عملنا تصنيف الصور باستخدام بنية شبكة عصبية ملتفة. الذكاء ولديه القدرة على إحداث ثورة في العديد من الصناعات.

الكلمات المفتاحية: تصنيف الصور، التعلم العميق (DL)، تعلم الألة (ML)، الشبكة العصبية الملتفة (CNN)، كاميرا ويب , نقل التعلم .

Content

Chapter 3: Modeling Image Classification Through Simulation

Figures List

List of Tables

Introduction

Advancements in technology, such as the availability of large amounts of data and improved processing power, have made it possible for artificial intelligence systems to store, analyze, index, and search through vast amounts of information in today's data-driven society. One branch of artificial intelligence called machine learning focuses on creating algorithms that can learn from data. Classification, which involves grouping data into different categories based on their characteristics, is a powerful method to address these challenges.

In the field of computer vision, deep learning has emerged as a powerful technique for image classification. Deep learning involves building models called convolutional neural networks (CNNs) that are adept at extracting important features from images and making accurate classifications. CNNs have achieved remarkable results in tasks such as image recognition, object detection, and image segmentation, revolutionizing the field of computer vision.

This work primarily focuses on supervised learning problems, with an emphasis on deep learning. It explores the challenges associated with dealing with large datasets and highlights the importance of structuring and searching the data. Classification is a valuable tool in mitigating these challenges, particularly when it comes to image datasets. By categorizing images into different classes (e.g., animals, humans, transportation), classification enables more effective utilization of these datasets.

The goal of image classification is to narrow the gap between computer vision and human vision by training computers to understand and interpret visual data. The use of deep learning, a subset of machine learning, has been instrumental in achieving this goal. It involves constructing CNN models that excel at extracting relevant features from images and performing accurate classifications.

The manuscript is divided into three main chapters to provide a comprehensive understanding of the topic:

- ❖ **The first chapter** introduces machine learning and its different types, followed by an explanation of the basics of image classification.
- ❖ **The second chapter** focuses on neural networks, describing their various types, the architecture of CNNs, and different activation functions.
- ❖ **The third chapter** presents the simulation part of the work, including the tools and software used, the dataset employed, and the results obtained.

By following this structure, the manuscript aims to provide readers with a clear and thorough understanding of the subject matter.

Chapter I

Image classification and Machine learning

Chapter I: Image Classification and Machine Learning

1. Introduction:

This chapter highlights and introduces the concepts and basics of machine learning, we will talk briefly about the importance of machine learning (ML), and then we will go through its different types. In the second part of the chapter, we will focus on the classification of images which is the main subject of this research.

2. Machine learning:

2.1. What is machine learning:

Machine learning is a type of artificial intelligence that allows computer systems to automatically improve their performance on a specific task, without being explicitly programmed. It involves training algorithms on data and using that learning to make predictions or decisions without being explicitly told how to perform the task.[1]

2.2. Types of machine learning:

Finding patterns in the database without any human interventions or actions is based upon the data type, i.e., labeled or unlabeled and based upon the techniques used for training the model on a given dataset. Machine learning is further classified as Supervised, Unsupervised, Learning algorithms; both of these types of learning techniques are used in different applications.

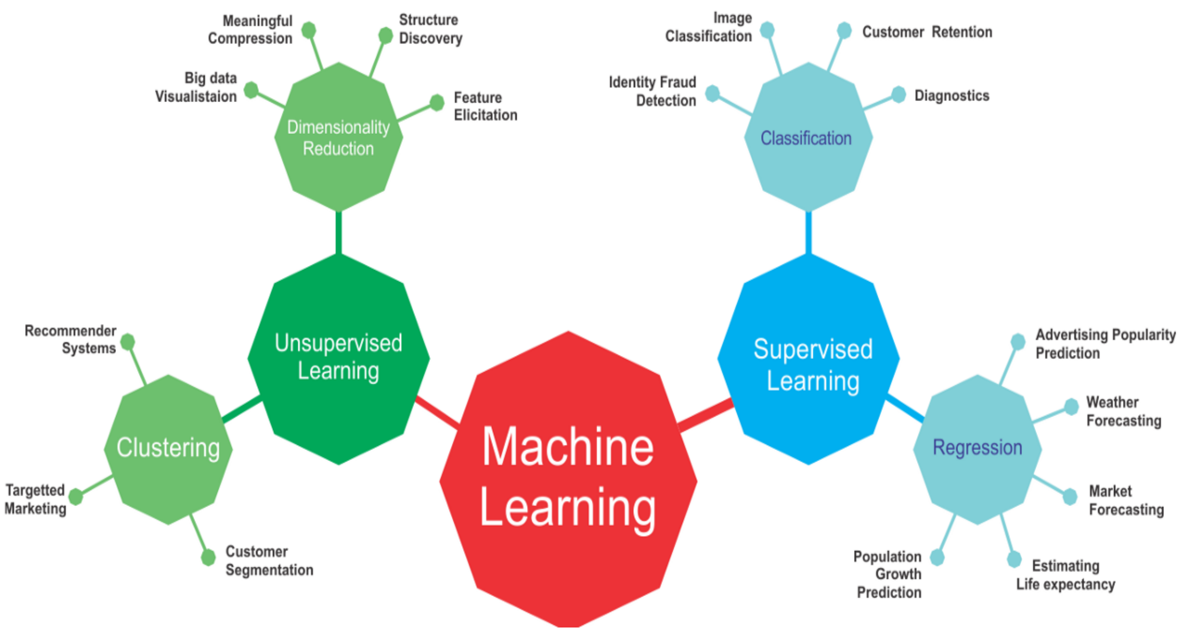


Figure I.1: Machine Learning subcategories [2]

2.2.1 Supervised machine learning:

Supervised learning algorithms are used when the output is classified or labeled. These algorithms learn from the past data that is inputted, called training data, runs its analysis and uses this analysis to predict future events of any new data within the known classifications. The accurate prediction of test data requires large data to have a sufficient understanding of the patterns. The algorithm can be trained further by comparing the training outputs to actual ones and using the errors for modification of the algorithms.[3]

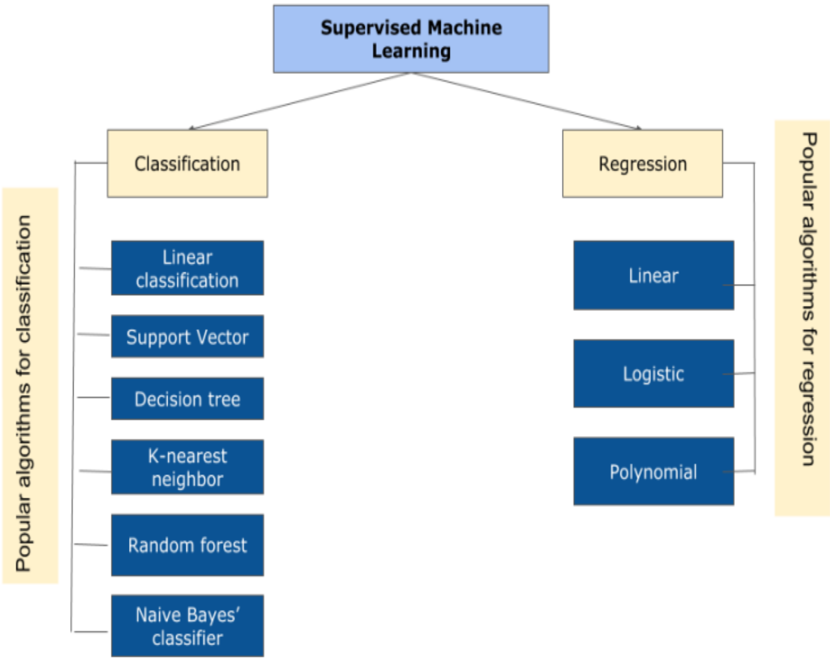


Figure I.2: Algorithms of supervised machine learning [4]

2.2.2 Unsupervised machine learning:

Unsupervised machine learning is a type of machine learning where the algorithm is trained on an unlabeled dataset and tries to find patterns, structure, and relationships within the data without any prior knowledge or guidance. Unlike supervised learning, unsupervised learning does not have a defined target variable or label that the algorithm is trying to predict.

Unsupervised learning algorithms can be used for various tasks, such as dimensionality reduction, clustering, anomaly detection, and data visualization.

Clustering, for example, is an unsupervised learning task that involves grouping similar instances together into clusters. The algorithm tries to find the optimal number of clusters and assign each instance to the most appropriate cluster.

Dimensionality reduction is another unsupervised learning task that involves reducing the number of features in a dataset, while preserving as much information as possible. This can be useful for visualizing high-dimensional data, as well as for reducing computational costs and improving the performance of supervised learning algorithms.

Unsupervised learning algorithms include techniques such as K-Means, Hierarchical Clustering, Principal Component Analysis (PCA), and Autoencoders. These algorithms have a wide range of applications, including market segmentation, customer behavior analysis, and image compression.[3]

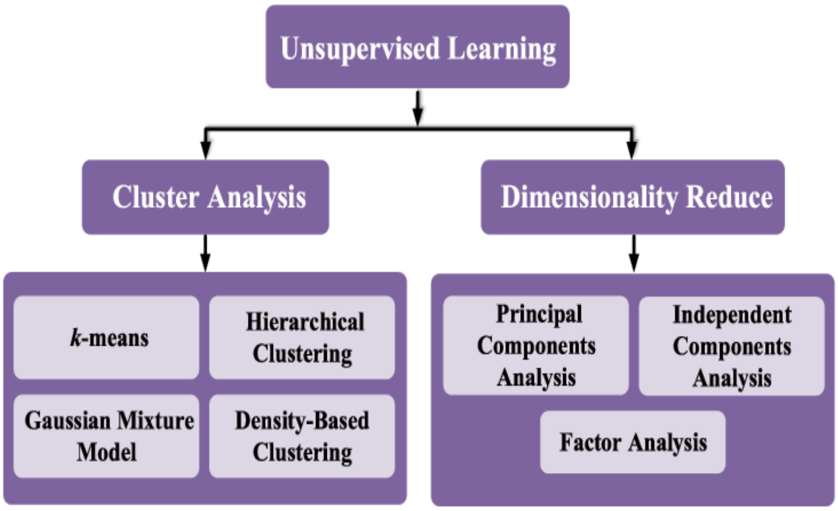


Figure I.3: Algorithms of unsupervised machine learning [5]

3. Image Classification:

3.1.Classification Definition:

Image classification is a process in which a computer algorithm is trained to recognize and distinguish various objects, scenes, and actions within images. This is done by providing the algorithm with a dataset of labeled images, where each image is assigned a specific class or label, such as "dog" or "cat" the algorithm then learns to identify patterns and features within the images that are characteristic of each class. Once trained, the algorithm can then be used to classify new, unseen images, assigning them a class based on its learned patterns and features. This technology is widely used in areas such as self-driving cars, medical imaging, and security systems.[6]

3.2.Image classification system:

Building an image classification system involves several steps, which can be summarized as follows:

3.2.1 Data Set: An image dataset is a collection of digital images that are organized and used for various purposes, such as computer vision tasks, image processing, pattern recognition, or machine learning algorithms that deal with visual data. Image datasets are widely used in fields like computer vision, deep learning, and artificial intelligence.

Image datasets can include images of different formats, resolutions, and sizes. They can consist of various types of images, such as photographs, illustrations, medical images, satellite imagery, or any other visual content that can be represented digitally. Image datasets often have a specific theme or purpose, such as face recognition, object detection, image classification, or semantic segmentation.

Image datasets can be created through different means, including manual collection, scraping from the web, capturing images using sensors or cameras, or combining existing datasets. They can be publicly available, shared by research institutions or organizations, or created and used for specific research projects or applications.

Examples of popular image datasets include ImageNet, COCO (Common Objects in Context), CIFAR-10, and MNIST (Modified National Institute of Standards and Technology) dataset. These datasets have been widely used for training and evaluating various computer vision algorithms and models. [7]

3.2.2 Features extraction: Is a process in which meaningful features are extracted from raw data, such as images, so that they can be used as input for a machine learning algorithm, such as an image classification algorithm. In image classification, feature extraction involves transforming the raw pixel values of an image into a feature vector that represents the image in a numerical format that can be used by the algorithm.

There are several approaches to feature extraction in image classification, including:

- ❖ **Hand-crafted Features**: Hand-crafted features are pre-defined features that are manually designed by domain experts. These features are typically based on visual characteristics such as texture, shape, color, and edge orientation, and are extracted using techniques such as Histogram of Oriented Gradients (HOG), Local Binary Patterns (LBP), and Scale-Invariant Feature Transform (SIFT).

- **Deep Learning Features**: Deep learning-based feature extraction involves using pre-trained convolutional neural networks (CNNs) such as VGG, ResNet, or AlexNet. The pre-trained

  CNNs have already learned a set of features that are useful for a wide range of image classification tasks, and these features can be used directly or fine-tuned for a specific task.
- **Hybrid Approaches**: Hybrid approaches involve combining hand-crafted features with deep learning-based features to extract more informative features from the images. For example, the hand-crafted features can be concatenated with the output of a pre-trained CNN to form a hybrid feature vector.

Once the features are extracted, they are typically normalized and transformed into a fixed-length feature vector that can be used as input to a machine learning algorithm, such as a support vector machine (SVM), decision tree, or neural network, to train a model for image classification. [8]

3.2.3 Model Classification:

Model Classification: is a task that involves categorizing data into predefined classes or categories. It uses patterns and features within the data to train a model, which can then predict the class of new, unseen instances with a certain level of accuracy.

- **K-Nearest Neighbor:**

k-Nearest Neighbor (k-NN) is a simple machine learning algorithm used for classification and regression tasks. In image classification, the k-NN algorithm works by finding the k closest data points to a new, unknown image, based on the features of the image. These data points, or "neighbors," are then used to determine the class label of the unknown image by majority voting, where the class with the most votes among the k neighbors is chosen as the prediction.

In k-NN, the distance metric between instances is an important factor. Commonly used distance metrics are Euclidean distance, Manhattan distance, or Minkowski distance.

The value of k, the number of nearest neighbors, can be chosen through cross-validation or by using a heuristic approach. When k = 1, the algorithm is known as the nearest neighbor algorithm, and the prediction is based on the single closest training example. When k is increased, the algorithm becomes more robust to noise and outliers, but the boundaries between classes become smoother.

k-NN is a simple and intuitive algorithm that is easy to understand and implement. It can be used for both binary and multi-class classification, as well as for regression tasks. However, k-NN can be computationally expensive when the number of instances and features is large, as it requires a large amount of memory to store the entire dataset, and a large amount of time to compute the nearest neighbors for each prediction.[9]

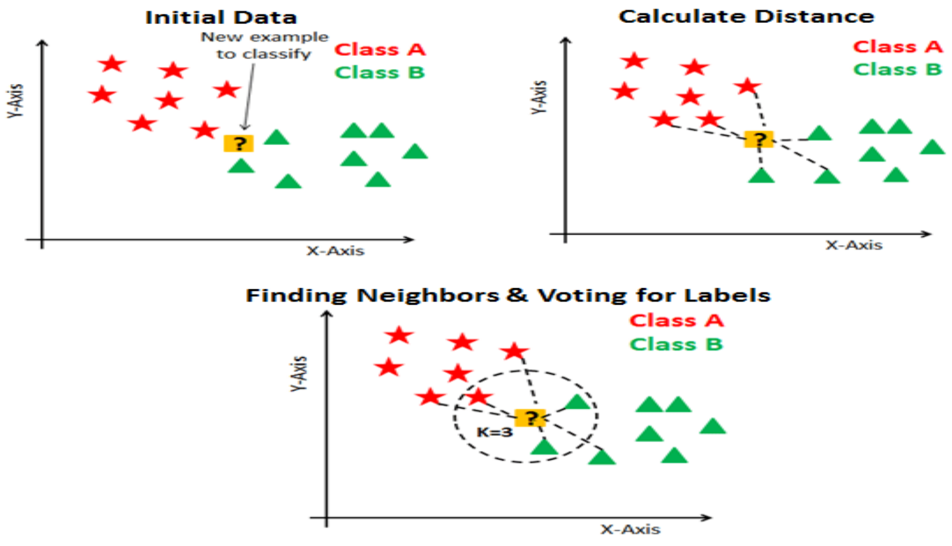


Figure I.4: K-NN algorithms application example [10]

❖ **Fuzzy c-means :**

Fuzzy C-means (FCM) is a variant of the k-means algorithm that allows for soft clustering, where each data point can belong to multiple clusters with varying degrees of membership. This is useful in situations where a data point may not fit neatly into one cluster.

In FCM, the cluster membership of each data point is represented by a membership degree, which is a value between 0 and 1 that indicates the degree of membership of the data point in a particular cluster. The membership degrees for a data point are computed based on the distances between the data point and the cluster centroids.

The FCM algorithm works as follows:

- Initialize the cluster centroids randomly
- Compute the membership degrees for each data point based on the distances between the data point and the cluster centroids.
- Recalculate the cluster centroids based on the weighted average of the data points.
- Repeat steps 2 and 3 until the cluster centroids stop moving or a maximum number of iterations is reached.

FCM is a popular algorithm for clustering data with uncertainty or ambiguity and has been applied in a variety of fields including image processing, pattern recognition, and control systems [11]

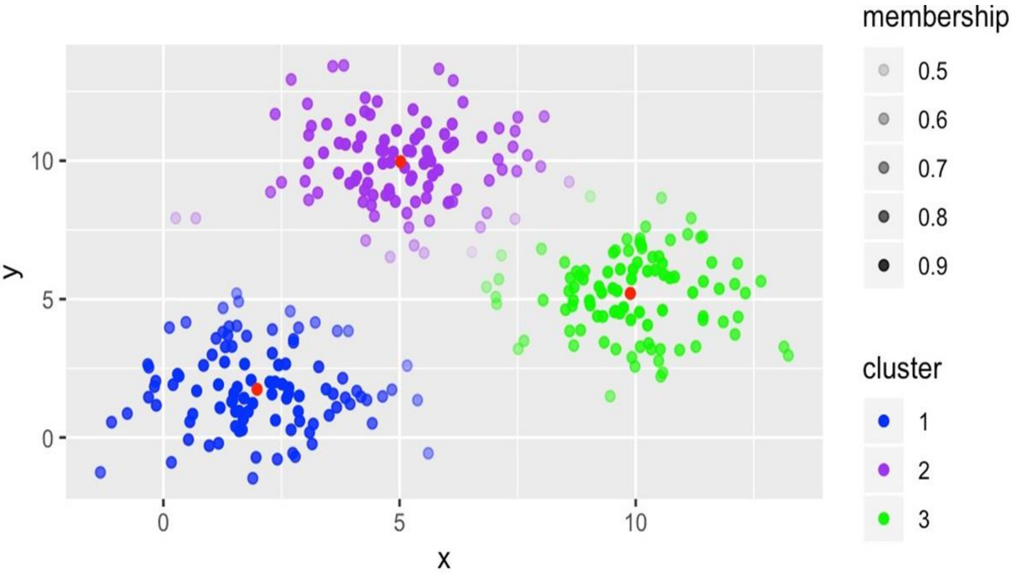


Figure I.5: Fuzzy c-means algorithms application example[12]

- **Support vector machine:**

Support Vector Machine (SVM) is a supervised machine learning algorithm used for both classification and regression. Though we say regression problems as well its best suited for classification. The objective of a Support Vector Machine (SVM) is to find a hyperplane that maximally separates the data into classes. The SVM algorithm searches for the hyperplane with the largest margin between the classes, which means the distance between the closest data points from both classes is maximized. This results in a classifier that is both accurate and robust to overfitting.[13]

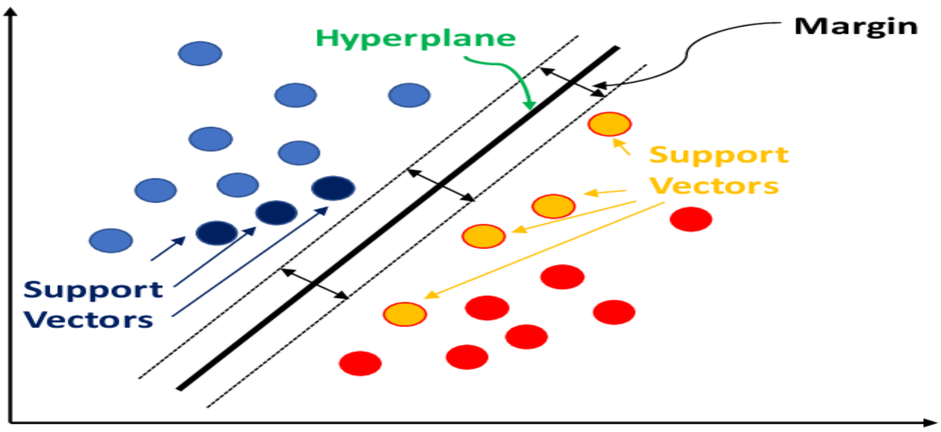


Figure I.6 : SVM's algorithms application Example [14]

- **Convolution Neural Network:** CNNs are a type of deep learning model commonly used for image classification tasks. They are designed to automatically learn and extract features from images through convolutional layers. These layers apply filters to detect local patterns and features, while pooling layers downsample the feature maps. Fully connected layers at the end of the network perform the classification based on the learned features. By leveraging the hierarchical representations learned from data, CNNs have achieved remarkable success in image classification tasks, surpassing traditional methods by a significant margin.[15]

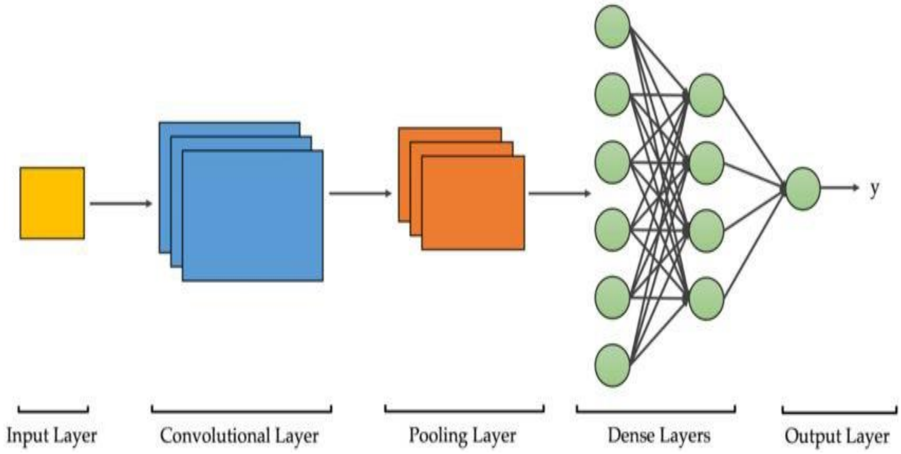


Figure I.7 : CNN model Example [16]

3.2.4 Model evaluation: Is the process of assessing the performance of a machine learning model on a test dataset. The goal of model evaluation is to determine how well the model will perform on new, unseen data. usually evaluated using metrics such as accuracy, precision, recall, F1 score, and receiver operating characteristic (ROC) curves.

- **Accuracy:** Is the most commonly used metric and it is defined as the ratio of correctly classified samples to the total number of samples. For example, in a binary classification problem with two classes, accuracy is defined as follows:

Accuracy = (True Positives + True Negatives) / Total samples

- **Precision**: Measures the proportion of positive predictions that are actually correct. Precision is defined as follows:

Precision = True Positives / (True Positives + False Positives)

- **Recall**: Measures the proportion of actual positive examples that are correctly identified by the model. Recall is defined as follows:

  Recall = True Positives / (True Positives + False Negatives)

- **F1 score**: Is the harmonic mean of precision and recall and it provides a balance between the two metrics. F1 score is defined as follows:

  F1 score = 2 * (Precision * Recall) / (Precision + Recall)

- **ROC curves**: are used to evaluate binary classification models. They plot the true positive rate against the false positive rate at various threshold settings. The area under the ROC curve (AUC) is a common metric used to evaluate the performance of binary classifiers.[7]

3.3.Image classification and machine learning:

Manual methods have proven to be very difficult to apply for seemingly simple tasks such as image classification, object recognition in images, or speech recognition. Data from actual samples of sound or pixels in an image are complex, variable, and noise-tinged. For a machine, an image is an array of numbers indicating the brightness (or color) of each pixel, and an audible signal is a sequence of numbers indicating the air pressure at each moment. It is virtually impossible to write a program that will work robustly in any situation. That's where machine learning comes in. It's learning that drives the systems of all major Internet companies.[17]

They have long used it to filter desirable content, order responses to a search, make recommendations, or select information of interest to each user. A controllable system can be seen as a black box with input, for example an image, sound or text, and an output that can represent the category of the object in the image, spoken word or subject that the text is talking about. These are called classification systems or pattern recognition. In its most widely used form, machine learning is supervised: at the input of the machine is shown an image of an object, for example a car, and it is given the desired output for a car. Then I show him the picture of a dog with the desired exit for a dog. After each example, the machine adjusts its internal parameters to bring its output closer to the desired result. After showing the machine thousands or millions of examples labeled with their category, the machine becomes able to classify most of them correctly. But what is more interesting is that he can also correctly classify images of cars or dogs that he has never seen during the learning phase. This is called the ability to generalize.[18]

4 Conclusion:

In this chapter, we have introduced the common types of machine learning, we have also summarized the types that are supervised, unsupervised machine learning. We also analyzed some image classification algorithms. Finally, we define the relationship between machine learning and image classification.

Traditional machine learning approaches in image classification faced the challenge of manually extracting relevant features. This process required domain knowledge and expert feature engineering to identify discriminative features for distinguishing between different image classes. It was time-consuming and subjective, relying on human expertise. So what can be a solution to this Problem?

Chapter Ⅱ

Deep Learning And Convolutional Neural Network

1. Introduction:

In this chapter, we will provide an overview of Deep Learning and neural networks, with a focus on CNN architecture. We will talk about deep learning concept and its relation with machine learning. Furthermore, we will investigate deeply CNN its layers and main architectures used for image classification. Finally, By the end of this chapter, you should have a solid understanding of the fundamentals of Deep Learning and how it can be applied to image classification tasks.

2. Definition of Deep learning:

Deep learning is a subset of machine learning, which is essentially a neural network with three or more layers. These neural networks attempt to simulate the behavior of the human brain—albeit far from matching its ability—allowing it to "learn" from large amounts of data. While a neural network with a single layer can still make approximate predictions, additional hidden layers can help to optimize and refine for accuracy.

Deep learning drives many artificial intelligence (AI) applications and services that improve automation, performing analytical and physical tasks without human intervention. Deep learning technology lies behind everyday products and services (such as digital assistants, voice-enabled TV remotes, and credit card fraud detection) as well as emerging technologies (such as self-driving cars).[19]

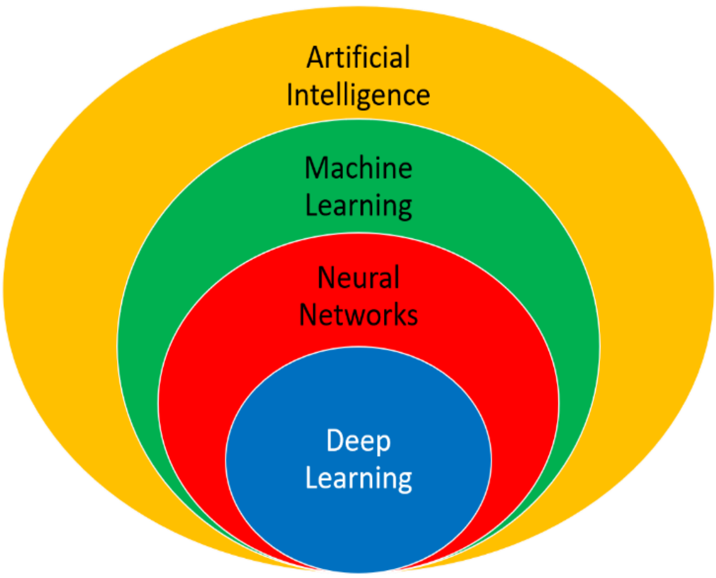


Figure II.1: The relation between AI, Machine learning, Neural networks and Deep learning [20]

That is, machine learning is a subfield of artificial intelligence. Deep learning is a subfield of machine learning, and neural networks make up the backbone of deep learning algorithms. In fact, it is the number of node layers, or depth, of neural networks that distinguishes a single neural network from a deep learning algorithm, which must have more than three.

3. Deep Learning Vs Machine Learning:

If deep learning is a subset of machine learning, how do they differ? Deep learning distinguishes itself from classical machine learning by the type of data that it works with and the methods in which it learns.

Machine learning algorithms leverage structured, labeled data to make predictions—meaning that specific features are defined from the input data for the model and organized into tables. This doesn't necessarily mean that it doesn't use unstructured data; it just means that if it does, it generally goes through some pre-processing to organize it into a structured format.

Deep learning eliminates some of data pre-processing that is typically involved with machine learning. These algorithms can ingest and process unstructured data, like text and images, and it automates feature extraction, removing some of the dependency on human experts. For example, let's say that we had a set of photos of different pets, and we wanted to categorize by "cat", "dog", "hamster", et cetera. Deep learning algorithms can determine which features (e.g. ears) are most important to distinguish each animal from another. In machine learning, this hierarchy of features is established manually by a human expert.

Then, through the processes of gradient descent and backpropagation, the deep learning algorithm adjusts and fits itself for accuracy, allowing it to make predictions about a new photo of an animal with increased precision.

Machine learning and deep learning models are capable of different types of learning as well, which are usually categorized as supervised and unsupervised learning. Supervised learning utilizes labeled datasets to categorize or make predictions; this requires some kind of human intervention to label input data correctly. In contrast, unsupervised learning doesn't require labeled datasets, and instead, it detects patterns in the data, clustering them by any distinguishing characteristics. [21]

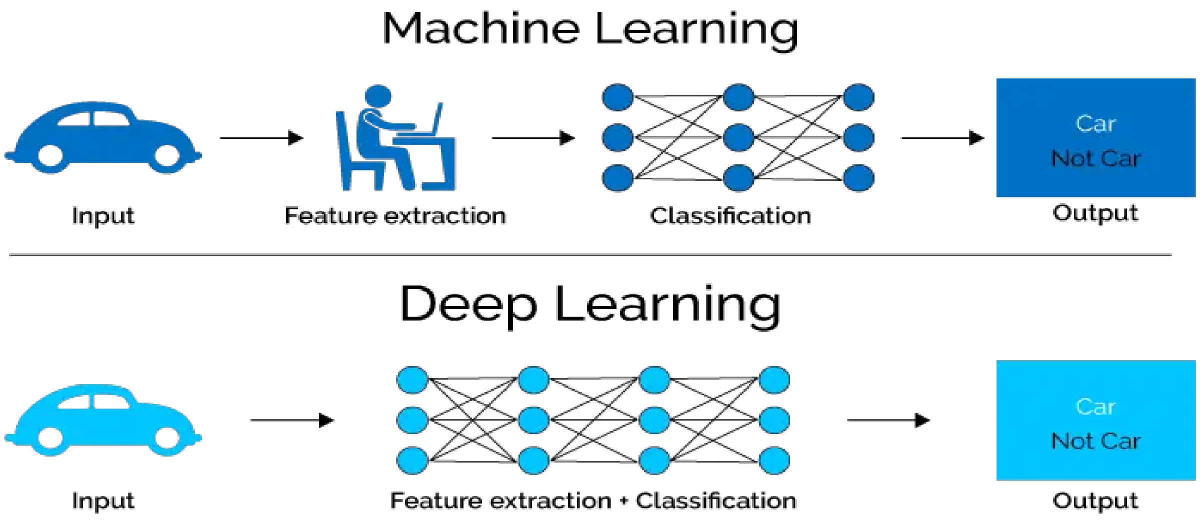


Figure II.2: Machine Learning and Deep Learning Strategies [22]

4. How Deep Learning Works:

Deep learning neural networks, or artificial neural networks, attempts to mimic the human brain through a combination of data inputs, weights, and bias. These elements work together to accurately recognize, classify, and describe objects within the data.

Deep neural networks consist of multiple layers of interconnected nodes, each building upon the previous layer to refine and optimize the prediction or categorization. This progression of computations through the network is called forward propagation. The input and output layers of a deep neural network are called visible layers. The input layer is where the deep learning model ingests the data for processing, and the output layer is where the final prediction or classification is made.

Another process called backpropagation uses algorithms, like gradient descent, to calculate errors in predictions and then adjusts the weights and biases of the function by moving backwards through the layers in an effort to train the model. Together, forward propagation and backpropagation allow a neural network to make predictions and correct for any errors accordingly. Over time, the algorithm becomes gradually more accurate.

The above describes the simplest type of deep neural network in the simplest terms. However, deep learning algorithms are incredibly complex, and there are different types of neural networks to address specific problems or datasets. For example,

- **Convolutional neural networks (CNNs)** used primarily in computer vision and image classification applications, can detect features and patterns within an image, enabling tasks, like object detection or recognition. In 2015, a CNN bested a human in an object recognition challenge for the first time.[23]

5. Neural Networks:

Neural networks are a type of machine learning algorithm inspired by the structure and function of the human brain. They consist of interconnected nodes, or neurons, organized into layers. Each neuron receives input from other neurons, processes that input using an activation function, and produces an output that is passed on to other neurons in the next layer. Neural networks can be trained to learn patterns in data by adjusting the weights and biases of the connections between neurons using algorithms such as backpropagation. They have been successfully applied to a wide range of tasks, including image and speech recognition, natural language processing, and game playing.[24]

5.1.Neural network Types:

5.1.1 Feed-Forward Neural Network :

This is a basic neural network that can exist in the entire domain of neural networks. As the name suggests, the motion of this network is only forward, and it moves till the point it reaches the output node. There is no back feedback to improve the nodes in different layers and not much self-learning mechanism. Below is a simple representation one-layer neural network.[24]

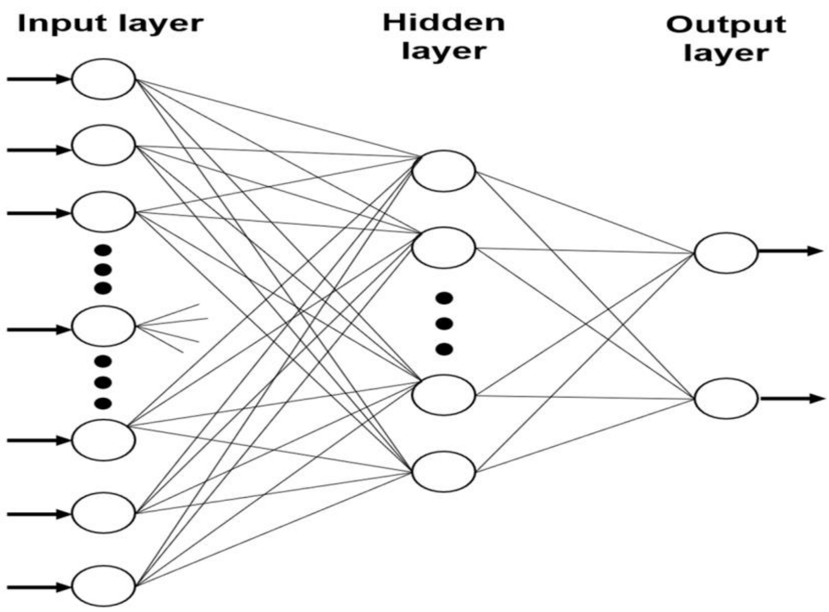


Figure II.3: Feed Forward Neural Network Structure [25]

5.2.1 Multilayer Perceptron:

Now, slowly we would move to neural networks having more than 2 layers, i.e., more than one hidden layer. In a Multilayer Perceptron, the main intuition of using this method is when the data is not linearly separable. Each node in a layer consists of a non-linear activation function for processing. These functions are typically Sigmoid/Logistic Function, tanh/Hyperbolic Tangent function, ReLU (Rectified Linear Unit), Softmax. This neural network is fully connected and also has the capability to learn by itself by changing the weights of connection after each data point is processed and the amount of error it generates.[24]

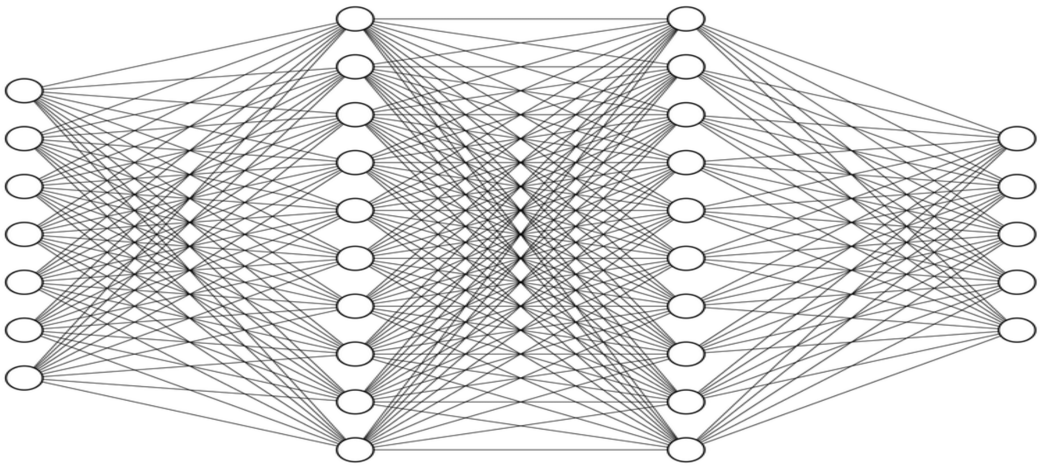


Figure II.4: Multilayer Perceptron Structure [26]

5.2.3 Convolutional neural network:

Now coming on to Convolutional Neural Network, this type of neural network is an advanced version of Multilayer Perceptron. In this type, there is one or more than one convolutional layer. Now the basic question is what exactly is a convolutional layer? Convolution is nothing but a simple filtering mechanism that enables an activation. When this filtering mechanism is repeated, it yields the location and strength of a detected feature. As a result of this ability, these networks are widely used in image processing, natural language processing, recommender systems so as to yield effective results of the important feature detected.[24]

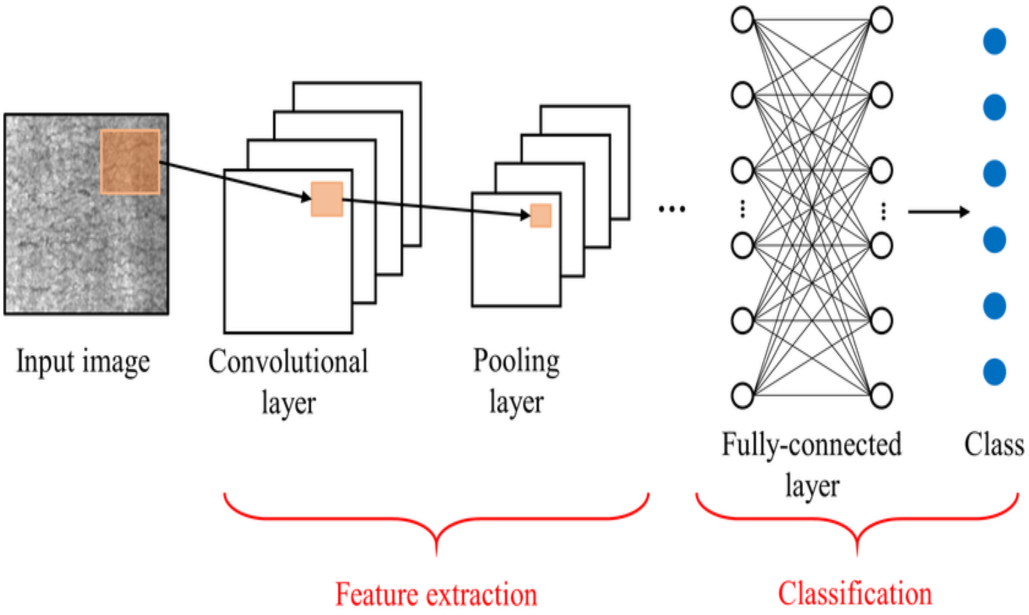


Figure II.5: Convolutional Neural Network Architecture [27]

6. Convolutional Neural Network Layers:

6.1.Convolutional layer:

The convolution operation is one of the fundamental building blocks of a convolutional neural network. Th e convolutional layer's parameters consist of a set of learnable filters (kernels). Every filter is small spatially (along width and height), but extends through the full depth of the input volume. Typical filter sizes might have size 3x3, 5x5, 7x7. The third dimension of the filter corresponds to the number of channels in the input. Th e grayscale image depth is 1 and the color image has 3 (RGB) color channels.

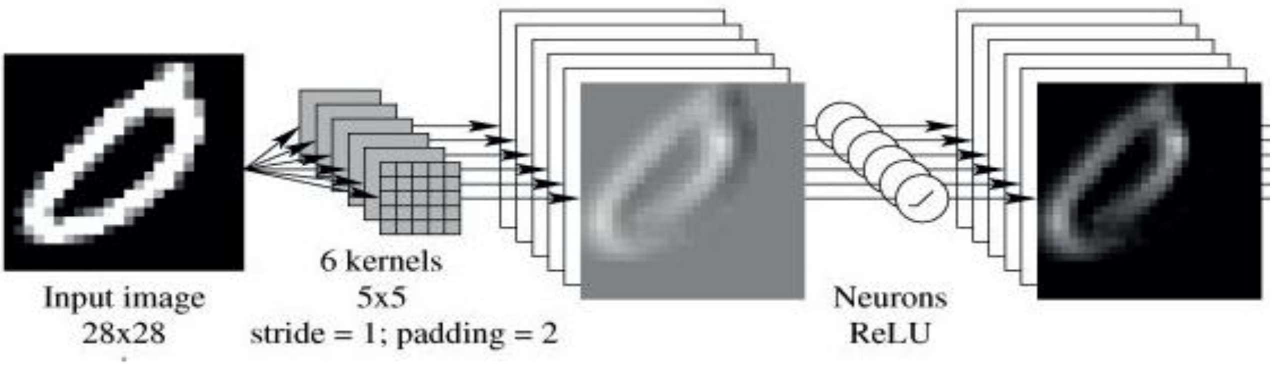


Figure II.6: One convolution layer [28]

During the forward propagation, each filter performs convolution on the input volume across the width and height and compute the dot products between the entries of the filter and the input at any position, this operation is followed by a nonlinear activation function (sigmoid, tanh, ReLU etc.), the resulting outputs are called feature maps. Th e feature map (also known as an activation map), gives the responses of the filter at every spatial position. An example of convolution layer followed by nonlinear activation is shown above. We stack these activation maps along the depth dimension and produce the output volume. The output volume depends on three hyperparameters: depth, stride and padding.

◆ The depth of the output volume represents the number of filters that are used in the convolution operation. Each filter is learning something different in the input, edges, blobs, colors.

◆ The stride is the number of steps that we slide the filter in the input. When the stride is 1 then we move the filters one pixel at a time. When the stride is 2 then the filters jump 2 pixels at a time as we slide them around. Th is will produce smaller output volumes spatially.

◆ Padding allows controlling the output size. Applying convolution to an input, reduce the output size that leads to losing information. To avoid that, we pad the input volume with zeros around the border. Two common choices are valid convolution and the same convolution. The valid convolution means no padding, the same convolution means that the output size remains the same as the input size. The output size is calculated in the following way:

$$\boldsymbol{(n + 2p - f) / s + 1}$$

Where n is the number of filters, p is the amount of padding, f is the filter size and s is the stride.[29]

6.2.Pooling Layer:

CNNs often use pooling layer operation aft er convolution layers, its function is to reduce the dimension, also referred as subsampling or down sampling. Hyperparameters of pooling layer represent the filter size and strides. Most commonly used pooling layer is with filter size 2 and with stride 2. Two common types of pooling layers are max pooling and average pooling, where the maximum and average value is taken, respectively. Max pooling is used often than average pooling. Pooling layer does not have parameters to learn. Th e intuition of what max pooling is doing is that the large number means that there may be detected a feature. An example of convolution layer followed by pooling layer is shown in Figure II.7. [29]

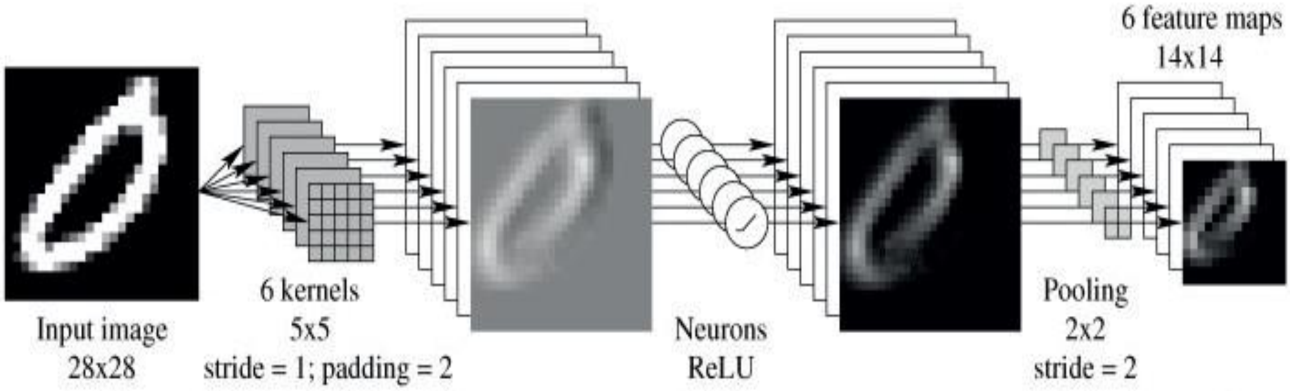


Fig II.7: Convolution layer followed by pooling layer [28]

6.3.Fully connected layer:

After several convolution and pooling layers, the CNN generally ends with several fully connected layers. The tensor that we have at the output of these layers is transformed into a vector and then we add several neural network layers. Th e fully connected layers typically are the last few layers of the architecture as shown in the Fig.14, the dropout regularization technique can be applied in the fully connected layers to prevent overfitting. Th e final fully connected layer in the architecture contains the same amount of output neurons as the number of classes to be recognized. [29]

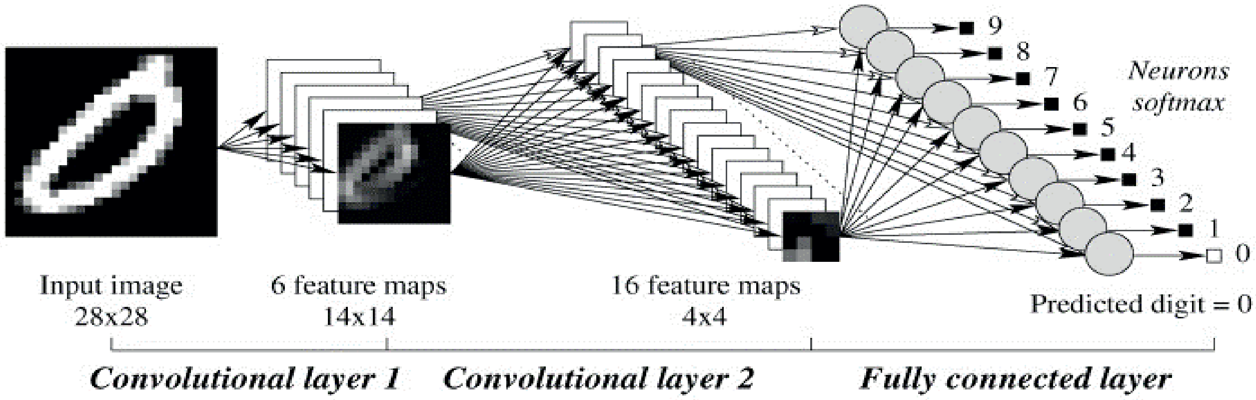


Fig II.8: Two convolutional layers followed by a fully connected layer [28]

7. Convolution Neural Network Architectures:

We have so far described different layers of CNN. Now, we present how these layers are combined to form the architecture of the network. The most common CNN architectures stack a few convolutional layers together, it follows the pooling layer, then this pattern repeats, and we add at the end the fully connected layers.

7.1.AlexNet:

The AlexNet architecture was the first work that popularized Convolutional Networks in Computer Vision, it was the winner of the ImageNet ILSVRC competition in 2012, it had a 15.4% top-5 error rate vs 26.2% for the next lowest network. AlexNet follows the pattern of the LeNet-5 architecture, but it was deeper, bigger, and featured convolutional layers stacked on top of each other.

The input image resolution is 224 × 224, the architecture consists of 5 convolution layers, three 2x2 max-pooling layers and 2 fully connected layers. As shown in Fig. 15, the filter size of the first convolution layer is (11, 11, 3, 96) with a stride of 4, and the first output shape is (55, 55, 96). As a regularization technique, dropout [16] is used in the fully connected layers to reduce overfitting. The total number of parameters in AlexNet is 60 million. [30]

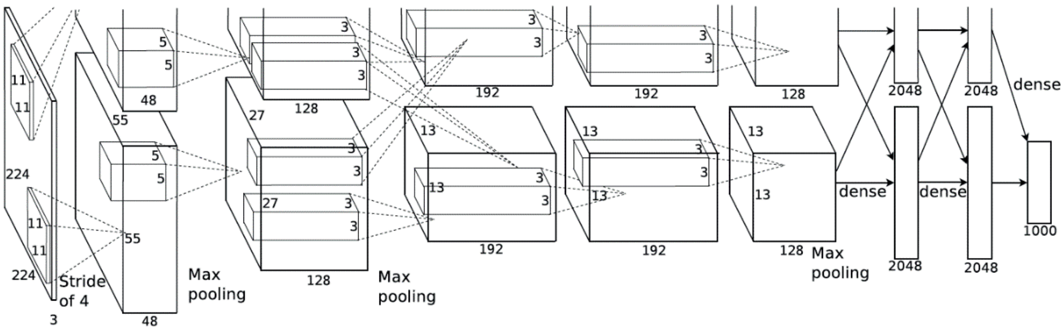


Fig II.9: AlexNet Architecture [28]

7.2.VGGNet:

The VGG Network is introduced in 2014 by Karen Simonyan and Andrew Zisserman. At that time, it was considered as a very deep network. Its main contribution was in showing that the depth of the network is a critical component to achieve better recognition or classification accuracy in CNNs. VGGNet shown in Fig.16, used 3x3 filters, the authors give the intuition behind this that having two consecutive two 3x3 filters gives an effective receptive fi eld of 5x5, and three 3x3 filters give a receptive fi eld of 7x7 filters. The number of filters in the architecture double aft er every max-pooling operation.[30]

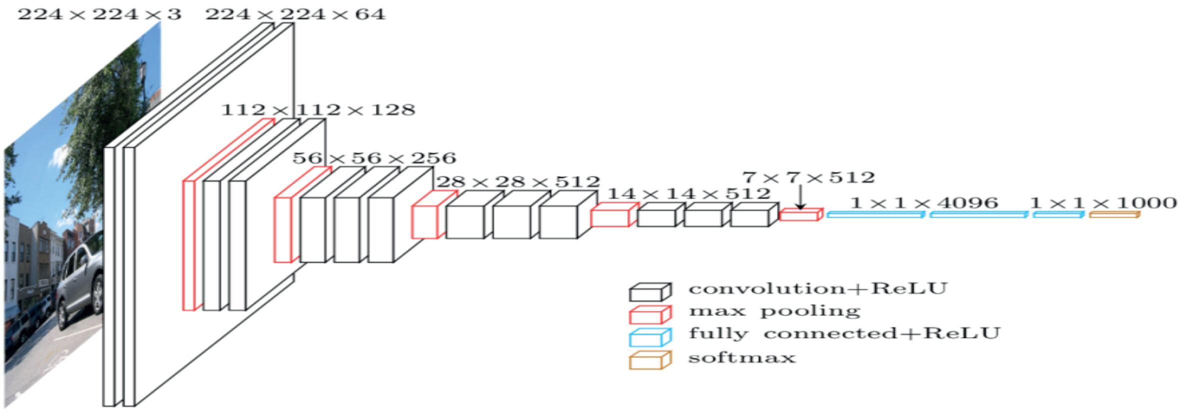


Fig II.10: VGGNet Architecture [28]

7.3.GoogleNet:

The network that won ILSVRC in 2014 is the network GoogleNet. GoogleNet, shown in Figure II.12, is considered to be the first use of modern CNN architecture, which is not composed only on successive convolution and pooling layers, it used inception architecture, that is kind of network in network (NIN).. The inception module skips connections in the network essentially forming a mini-module and that module is repeated throughout the network. The inception module dramatically reduced the number of parameters in the network, GoogleNet employed around 7 million parameters, which represented a 9 times reduction with respect to its predecessor AlexNet, which used 60 million parameters. Furthermore, VGGNet employed about 3 times more parameters than AlexNet.

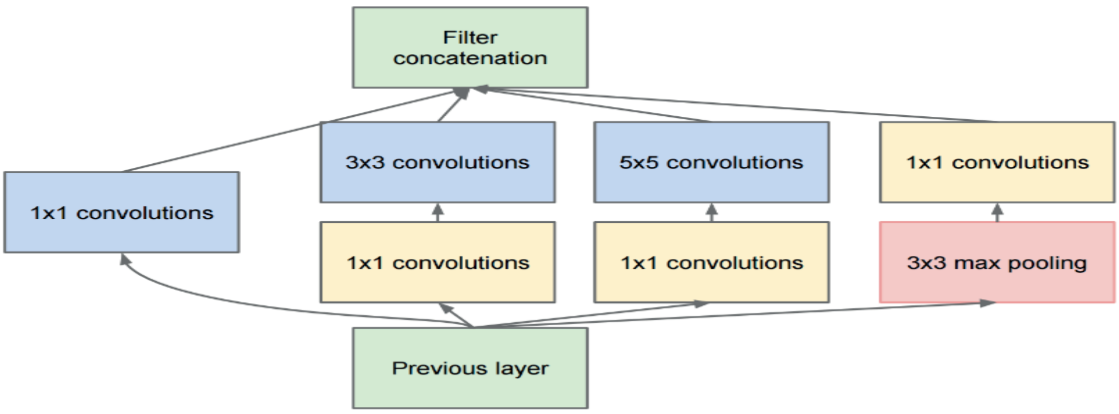


Fig II.11: Inception Model [28]

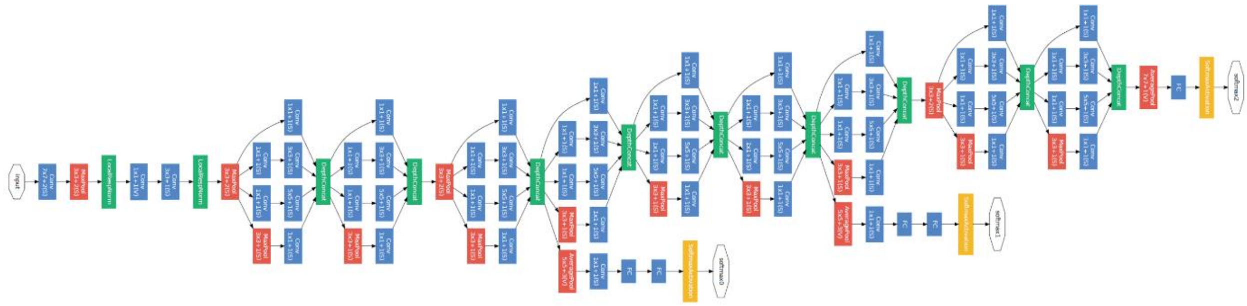


Fig II.12: GoogleNet Architecture [28]

GoogleNet uses 9 inception modules, and it eliminates all fully connected layers using average pooling to go from 7x7x1024 to 1x1x1024, eliminating a large number of parameters that do not seem to matter much. As a form of data augmentation, multiple crops of the same image were created and the network was trained on it. There are also several follow up versions to the GoogleNet, most recently Inception-v4. [31]

8. Activation Function:

Activation Functions are specially used in artificial neural networks to transform an input signal into an output signal which in turn is fed as input to the next layer in the stack. In an artificial neural network, we calculate the sum of products of inputs and their corresponding weights and finally apply an activation function to it to get the output of that particular layer and supply it as the input to the next layer.

8.1.Binary Step function:

When we have an activation function, the most important thing to consideration is threshold-based classifier which means that whether the linear transformation’s value must activate the neuron or not or we can say that a neuron gets activated if the input to the activation function is greater than a threshold value, or else it gets deactivated. In that case the output is not fed as input to the next layer.

During the creation of a file binary activation function the binary workbook is generally used. But the binary step function cannot be used in the case of multi-class classification in the target Portable. Also, the gradient of the binary step function is zero which can cause a hindrance in the back of the propagation step, that is, if we calculate the derivative (fx) relative to x, it is equal to zero. [15]

The binary step function can be defined mathematically as:

$$f(x) = 1, x>=0$$

$$f(x)= 0, x<0$$ [19]

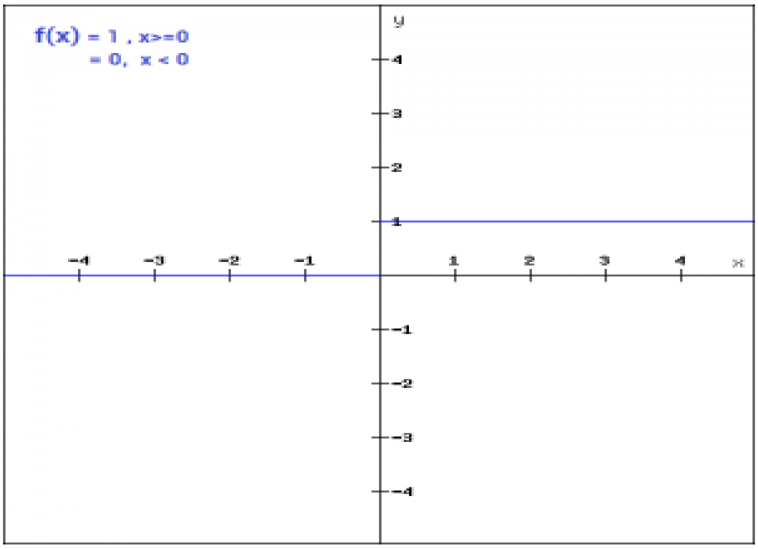


Fig II.13: Binary Step Function [32]

8.2.Linear Activation Function:

The linear activation function is directly proportional to the input. The main drawback of the binary step function was that it had zero gradient because there is no component of x in binary step function. In order to remove that, linear function can be used. It can be defined as:

$$F(x) = ax$$

The value of variable a can be any constant value chosen by the user.

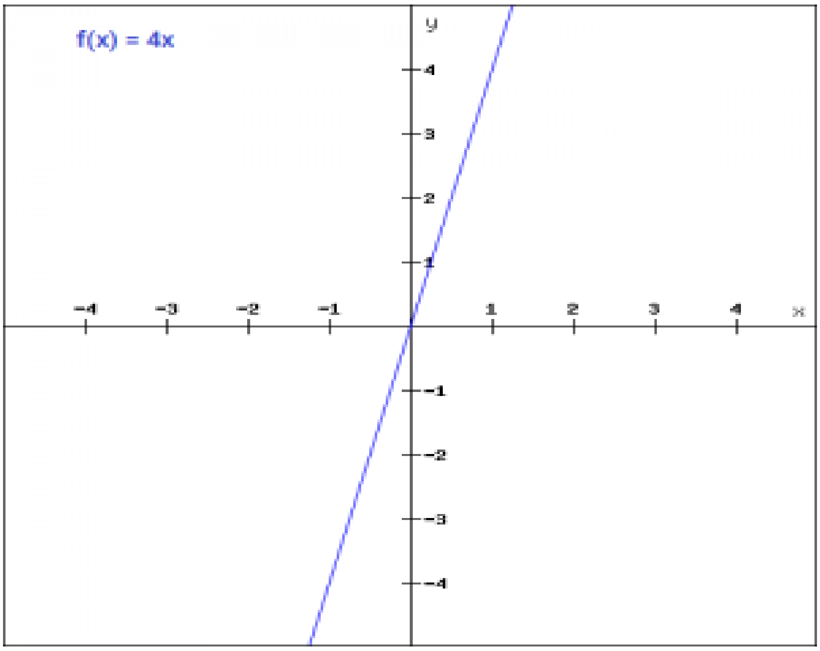


Fig II.14: Linear Activation Function [32]

Here the derivative of the function f(x) is not zero but is equal to the value of constant used. The gradient is not zero, but a constant value which is independent of the input value x which implies that the weights and biases will be updated during backpropagation step although the updating factor will be the same.

There isn't much benefit of using linear function because the neural network would not improve the error due to the same value of gradient for every iteration. Also, the network will not be able to identify complex patterns from the data. Therefore, linear functions are ideal where interpretability is required and for simple tasks.[33]

8.3.Sigmoid Activation Function:

It is the most widely used activation function as it is a non-linear function. Sigmoid function transforms the values in the range 0 to 1. It can be defined as:

$$f(x) = 1/e^{-x}$$

Sigmoid function is continuously differentiable and a smooth S-shaped function. The derivative of the function is:

$$f'(x) = 1\text{-sigmoid}(x)$$

Also, sigmoid function is not symmetric about zero which means that the signs of all output values of neurons will be same. This issue can be improved by scaling the sigmoid function.[34]

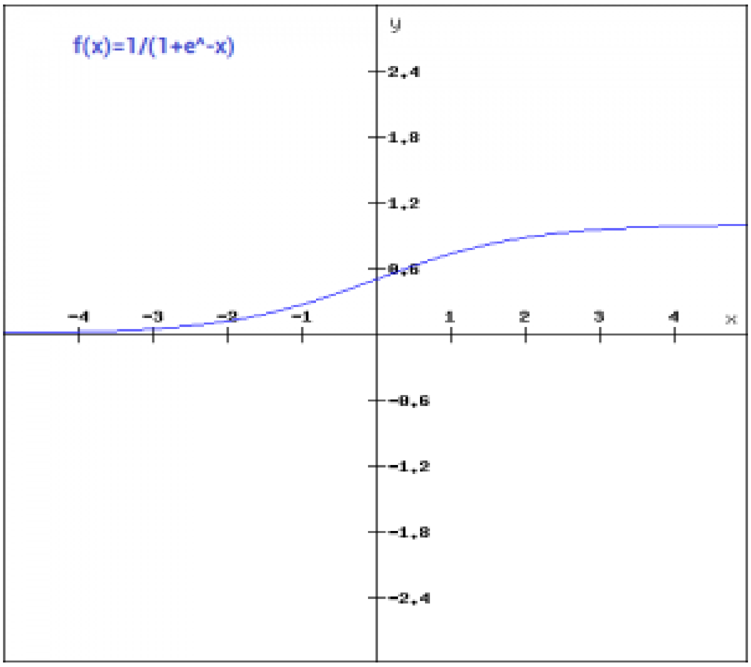


Fig II.15: Sigmoid Activation Function [32]

9. Conclusion:

In conclusion, this chapter has provided an overview of deep learning, specifically focusing on Convolutional Neural Networks (CNNs) and their significance in image classification tasks. We began by examining the fundamental concepts of deep learning and its connection to machine learning, highlighting the role of neural networks in this paradigm. CNNs, a specialized type of neural network, have demonstrated exceptional performance in processing and analyzing images. We delved into the key layers of CNNs, including the convolutional layer, pooling layer, and fully connected layer, elucidating their respective functions and contributions to feature extraction and classification. Additionally, we explored prominent CNN architectures, such as AlexNet and GoogLeNet, which have significantly contributed to the advancements in image classification. The continuous evolution of deep learning techniques promises further breakthroughs in image analysis and classification, opening up exciting opportunities for numerous applications in diverse domains.

Chapter III

Modeling Image Classification Through Simulation

Chapter III: Modeling Image Classification Through Simulation

1. Introduction

The field of image classification has experienced remarkable advancements in recent years, primarily driven by the development of deep learning techniques. Transfer learning, a subfield of deep learning, has emerged as a powerful approach to leverage pre-trained neural networks for solving new classification tasks. In this chapter, we present a study on the simulation of an image classification system using transfer learning in MATLAB.

The chapter is divided into two parts, each addressing an aspect of image classification. The first part focuses on training a simple image classification system. By understanding the fundamental steps involved in training an image classifier, we establish a solid foundation for the subsequent part of our study. The second part delves into the exciting realm of real-time image classification using a webcam and a pre-trained neural network(transfer learning). We explore the concept of transfer learning, which allows us to harness the knowledge encoded in pre-existing neural networks and apply it to new, unseen image data. By utilizing the webcam as an input source, we enable the system to classify images in real-time, opening up possibilities for applications such as surveillance, object recognition, and interactive systems.

Throughout this chapter, we aim to provide a understanding of the image classification process and showcase the practical implementation of transfer learning in MATLAB. By the end, readers will gain proficiency in training a simple image classifier and utilizing pre-trained neural networks for real-time image classification. This chapter serves as a valuable resource for researchers and practitioners in the field of computer vision, enabling them to explore and experiment with image classification systems.

2. Software and Tools

2.1 Matlab2023:

MATLAB 2023 is the latest version of the renowned computational tool used in a wide range of disciplines, including engineering, science, and data analytics. This version brings several exciting features and enhancements that empower users to achieve even greater computational excellence and innovation**.** It expands Deep Learning Toolbox by:

Recognizing the transformative power of deep learning, MATLAB 2023 expands its Deep Learning Toolbox. This update includes updated pre-trained models, advanced network architectures, and streamlined workflows for training and deploying deep learning models. Users can leverage the latest advancements in AI and machine learning to solve complex problems effectively.

2.2 Deep learning toolbox:

The Deep Learning Toolbox is a software library that provides tools and functions for implementing and deploying deep learning models. It is commonly used in the field of machine learning and artificial intelligence to develop and train deep neural networks.

Here's an overview of the main features and capabilities typically offered by a deep learning toolbox:

- Neural Network Architectures
- Model Training
- Data Preprocessing
- Transfer Learning
- Hyperparameter Optimization
- Model Evaluation

It's important to note that the specific features and capabilities of a deep learning toolbox may vary depending on the software or library you're using.

In the realm of image classification, various methods have been explored to decipher the content and meaning within images. While alternative approaches exist, it is undeniable that the utilization of neural networks has revolutionized the field, leading to unprecedented advancements.

In this chapter we are going to explore together why it is better to rely on neural networks then other methodologies.[35]

3. **Part One**: Training A simple Image Classification System

3.1 Data set:

The **Digit-dataset** consists of 10000 28x28 gray scale images in 10 classes (check figure III.1), with 1000 images per class. The images will be divided into 750 into training and 250 into test. The dataset is divided into three training batches and one test batch, each with 10000 images. The test batch contains exactly 250 randomly-selected images from each class. The training batch contain the remaining images in random order. But some training batch may contain more images from one class than another.

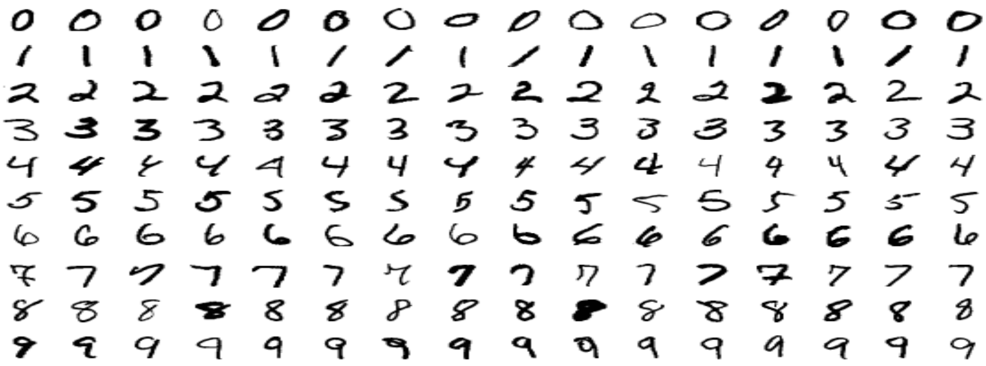


Figure III.1: Digit-Data Set [36]

3.2 Designing the Network Architecture:

It is a good practice, to start with a basic model at first and then keep trying to improve it in every step. To further improve the Model, we need to:

- ❖ **Define the layers:**

in our simulation we used the following 7 layers

- ❖ Image Input Layer (28*28*1)
- ❖ convolution Layer (30*10*10*1)
- ❖ batch Normalization Layer
- ❖ RELU Layer
- ❖ fully Connected Layer (10 Layers)
- ❖ SOFTMAX Layer
- ❖ Classification Layer

ANALYSIS RESULT

	Name	Type
1	imageinput 28×28×1 images with 'zerocenter' norm...	Image Input
2	conv 30 10×10×1 convolutions with stride [1 1...	2-D Convolution
3	batchnorm Batch normalization with 30 channels	Batch Normalization
4	relu ReLU	ReLU
5	fc 10 fully connected layer	Fully Connected
6	softmax softmax	Softmax
7	classoutput crossentropyex with '0' and 9 other clas...	Classification Output

Figure III.2: Training Layers

in our experiment we only going to manipulate the **Convolutional Layer**

❖ **Training:** in our training we are going to manipulate different options like Optimization Algorithms (**Adam, Sgdm**), **Epoch** number, **Validation Frequency** , size and filters in **Convolutional Layer**

```
% Train Network
options = trainingOptions('adam', ...
    'MaxEpochs',3, ...
    'ValidationData',imdsValidation, ...
    'ValidationFrequency',80, ...
    'Verbose',false, ...
    'Plots','training-progress');

net = trainNetwork(imdsTrain,layers,options);
```

Figure III.3: Training options

- **Optimization algorithms:** mathematical procedures or iterative methods used to minimize (or maximize) an objective function with respect to a set of variables or parameters. In the context of machine learning and deep learning, these algorithms are employed to adjust the parameters of a model in order to minimize the loss function and improve its performance. There are few optimization algorithms such as:

- **Adam (Adaptive Moment Estimation)**: is an optimization algorithm commonly used in neural network training. It adapts the learning rate based on gradient estimates, allowing for faster convergence.[37]
- **Sgdm (Stochastic Gradient Descent with Momentum)**: is an optimization algorithm used in training neural networks. It extends the standard stochastic gradient descent (SGD) by incorporating momentum, which helps accelerate convergence and overcome local minima. SGDM accumulates a weighted average of past gradients to update the parameters, allowing for more consistent and stable updates during training.[38]

- **Epoch:** An epoch is a complete pass through the entire training dataset during model training. It consists of forward propagation, calculating the loss, backward propagation, and updating the model's parameters. In each epoch, the model makes predictions on all training examples, evaluates the error, and adjusts the model's parameters to minimize the error.[39]
- **Validation frequency:** it refers to how often the model's performance is evaluated on a validation set during the training process. It determines how frequently the model's progress is assessed and can impact the training efficiency and the ability to detect overfitting.[39]

3.3 Results

At first, we set the optimization algorithm to **"Adam"** and the results are shown in **Table III.1**

Validation frequency / Epoch	10	30	60
2	**95.48%**	**93.68%**	**94.76%**
4	**98.04%**	**98.16%**	**97.72%**

Table III.1: Classification accuracy using Adam algorithm

Then we set the optimization algorithm to **"Sgdm"** and the results are shown in **Table III.2**

Validation frequency / **Epoch**	**10**	**30**	**60**
2	**96.88%**	**96.96%**	**97.28%**
4	**98.52%**	**98.92%**	**98.92%**

Table III.2: Classification accuracy using Sgdm algorithm

Comment: Sgdm model with **4** Epoch and **60** validation frequency gave the best accuaracy of **98.92%** this result is acquired in convlotional layer with a size of 5 and 20 number of layer but we want more precesion so we will manupilate the **size** and **number** of the convolutional layer

Number of layers / **Size**	**10**	**30**	**60**
5	**97.96%**	**98.20%**	**94.20%**
10	**97.72%**	**99.20%**	**98.64%**

Table III.3: Accuracy of Image Classification with parameter variation

Size = 10, Number of layers=30

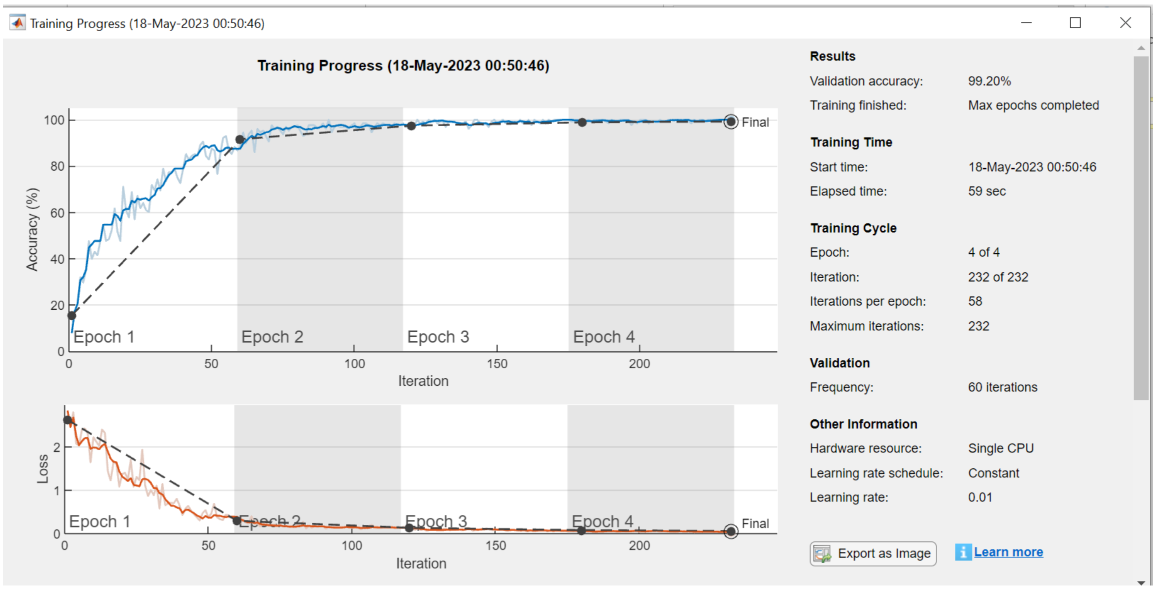


Figure III.4: Optimal Accuracy of Image Classification using Deep Learning

3.4 Discussion

After manipulating the training options by changing the optimization algorithms, number of epoch, validation frequency, size and number layers of the convolutional layer by observing the results giving by the program we have reached the outcome that the training options which gives the best results are **Sgdm** as an optimization algorithm, **4** as number of epoch and **60** as validation frequency, 10 as a size and 20 as number of convolutional layers with **99.20%** accuracy **Figure III.4** summarize the training process. The results are shown in **Table III.3**

The results revealed that appropriate choices of training options could significantly affect the accuracy and convergence speed of the deep learning architecture. By experimenting with different training configurations, insights were gained into the optimal settings for achieving improved image classification performance.

4. **Part two:** Image Classification using transfer learning

In this part we are going to use a pre-trained neural network (transfer learning) to test its performance and observe the factors that may affect the accuracy of the system such as the WEBCAM, background, brightness, right positioning…. etc.

4.1 The System:

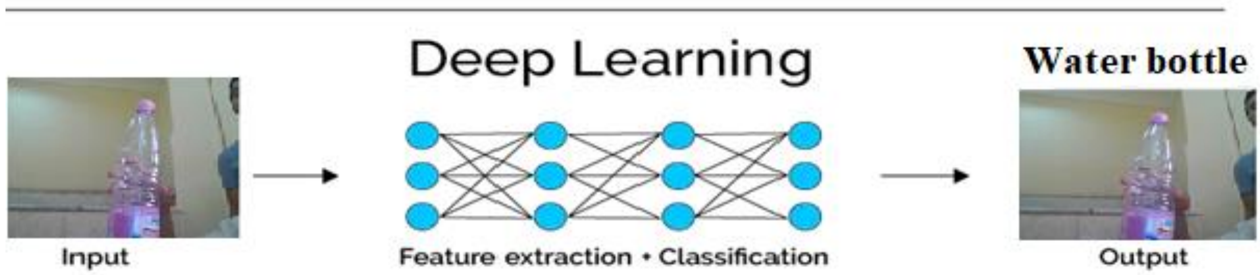


Figure III.5: Image Classification System

4.1.1 Input: our input is live view from our webcam for random objects such: a computer mouse, water bottle, phone, a toy supercar, modem, book, charger…

Figure III.6: Input Images Was Taken by Our System Webcam

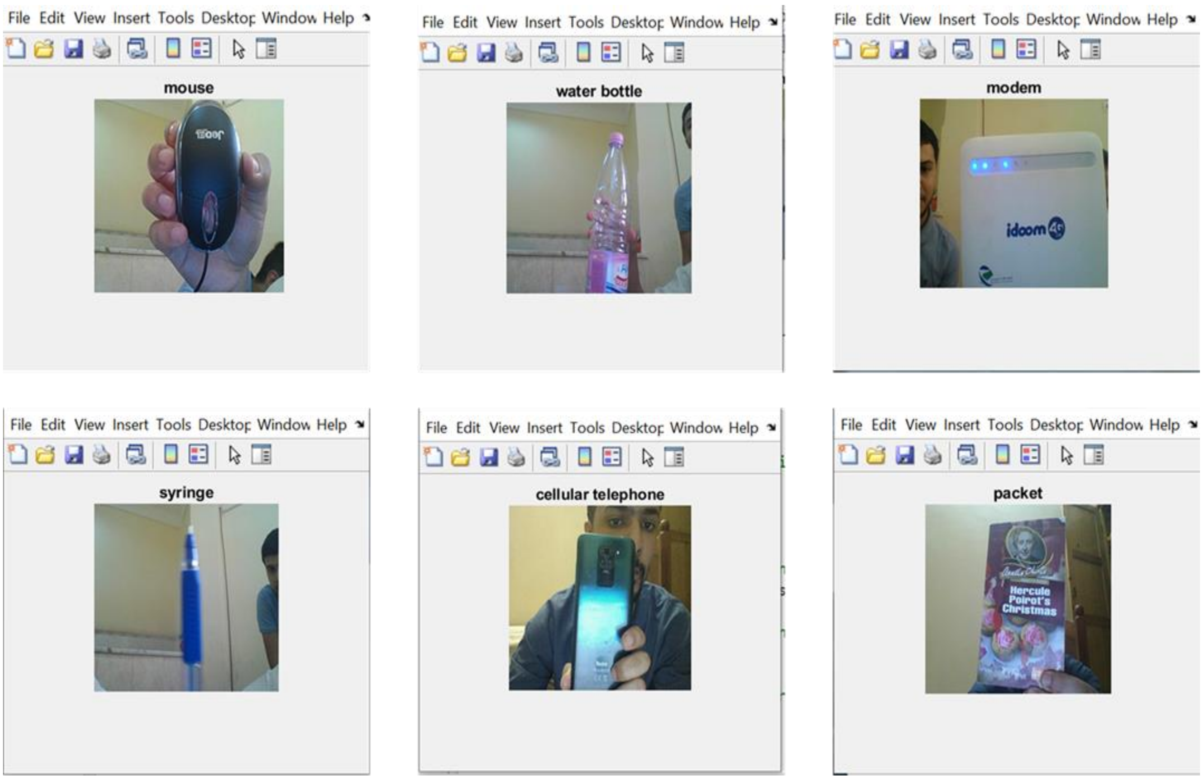
4.1.2 Deep learning: We going to uses pre-trained neural network(transfer learning); The models used to classify the images are "GoogleNet","Vgg16","AlexNet".

These CNN models work by employing convolutional layers to extract features from input images. These convolutional layers consist of filters that scan the image, capturing patterns and features. The extracted features are then passed through activation functions (such as ReLU) to introduce non-linearity. Pooling layers down sample the spatial dimensions, reducing the computational load and capturing important information. The resulting feature maps are flattened and fed into fully connected layers that perform the final classification. During training, the model adjusts its weights using back propagation and optimization algorithms to minimize the prediction error. **GoogleNet,** also known as Inception-v1 it that was trained on the ImageNet dataset. The

ImageNet dataset is a large-scale visual recognition challenge that contains millions of labeled images from thousands of categories. The ImageNet dataset includes a wide range of categories covering various objects, animals, scenes, and concepts. Some of the common categories that GoogleNet is trained on, as part of the ImageNet dataset, include:

- Animals: Various species of mammals, birds, reptiles, insects, etc.
- Objects: Everyday objects such as chairs, tables, cars, bicycles, etc.
- Fruits and Vegetables: Different types of fruits, vegetables, and crops.
- Electronics: Devices like smartphones, laptops, cameras, TVs, etc.
- Vehicles: Different kinds of vehicles including cars, trucks, airplanes, boats, etc.
- Buildings: Different types of architectural structures like houses, skyscrapers, bridges, etc.
- People: Individuals from various ethnicities, ages, and professions.
- Sports: Different sports and related equipment like soccer, basketball, tennis, etc.
- Fine Arts: Paintings, sculptures, and artworks from various periods and artists.

4.1.3 Output: It represents the results giving by the system and it also represents the decision made by our system regarding the input inserted earlier and it is as follows:



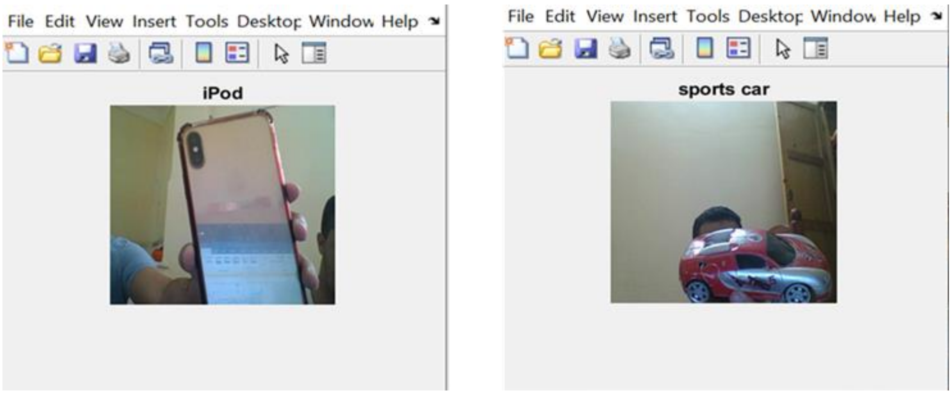


Figure III.7: Output Results

We notice that there are few mistakes in classifying some of the objects so we tried to change item's positioning, brightness and background and see if it has something to do with those mistake the figure below show the results of those changes

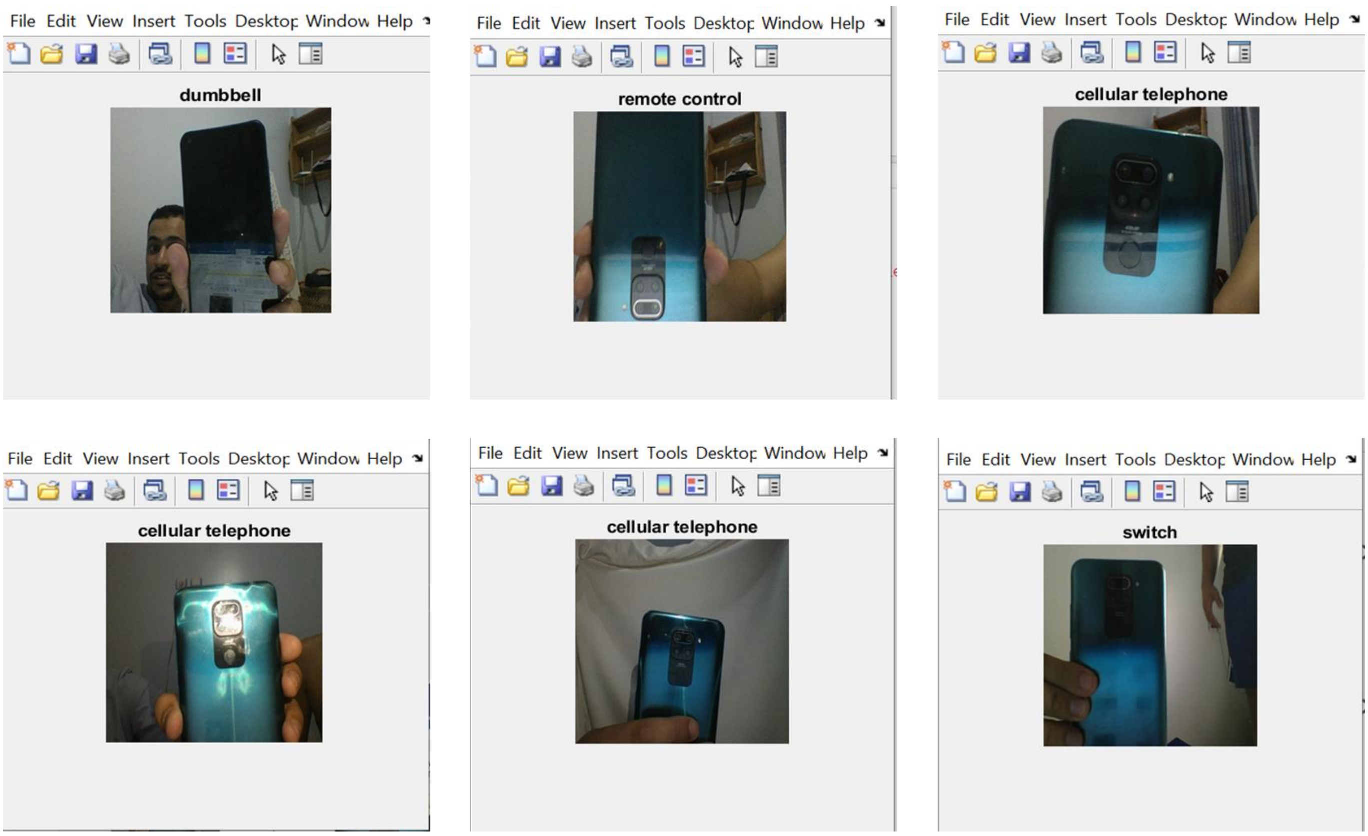


Figure III.8: Output result with different position, brightness and background

4.2 Discussion:

After executing the system by trying out different CNN models we reached the outcome that the best model that fits our system is **GoogleNet**, as for the errors it is mainly due to webcam quality, item positioning, brightness.

The choice of a good CNN model and the quality of the webcam , item positioning , brightness has a direct effect on the decision making of the system. By experimenting with different CNN models and other configurations, we got the system to be in a good conditions to make proper decisions regarding images classification.

5. Conclusion

This chapter has presented an exploration of simulation techniques for image classification. We commenced by discussing the training of a simple image classification system, covering the main steps such as data acquisition, network selection, and training. This foundation laid the groundwork for the subsequent part of our study. We also focused on utilizing a webcam to perform real-time image classification with the aid of pre-trained neural networks. By integrating the webcam into the system, we showcased the potential for real-time applications and emphasized the significance of transfer learning in practical scenarios.

Conclusion

In conclusion, this master's dissertation has provided an exploration of deep learning techniques for image classification. The availability of large volumes of data and advancements in processing power have empowered artificial intelligence-based systems to store, analyze, index, and search through vast amounts of information. Machine learning, particularly classification, has emerged as a powerful tool for organizing and categorizing data based on their characteristics.

Within the context of image datasets, classification plays a pivotal role in categorizing images into various classes, enabling more effective utilization of these datasets. The goal of image classification is to bridge the gap between computer vision and human vision by training computers to interpret and understand visual data. Deep learning, as a subset of machine learning, has proven to be a potent technique in image classification.

Convolutional neural network (CNN) models, specifically designed for image processing, excel at extracting relevant features from images and performing accurate classifications. These models have revolutionized the field of computer vision, achieving remarkable results in image recognition, object detection, and image segmentation tasks.

In the first part, a simple deep learning architecture is designed and trained using digit datasets. Different training options, including optimization algorithms, are investigated to understand their impact on model performance.

The second part focuses on transfer learning for image classification, utilizing a pre-trained convolutional neural network (CNN) model and webcam images as the target dataset. MATLAB's deep learning toolbox enables the implementation and evaluation of this transfer learning approach.

References

[1] Tom, M. "Mitchell: Machine Learning." 1997 Burr Ridge 45.37 (1997): 870-877.

[2] Onuoha, F. (2021). 10 Important Machine Learning Terminologies you should always keep in mind. DEV Community. https://dev.to/heyonuoha/10-important-machine-learning-terminologies-you-should-always-keep-in-mind-3mna

[3] Thomas H. Davenport. Machine learning. https://www.sas.com/en_us/insights/analytics/machine-learning.html.

[4] Deb, D. (2020). Supervised Machine Learning: a beginner's guide. Dibyendu Deb. https://dibyendudeb.com/supervised-machine-learning/

[5] Fig. 5. Main unsupervised learning algorithms developed in Julia. (n.d.). ResearchGate. https://www.researchgate.net/figure/Main-unsupervised-learning-algorithms-developed-in-Julia_fig5_340114090

[6] Boesch, G. (2023). A Complete Guide to Image Classification in 2023. viso.ai. https://viso.ai/computer-vision/image-classification/

[7] Prince, Simon JD. Computer vision: models, learning, and inference. Cambridge University Press, 2012.

[8] Nixon, Mark, and Alberto Aguado. Feature extraction and image processing for computer vision. Academic press, 2019.

[9] K-Nearest Neighbor (KNN) Algorithm for Machine Learning - Javatpoint. (n.d.). www.javatpoint.com. https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning

[10] Fig.4-KNN K-NN Algorithms: -K-NN algo following some steps. 1. Pick a. . . (n.d.). ResearchGate. https://www.researchgate.net/figure/KNN-K-NN-Algorithms-K-NN-algo-following-some-steps-1-Pick-a-value-for-K-2-Calculate_fig2_341831889

[11] Bezdek, James C. Pattern recognition with fuzzy objective function algorithms. Springer Science & Business Media, 2013.

[12] Gupta, A. (2022, January 30). Fuzzy C-Means Clustering (FCM) Algorithm - Geek Culture - Medium. Medium. https://medium.com/geekculture/fuzzy-c-means-clustering-fcm-algorithm-in-machine-learning-c2e51e586fff

[13] GeeksforGeeks. (2023b). Support Vector Machine Algorithm. GeeksforGeeks. https://www.geeksforgeeks.org/support-vector-machine-algorithm/

[14] Fig. 6. Support Vector Machine visualization. (n.d.). ResearchGate. https://www.researchgate.net/figure/Support-Vector-Machine-visualization_fig5_332248436

[15] Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. Deep learning. MIT press, 2016.

[16] Figure 2. A basic convolutional neural network structure for image. . . (n.d.). ResearchGate. https://www.researchgate.net/figure/A-basic-convolutional-neural-network-structure-for-image-classification-Convolutional_fig1_342316332

[17] Neil D. Lawrence. Machine learning motivation.

[18] LeCun, Yann, et al. "Gradient-based learning applied to document recognition." Proceedings of the IEEE 86.11 (1998): 2278-2324.

[19] What is Deep Learning? | IBM. (n.d.). https://www.ibm.com/topics/deep-learning

[20] Hauptfleisch, K. (2021). This Is How Artificial Intelligence Will Shape eLearning For Good. eLearning Industry. https://elearningindustry.com/artificial-intelligence-will-shape-elearning

[21] Coursera. (2023). Deep Learning vs. Machine Learning: Beginner's Guide. Coursera. https://www.coursera.org/articles/ai-vs-deep-learning-vs-machine-learning-beginners-guide

[22] I will do python, numpy, pandas,scikit learn and matplotlib. (n.d.). Fiverr. https://www.fiverr.com/huwaiza/do-python-numpy-pandas-scikit-learn-and-matplotlib

[23] Chollet, Francois. "The limitations of deep learning." Deep learning with Python (2017).

[24] AGGARWAL, Charu C., et al. Neural networks and deep learning. Springer, 2018, vol. 10, no 978, p. 3.

[25] Figure 4. Feedforward neural network structure. (n.d.). ResearchGate. https://www.researchgate.net/figure/Feedforward-neural-network-structure_fig4_341786016

[26] Figure 3.3: The configuration of the deep neural network used to. . . (n.d.). ResearchGate. https://www.researchgate.net/figure/The-configuration-of-the-deep-neural-network-used-to-approximate-Q-values_fig3_343745307

[27] Lee, Soo Young, et al. "Steel surface defect diagnostics using deep convolutional neural network and class activation map." Applied Sciences 9.24 (2019): 5449.

[28] Džakula, N. (2019). Convolutional Neural Network Layers and Architectures. https://doi.org/10.15308/sinteza-2019-445-451

[29] Bačanin Džakula, Nebojša. "Convolutional neural network layers and architectures." Sinteza 2019-International Scientific Conference on Information Technology and Data Related Research. Singidunum University, 2019.

[30] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Communications of the ACM 60.6 (2017): 84-90.

[31] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, "Going Deeper with Convolutions," IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 1-9

[32] Sharma, Sagar, Simone Sharma, and Anidhya Athaiya. "Activation functions in neural networks." Towards Data Sci 6.12 (2017): 310-316.

[33] Haykin, Simon. Neural networks and learning machines, 3/E. Pearson Education India, 2009.

[34] Karlik, B., & Olgac, A. V. (2011). Performance Analysis Of Various Activation Functions In Generalized MLP Architectures Of Neural Networks. International Journal Of Artificial Intelligence And Expert Systems, 1(4), 111-122

[35] Deep Learning Toolbox Documentation. (n.d.). https://www.mathworks.com/help/deeplearning/

[36] Ardi, M. (2021, December 15). Simple Neural Network on MNIST Handwritten Digit Dataset. Medium.https://becominghuman.ai/simple-neural-network-on-mnist-handwritten-digit-dataset-61e47702ed25

[37] Kingma, D. P., and J. L. Ba. "Adam: A method for stochastic optimization. Conference paper at the Third International Conference on Learning Representations 2015, San Diego, USA, 15 pp." (2015).

[38] Qian, Ning. "On the momentum term in gradient descent learning algorithms." Neural networks 12.1 (1999): 145-151.

[39] Options for training deep learning neural network - MATLAB training Options. (n.d.). https://www.mathworks.com/help/deeplearning/ref/trainingoptions.html