PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA MINISTRY
OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

IBN KHALDOUN UNIVERSITY – TIARET

THESIS

PRESENTED BY:
Khiar sadek
Taouch adel abdelmonsif

FACULTY OF MATHEMATICS AND INFORMATICS
DEPARTMENT OF INFORMATICS
FOR THE COMPLETION OF
THE MASTERS DEGREE IN COMPUTER SCIENCE
Field: software engineering
In the subject

Transfer Learning Applied to Algerian Gastronomy Images Recognition

Publicly presented in Tiaret before a jury composed of

Mr BERBER EL-Mehdi	Ibn Khaldoun University Tiaret	**President**
Mr MEZZOUG Karim	Ibn Khaldoun University Tiaret	**Supervisor**
Mr CHENINE Abdelkader	Ibn Khaldoun University Tiaret	**Examiner**

2022-2023

Thanks

At the end of this long adventure, I would first like to thank God Almighty for giving me the courage and patience needed to finish this work.

I would especially like to thank my supervisor,

Ms. Muzouk Karim, for the competent assistance she provided me, For his patience and encouragement.

His critical eye has been invaluable to me in organizing the work and improving the quality of the various departments. I would like to thank Professor BERBE Al-Mahdi 66and Professor CHENINE Abdelkader for considering the evaluation of my graduate thesis.

Thank you, Ms. Saji Fatima, for her support in all financial and economic matters. ,

Last but not least, I would like to thank my family, starting with my very dear parents to whom I have dedicated this title as a token of gratitude. No devotion can express my great eternal love for you and my great gratitude for it

The sacrifices you made for my education and my well-being. I thank all my relatives for their help, love and precious presence by my side through all these years.

Taouch Adel Abdelmonsif

Thanks

At the end of this long adventure, I would first like to thank God Almighty for giving me the courage and patience needed to finish this work.

I would especially like to thank my supervisor,

Ms. Muzouk Karim, for the competent assistance she provided me, For his patience and encouragement.

His critical eye has been invaluable to me in organizing the work and improving the quality of the various departments. I would like to thank Professor BERBER Al-Mahdi and Professor CHENINE Abdelkader for considering the evaluation of my graduate thesis.

Thanks, Ms. Saji Fatima, for her support in all financial and economic.

Last but not least, I would like to thank my family, starting with my very dear parents to whom I have dedicated this title as a token of gratitude. No devotion can express my great eternal love for you and my great gratitude for i The sacrifices you made for my education and my well-being. I thank all my relatives for their help, love and precious presence by my side through all these years.

KHIAR Sadek

Abstract

Automatic food recognition has become a relevant necessity in the computer vision (CV) systems industry, i.e., the mobile applications industry for intelligent food delivery or smart agriculture, food ripeness rates, etc., and a topical scientific subject for ongoing scientific research, and a highly topical scientific subject for which scientific research is ongoing. to advance and improve the algorithms used in the field, as well as the performance of this type of embedded system, which on the one hand is in direct contact with the real environment, and on the other is tolerant of a certain acceptable error rate given people's state of health and dietary desires. Among these systems is machine learning (ML), which is part of the systems generated by artificial intelligence (AI), and among its sub-disciplines is deep learning (DL), which is a widely used approach in the field of artificial neural networks (ANN). In this context, we will explore these concepts and take part in the development of an automated classification system for images of Algerian gastronomy, based on deep learning using the idea of transfer learning from the best-performing classical models in the field.

Image classification, such as InceptionV3, resNet, etc.

Experiments will be carried out on an existing dataset developed by the Ministry containing 6 food categories, and we have increased this dataset to 9 categories.

Keywords:

Artificial Vision, Machine Learning, Deep Learning, Artificial Neural Networks, image classification,

InceptionV3, resNet

Résume

La reconnaissance automatique des aliments est devenue une nécessité pertinente dans l'industrie des systèmes de vision artificielle (AV), c'est-à-dire l'industrie des applications mobiles pour la livraison intelligente de nourriture ou l'agriculture intelligente, les taux de maturité des aliments, etc., et un sujet scientifique d'actualité dont la recherche scientifique se poursuit. faire progresser et améliorer les algorithmes utilisés sur le terrain ainsi que les performances de ce type de systèmes embarqués, qui sont d'une part en contact direct avec l'environnement réel et d'autre part leur tolérance à un certain taux d'erreur acceptable compte tenu de l'état de santé des personnes et envies diététiques. Parmi ces systèmes, on peut citer le machine learning (ML) qui fait partie des systèmes générés par l'intelligence artificielle (IA) et qui parmi ces sous-disciplines est le deep learning (DL) qui est une approche largement utilisée à l'heure actuelle dans le domaine de réseaux de neurones artificiels (ANN). Ainsi, dans ce cadre, nous découvrirons tous ces concepts et participerons au développement d'un système de classification automatisé des images de la gastronomie algérienne basé sur le deep learning utilisant l'idée de transfert learning à partir des modèles classiques les plus performants en la matière.

Classification d'images, telles que: InceptionV3, resNet, etc.

Des expériences seront effectuées sur un ensemble de données existant développé par le ministère contenant 6 catégories d'aliments, et nous avons augmenté cet ensemble de données à 9 catégories.

Mots-clés

vision artificielle, le machine learning, deep learning, réseaux de neurones artificiels, modèles

الملخص

أصبح التعرف التلقائي على الطعام ضرورة ذات صلة في صناعة أنظمة الرؤية الآلية (AV)، أي صناعة تطبيقات الهاتف المحمول لتقديم الطعام الذكي أو الزراعة الذكية، ومعدلات نضج الطعام، وما إلى ذلك، وموضوع علمي حالي يستمر البحث العلمي منه في التقدم وتحسين الخوارزميات المستخدمة في هذا المجال بالإضافة إلى أداء هذا النوع من الأنظمة المضمنة، والتي هي في المقام الأول على اتصال مباشر بالبيئة الحقيقية وثانيًا تحملها لمعدل خطأ معين مقبول مع مراعاة صحة الأشخاص ونظامهم الغذائي الرغبات. من بين هذه الأنظمة، يمكننا أن نذكر التعلم الآلي (ML) وهو جزء من الأنظمة الناتجة عن الذكاء الاصطناعي (AI) والذي يعد من بين هذه التخصصات الفرعية التعلم العميق (DL) وهو نهج يستخدم على نطاق واسع في الوقت الحالي في مجال الشبكات العصبية الاصطناعية (ANN). وبالتالي،

سوف نكتشف كل هذه المفاهيم وأن نكون جزءًا من تطوير نظام تصنيف آلي لصور فن الطهو الجزائري على أساس التعلم العميق باستخدام فكرة نقل التعلم من أكثر النماذج الكلاسيكية كفاءة في هذا المجال. تصنيف الصور، مثل: InceptionV3 وresNet وغيرها.

سيتم إجراء التجارب على مجموعة بيانات حالية تم تطويرها على مستوى الأقسام تحتوي على 6 فئات غذائية، وقمنا بزيادة مجموعة البيانات هذه إلى 9 فئات.

الكلمات المفتاحية

الرؤية الآلية، التعلم الآلي، الذكاء الاصطناعي، التعلم العميق، الشبكات العصبية الاصطناعية، تصنيف آلي

Contents

Chapter III: Related work (state of art)

Chapter IV: implementation and experiments

List of figures

Table list

Abbreviation list

AV: Artificial Vision

RGB: RED, GREEN, BLUE

CV: Computer vision

AI: Artificial Intelligence

ANI: Artificial Narrow Intelligence

AGI: Artificial General Intelligence

ASI: Artificial Super Intelligence

ML: Machine Learning

ANN: Artificial Neural Network

BNN: Biological Neural Network

RNN: Recurrent Neural Network

CNN: Convolutional Neural Network

ReLU: Rectified Linear Unit

FC: Fully Connected

DL: Deep Learning

DNN: Deep Neural Network

VGG: Visual Geometry Group

ResNet: Residual Network

CL: Convolution Layer

PL: Pooling Layer

ACC: Accuracy

General introduction

General introduction

We live in a digital world, where information is stored, processed, indexed, and searched by computer systems, making retrieval a quick and inexpensive task. In recent years, considerable progress has been made in the field of image classification. This progress is due to the many works in this field and the availability of international image databases that have allowed researchers to credibly report the performance of their approaches in this field, with the possibility of comparing them to other approaches they use the same basics.

In the late 80s Yan le Cun developed a particular type of network called the convolutional neural network. These networks are a particular form of multilayer neural network whose architecture of connections is inspired by that of the visual cortex of mammals. For example, each element is only connected to a small number of neighboring elements in the previous layer. In 1995, Yan Lecun and two other engineers developed an automatic check reading system that has been widely deployed around the world. In the late 1990s, this system read between 10 and 20 percent of all checks issued in the United States. But these methods were rather difficult to implement with the computers of the time, and despite this success, convolutional networks and neural networks more generally were neglected by the research community between 1997 and 2012.

In 2011 and 2012 three events suddenly changed the situation. First, GPUs (Graphical Processing Units) capable of more than a trillion operations per second became available for a cheaper price. These powerful specialized processors, initially designed for graphics rendering in video games, have proven to be very efficient for neural network calculations. Second, experiments conducted simultaneously at Microsoft, Google, and IBM with the help of Geoff Hinton's lab showed that deep networks could halve the error rates of speech recognition systems. Third, several records in image recognition have been broken by convolutional neural networks. The most significant event was the resounding victory of the Toronto team in the “ImageNet” object recognition competition. The drop-in error rates were such a game-changer. Overnight, the majority of speech and vision research teams abandoned their preferred methods and switched to convolutional neural networks and other neural networks. The Internet industry immediately seized the opportunity and

started to invest heavily in deep learning research and development teams.

In our project we will use convolutional neural networks to classify images, we will create different models with different architectures.

To do this, we have structured our thesis in four chapters:

- In the first chapter we will present Artificial vision.
- The second chapter is devoted to Artificial Intelligence and machine learning.
- The third chapter, we have the state of the art.
- In the fourth chapter, we will show the experimental part of our work and discuss the different results obtained.

We end with a general conclusion.

Chapter I
Artificial Vision

1 Introduction

A photograph is a representation of a person or an object through a drawing, drawing, or painting photography, cinema, etc... It is also an organized collection of information that is displayed on a screen. This display makes sense to the human eye. Image processing can be defined as the set of methods and techniques that operate on the image for extraction the most relevant information or simply to provide a clearer picture of human eye.

In this chapter we introduce some basic concepts of image processing in the field of computer vision.

2 What is an image?

In computing, an image is a digital representation of a visual scene, object, or pattern. It can be thought of as a two-dimensional array of pixels, where each pixel corresponds to a specific location in the image and represents a particular color or grayscale value.

Images can be created using a variety of tools and methods, such as digital cameras, scanners, or software applications that allow users to draw or paint directly on a computer screen. They can also be generated automatically by machines, such as in medical imaging or satellite imagery.

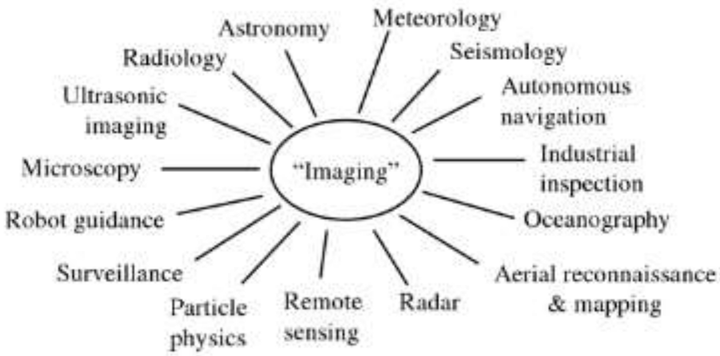


Figure 1: Definition of images

3 Digital images:

A digital image is an image whose surface is divided into elements of fixed size called cells or pixels, each having the characteristic of a level of gray or Color

[1].

To digitize an image is to convert it from an analog state into an image. Digital is represented by a two-dimensional (x,y) matrix of numeric values, as shown in figure 2 :

[1] Mohamed SANDELI, traitement d'images par des approches bio-inspirées application à la segmentation d'images, diplôme de Magister en informatique, université Constantine 2. 2014.

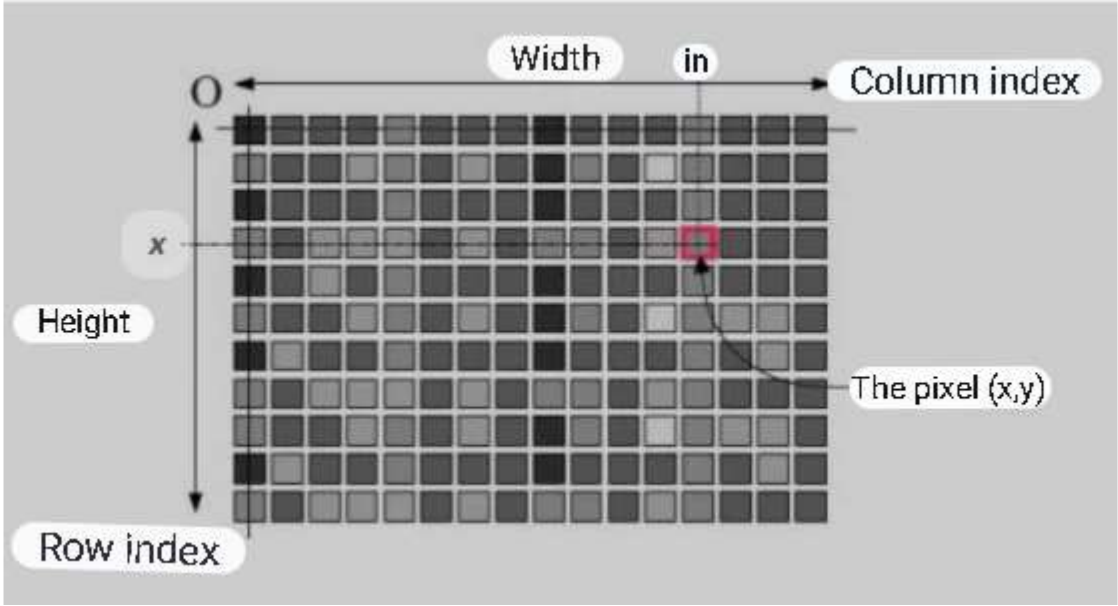


Figure 2: Digital image representation

- x, y: Cartesian coordinates of the image point.

- f (x, y): intensity level.

The value of each point represents the measurement of light intensity perceived by the sensor.

4 Acquiring an image

Image acquisition is one of the essential links in any image design and production chain. To be able to manipulate an image on a system computer, it is above all necessary to make it undergo a transformation that will make it readable and manipulated by this system. The transition from this external object (the original image) to its internal representation (in the processing unit) is done thanks to a procedure of scanning. These input systems, called optical, can be classified into two main categories: digital cameras and scanners[2].

5 Image features

Pixels

A pixel is the part of the digital image file that contains the image of the object.

The pixel can display many different shades of gray or many colors.

Pixels are little pieces of information that when put together make an image file.

Pixels are often called dots, as in dots per inch (dpi).

The use of the word dot was adopted to help people understand what pixels really are[3].

[2] http://thesis.univbiskra.dz/2271/6/Chapitre%2003.pdf

[3] Digital Imaging: A Practical Approach

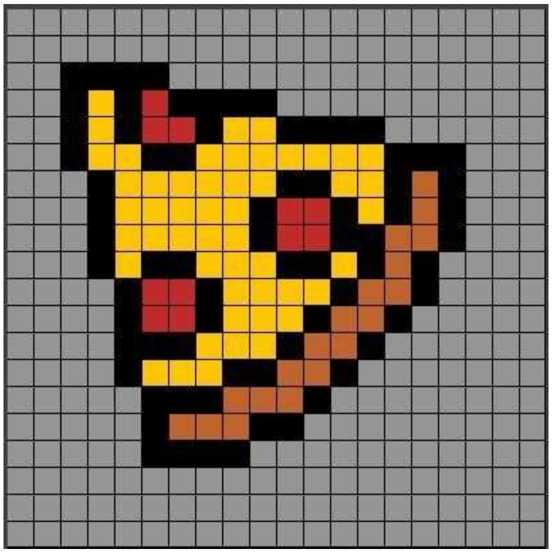


Figure 3: Group of pixels

Resolution

Resolution is the maximum number of **pixels** the **scanner** can record without interpolation or guessing information.

Optical resolution is true information. It is dependent on the sophistication of the scanner. High-end scanners will have optical resolution in the range of 2000 dpi while low-end will be closer to 600 dpi.

Dimension

The number of pixels in a digital image file. For example: 640 pixels wide by 480 pixels high[4].

RGB

RED, GREEN, BLUE. RGB is the setting most scanners and cameras use to capture color image file. The **CCD** itself is designed to interpret only red green and blue. Every color image file created is a mixture of those three colors. Think of the painter's palette and mixing paint colors to create new ones. In the sense that is what happens when an image file is created[5].

[4] Digital Imaging: A Practical Approach

[5] Digital Imaging: A Practical Approach

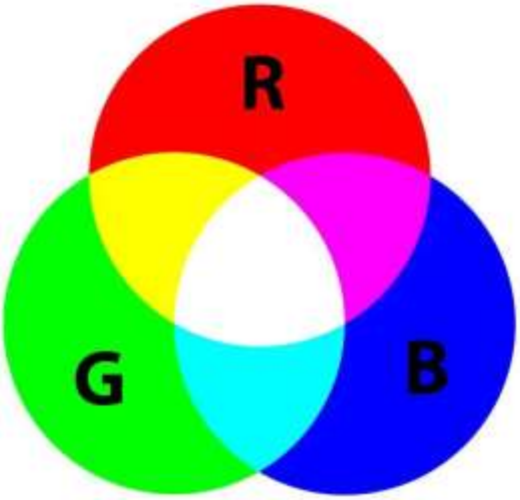


Figure 4: The color spectrum

Contrast

The marked difference between two regions of an image, plus precisely between the dark regions and the light regions of this image.

The contrast is defined according to the lighting of the two image areas[6].

If L1 and L2 are the degrees of luminosity respectively of two neighboring zones A1 and A2 of an image, the contrast C is defined by the ratio[7]:

$$C = \frac{\mathrm{L1} - \mathrm{L2}}{L1 + L2}$$

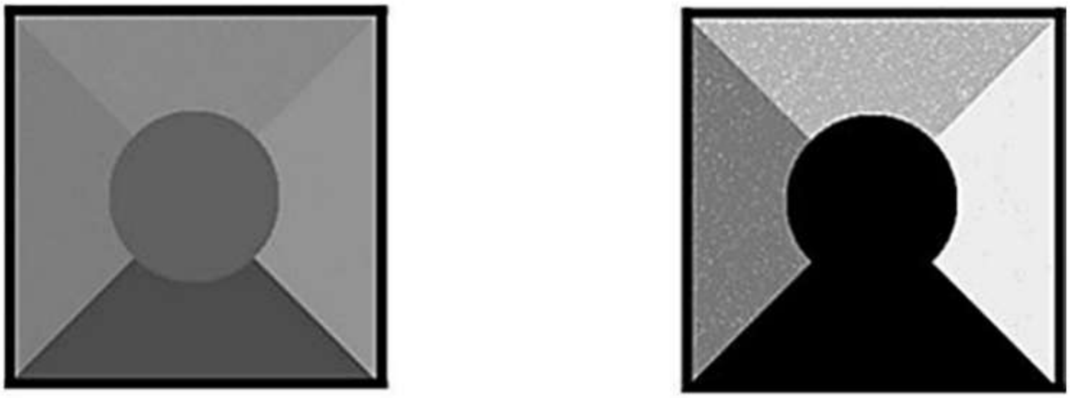


Figure 5: Images with different contrast

[6] Classification Automatique des images de Nourritures, master Réseaux et télécommunication

[7] Mémoire Master Professionnel,Domaine : Informatique et Technologie de l'Information,Spécialité : Réseaux convergence et sécurité,Présenté par : Sarra BENFRIHA et Asma HAMEL

Histogram (Grayscale)

The grayscale or color histogram of an image is a function that gives the frequency of appearance of each level of gray (color) in the image. For reduce the quantization error, to compare two images obtained under lightings different, or to measure certain properties on an image.

It allows giving a large amount of information on the distribution of gray levels (color) and to see between which limits are distributed the majority of gray levels (color) in the case of an image that is too light or an image that is too dark.

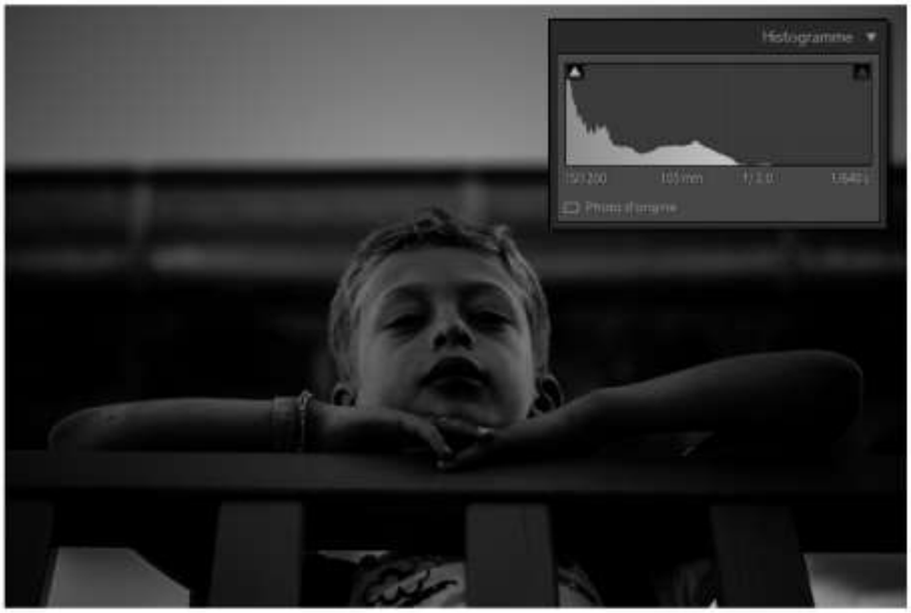


Figure 6: Grayscale

6 Image processing

The processing and analysis of digital images designed by the discipline of information and mathematical applications that study digital images and their transformations, for a specific purpose that is to improve their quality or to extract information from them.

It is a subset of signal processing and analysis given by images and derived data like video (as opposed to the signal processing parts dedicated to other types of data: sound and other one-dimensional signals particular), while operating in the digital domain (as opposed to technical signal processing analogues, such as traditional photography or television).

In the context of artificial vision, the line of images is placed after its stages of acquisition and digitization, ensuring the image transformations and the calculation part allowing going towards an interpretation of the preserved images. This interpretation phase is more and more integrated into image processing, in particular by calling on artificial intelligence to manipulate knowledge, mainly about

The information does not have these unique traits hear images (Domain knowledge)[8].

[8] Noura Zouad, “Généralités sur le traitement d’images”

Historical

The field of image processing can be traced back to the mid-20th century, when digital image processing techniques were first developed for use in military applications such as reconnaissance and target tracking. During this time, researchers began to explore techniques for image filtering, noise reduction, and edge detection, among other basic image processing tasks.

In the 1970s and 1980s, the field of image processing began to expand beyond its military origins and into a wider range of applications, including medical imaging, remote sensing, and industrial inspection. Researchers developed new algorithms and techniques for image enhancement, feature extraction, and pattern recognition, among other tasks.

In the 1990s and 2000s, the field of image processing continued to evolve, with the emergence of new technologies such as digital cameras, computer vision systems, and machine learning algorithms. Researchers began to explore more advanced applications of image processing, such as object recognition, 3D imaging, and augmented reality[9].

Today, the field of image processing continues to grow and evolve, with new applications and technologies emerging all the time. Some of the most exciting areas of research include deep learning algorithms for image recognition and semantic segmentation, as well as applications of image processing in fields such as autonomous vehicles, healthcare, and entertainment.

The Different Possible Transformations on the Digital Image

We can distinguish four types or modes of transformations, which can be carried out at from or to a digital image:

6.1.1 A. Image Processing

This refers to operations performed on an image to enhance or modify it. Image processing techniques can include filtering, noise reduction, image restoration, image enhancement, and color correction. The goal is to improve the visual quality or extract specific features from the image.

6.1.2 B. Image Analysis

Image analysis involves extracting meaningful information or measurements from an image. This can include tasks such as object detection, segmentation, pattern recognition, feature extraction, and image classification. Image analysis techniques are used to understand and interpret the content of the image in terms of its semantic meaning.

6.1.3 C. Image High-level Description

This type of transformation involves going beyond individual objects or features and aims to provide a higher-level understanding of the image. It typically involves image understanding or recognition, where the goal is to identify objects or scenes within an image and assign them semantic labels or object classes. This can be achieved using techniques such as deep learning, convolutional neural networks (CNNs), or other machine learning algorithms.

6.1.4 D. Image Reconstruction

Image reconstruction involves generating a new image or representation from a set of information or operators. It can include tasks such as image synthesis, image inpainting (filling in missing regions), image compression, or image transformation based on mathematical models or algorithms. Image reconstruction techniques can be used for various purposes, such as data compression, data transmission, or creating visually appealing images.

[9] Mr. Karim Mezzoug, “TRAITEMENT ET ANALYSE DES IMAGES NUMERIQUES”,

These four types of transformations encompass different aspects of manipulating and analyzing digital images, each serving a specific purpose. The choice of transformation depends on the desired outcome and the specific application or task at hand.

Image processing system

An image processing system is generally composed of the following units:

1. An acquisition and digitization system that allows sampling and quantization of an image.
2. A mass memory to store digitized images.
3. A visualization system.
4. A central unit to perform the various processing operations[10].

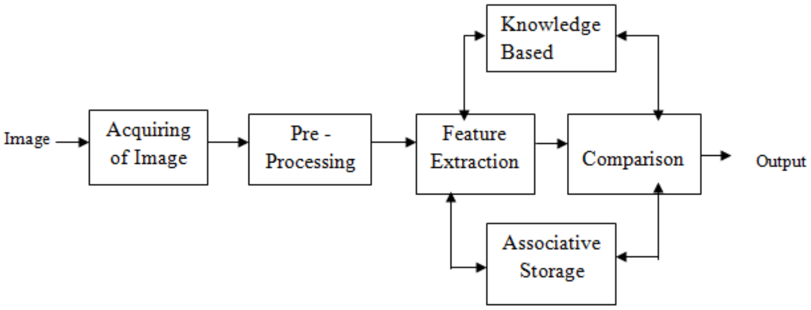


Figure 7: image processing system

Preprocessing system

Image preprocessing is a step in every pattern recognition system to improve its performance and used to reduce variations and produce a more consistent set of data.

It is the operations on images at the lowest level of abstraction which doesn't increase image information content.

The purpose of image preprocessing is to make the data ready for machine learning models by standardizing and cleaning up the acquired data, which are usually messy and come from different sources[11].

Image preprocessing involves cleaning raw images into a format that best improves the model.

There are several types of images preprocessing techniques, including:

- Standardizing images.
- Resizing images to a unified dimension.
- Converting colored images to grayscale.

[10] Mémoire Master Professionnel,Domaine : Informatique et Technologie de l'Information,Spécialité : Réseaux convergence et sécurité,Présenté par : Sarra BENFRIHA et Asma HAMEL

[11] Deep Learning for Vision Systems by Mohamed Elgendy

- Removing unwanted or noisy data.

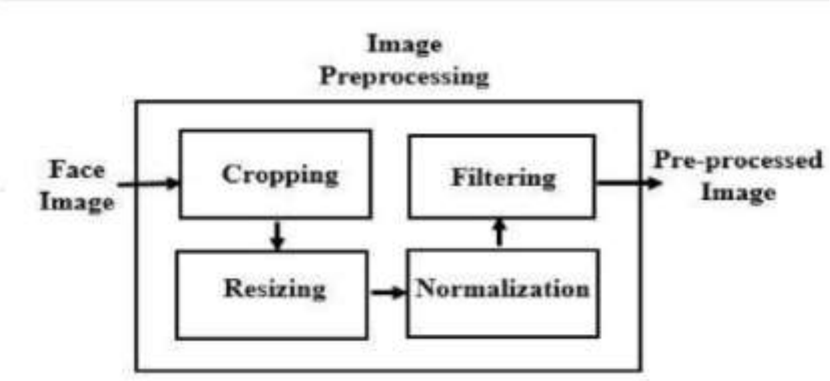


Figure 8:preprocessing system

7 Image segmentation

The process of partitioning an image into component regions or objects, so as to change the representation of an image into something that is more easier to analyze in order to get more information in the region of interest in an image which helps in annotation of the object scene is referred to as image segmentation. Image segmentation is needed in order to identify the content of the image carefully. In this context, edge detection is a fundamental tool for image segmentation. Several general-purpose algorithms and techniques have been developed for image segmentation[12].

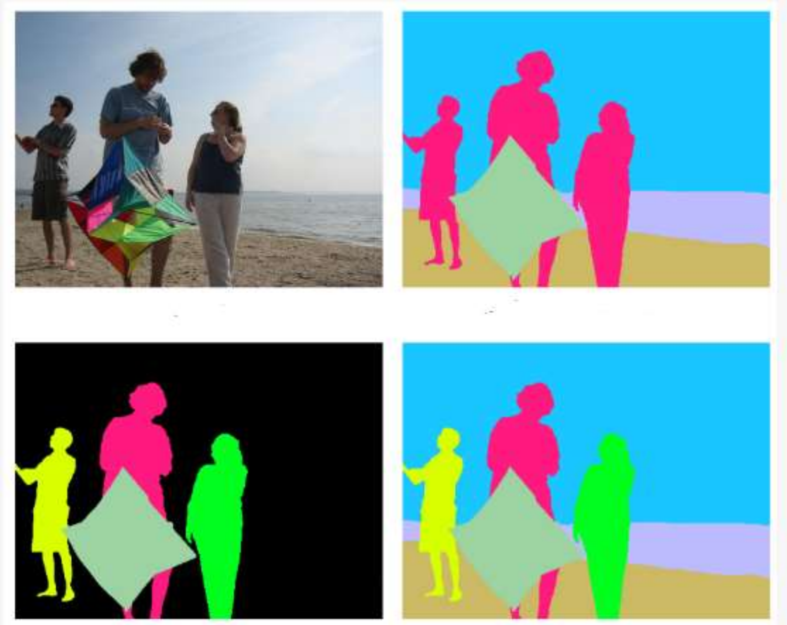


Figure 9:Example of image segmentation

Even though different segmentation techniques are at hand, every method is not equally appropriate for a particular type of image. Thus, the algorithm suitable for one class of image may not be suitable for another class of images.

[12] A Review on Image Processing and Image Segmentation, Jiss Kuruvilla, Dhanya Sukumaran, Anjali Sankar, Siji P Joy

Consequently, there is no unanimously supported method for image segmentation for all categories of images and as a result it remains a challenge in image processing and computer vision.

Segmentation can be classified as follows:

•Region Based

•Edge Based

•Threshold

•Feature Based Clustering

8 Computer vision

Image processing using AI, also known as computer vision, involves using machine learning algorithms to analyze and interpret digital images or video. This can include a wide range of tasks, from basic image processing techniques like filtering and enhancement, to more advanced applications like object detection, image recognition, and semantic segmentation.

Computer vision researchers, in parallel, are using mathematical techniques to restore the three-dimensional shape and appearance of objects in images. Here, progress in the past two decades has been rapid. We now have reliable techniques to compute a 3D model of a microenvironment from thousands of partially overlapping images. Given a large enough set of views of a given object or interface, we can generate accurate 3D dense surface models using stereo matching. We can even, with moderate success, identify most of the people and objects in a photo. However, despite all these advances, the dream of having a computer explain an image with the same level of detail and causality as a two-year-old remain elusive[13].

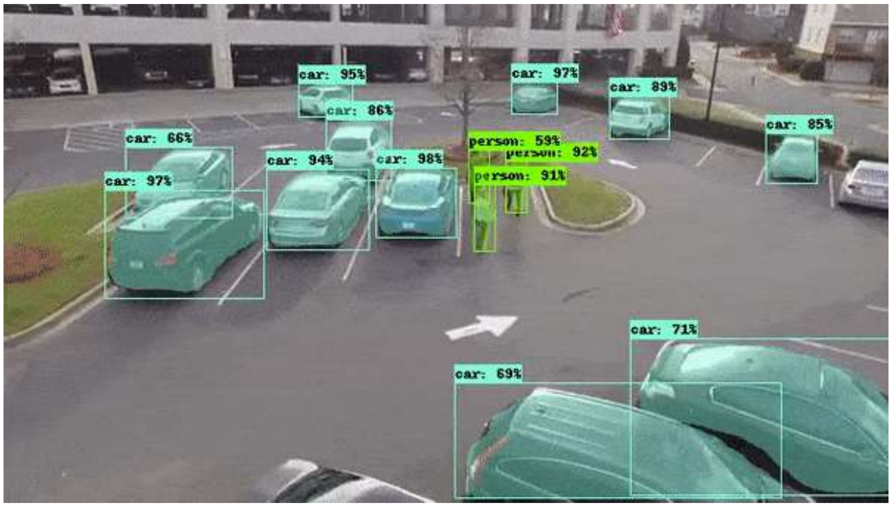


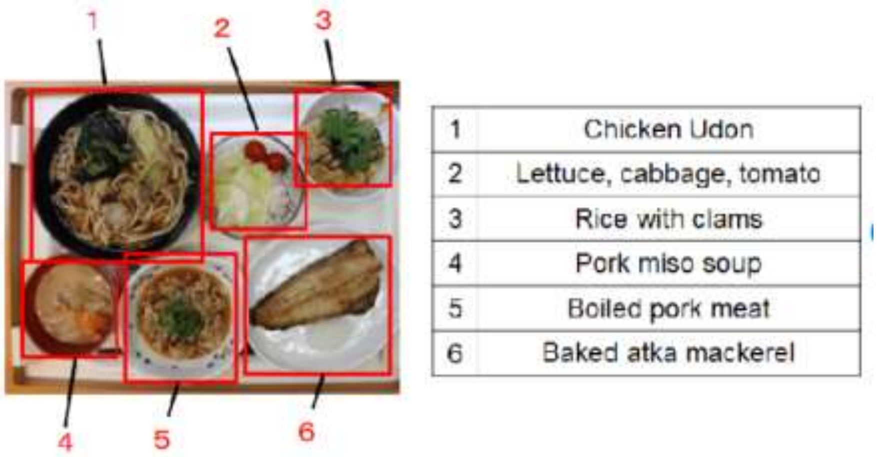
Figure 10 : computer vision in the image recognition field

[13] Computer Vision: Algorithms and Applications De Richard Szeliski

9 Image recognition

In a brief definition of image recognition, also known as image classification is a field within computer vision that focuses on identifying and classifying objects or patterns within digital images or videos. Includes training algorithms to analyze and understand visual information in order to recognize and assign labels to various objects or scenes depicted in the images.

Image recognition has numerous applications across various domains, including autonomous vehicles, facial recognition, medical imaging, surveillance systems, and object detection in robotics and many others.



1	Chicken Udon
2	Lettuce, cabbage, tomato
3	Rice with clams
4	Pork miso soup
5	Boiled pork meat
6	Baked atka mackerel

Figure 11: Example of the result of food recognition

10 Conclusion

In conclusion, this chapter provides a brief introduction to the basic concepts in the field of image processing, which play a crucial role in improving the quality of digital images by reducing noise and detecting certain shapes and bringing them closer to the original image. This prepares images for use in various applications. By leveraging synergies between physical data, mathematical modeling, and image processing technologies, we can enhance the accuracy and effectiveness of pattern recognition systems, enabling them to make informed predictions and facilitating further advances in various fields

Chapter I I

Artificial Intelligence and Machine Learning

1 Introduction

In this chapter, we present an overlook about the basic notions of artificial intelligence (AI). AI is a field of study focused on creating intelligent machines capable of performing tasks that typically require human intelligence. It has gained significant importance in recent years due to its potential to revolutionize various industries and improve our lives. We will explore the historical development of AI, key concepts such as intelligent agents, **machine learning** and its algorithms, **deep learning**, and computer vision. We will also touch upon ethical and societal considerations associated with AI. This chapter aims to provide a foundational understanding of AI and its relevance in today's world.

2 Artificial Intelligence

2.1 Definition

Artificial intelligence (AI) is the ability of machines to replicate or enhance human intellect, such as reasoning and learning from experience. Artificial intelligence has been used in computer programs for years, but it is now applied to many other products and services. For example, some digital cameras can determine what objects are present in an image using artificial intelligence software. In addition, experts predict many more innovative uses for artificial intelligence in the future, including smart electric grids.

AI uses techniques from probability theory, economics, and algorithm design to solve practical problems. In addition, the AI field draws upon computer science, mathematics, psychology, and linguistics. Computer science provides tools for designing and building algorithms, while mathematics offers tools for modeling and solving the resulting optimization problems

[14].

Figure 12: Artificial intelligence

2.2 History of Artificial Intelligence

The history of artificial intelligence (AI) began in antiquity, with myths, stories and rumors of artificial beings endowed with intelligence or consciousness by master craftsmen. The seeds of modern AI were planted by philosophers who attempted to describe the process of human thinking as the mechanical manipulation of symbols. This work culminated in the invention of the programmable digital computer in the 1940s, a machine based on the

[14] https://www.red-gate.com/simple-talk/development/data-science-development/introduction-to-artificial-intelligence/

abstract essence of mathematical reasoning. This device and the ideas behind it inspired a handful of scientists to begin seriously discussing the possibility of building an electronic brain.

The field of AI research was founded at a workshop held on the campus of Dartmouth College, USA during the summer of 1956. Those who attended would become the leaders of AI research for decades. Many of them predicted that a machine as intelligent as a human being would exist in no more than a generation, and they were given millions of dollars to make this vision come true.

Eventually, it became obvious that commercial developers and researchers had grossly underestimated the difficulty of the project In 1974, in response to the criticism from James Lighthill and ongoing pressure from congress, the U.S. and British Governments stopped funding undirected research into artificial intelligence, and the difficult years that followed would later be known as an "AI winter". Seven years later, a visionary initiative by the Japanese Government inspired governments and industry to provide AI with billions of dollars, but by the late 80s the investors became disillusioned and withdrew funding again.

Investment and interest in AI boomed in the first decades of the 21st century when machine learning was successfully applied to many problems in academia and industry due to new methods, the application of powerful computer hardware, and the collection of immense data sets15.

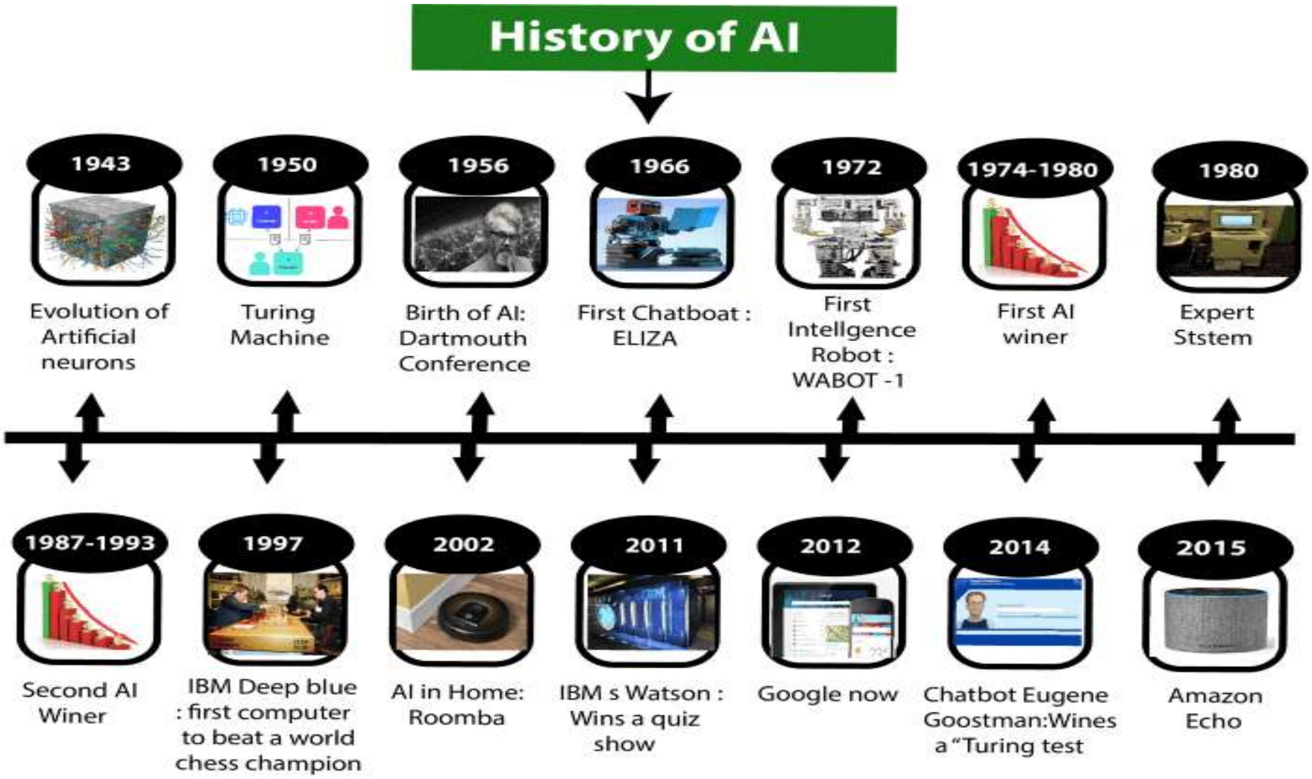

History of AI
1943
Evolution of Artificial neurons
1950
Turing Machine
1956
Birth of AI: Dartmouth Conference
1966
First Chatboat : ELIZA
1972
First Intellgence Robot : WABOT -1
1974-1980
First AI winer
1980
Expert Ststem
1987-1993
Second AI Winer
1997
IBM Deep blue : first computer to beat a world chess champion
2002
AI in Home: Roomba
2011
IBM s Watson : Wins a quiz show
2012
Google now
2014
Chatbot Eugene Goostman:Wines a "Turing test
2015
Amazon Echo

Figure 13: History of Artificial Intelligence

2.3 What is Artificial Intelligence

Artificial intelligence (AI), the ability of a digital computer or computer-controlled robot to perform tasks commonly associated with intelligent beings. The term is frequently applied to the project of developing systems

[15] https://en.wikipedia.org/wiki/History_of_artificial_intelligence

endowed with the intellectual processes characteristic of humans, such as the ability to reason, discover meaning, generalize, or learn from past experience. Since the development of the digital computer in the 1940s, it has been demonstrated that computers can be programmed to carry out very complex tasks—as, for example, discovering proofs for mathematical theorems or playing chess—with great proficiency. Still, despite continuing advances in computer processing speed and memory capacity, there are as yet no programs that can match human flexibility over wider domains or in tasks requiring much everyday knowledge. On the other hand, some programs have attained the performance levels of human experts and professionals in performing certain specific tasks, so that artificial intelligence in this limited sense is found in applications as diverse as medical diagnosis, computer search engines, and voice or handwriting recognition.

Figure 14:https://static.javatpoint.com/tutorial/ai/images/introduction-to-artificial-intelligence.png

2.4 Types of Artificial Intelligence

Artificial Intelligence can be divided in various types, there are mainly two types of main categorization which are based on capabilities and based on functionally of AI. Following is flow diagram which explain the types of AI [16].

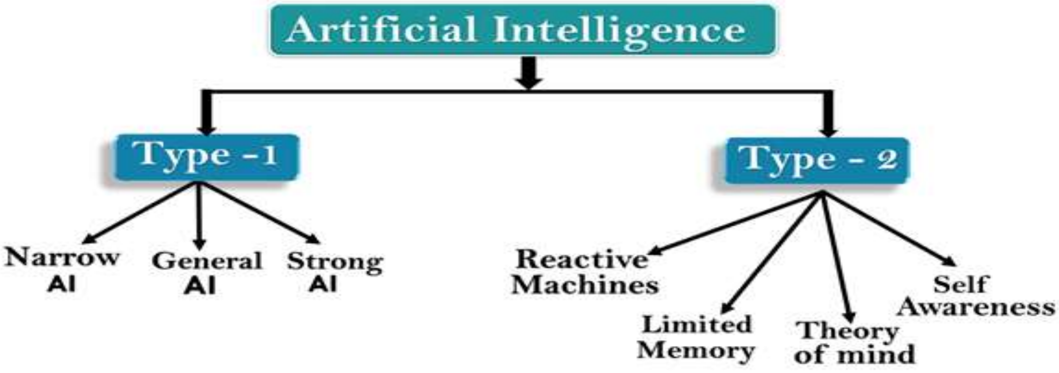


Figure 15:https://static.javatpoint.com/tutorial/ai/images/types-of-artificial-intelligence.png

- **Artificial Narrow Intelligence (ANI):** The only sort of artificial intelligence we have effectively generated yet is artificial narrow intelligence (ANI), often known as weak AI or narrow AI. Narrow AI is goal-oriented, designed to accomplish a single task – such as facial recognition, speech recognition/voice assistants, driving a car, or surfing the internet – and is extremely competent at fulfilling the objective.

[16] https://www.javatpoint.com/types-of-artificial-intelligence

- **Artificial General Intelligence (AGI):** Artificial general intelligence (AGI), also known as strong AI or deep AI, is the idea of a machine with general intelligence that can learn and apply its intelligence to solve any problem. In every given context, AGI can think, understand, and act in a manner that is indistinguishable from that of a human.

- **Artificial Super Intelligence (ASI):** Artificial super intelligence (ASI) is a hypothetical AI that does more than mimic or understands human intelligence and behavior; ASI is when computers become self-aware and exceed human intelligence and ability[17].

2.5 Why uses AI?

AI can achieve higher quality outcomes faster than humanly possible.

Today, AI is most often used to recognize patterns, make predictions, and provide insights previously out of reach due to the sheer amount of available data. It's able to do this because, unlike traditional computer technologies, AI is able to learn from examples as opposed to being explicitly programmed to execute specific instructions.

These systems are meant to augment our own intelligence and maximize our confidence. In a growing number of fields, AI is serving as a companion for professionals to enhance performance and reduce the time required to become an expert. It will aid in the pursuit of knowledge, to further our expertise, and to improve the human condition 18.

Common AI use cases

AI is a powerful toolbox that has many applications in domains far and wide. The types of problems that the AI toolbox is best equipped to solve can be split into six core intents, as described on IBM's Watson site:

1. Accelerate research and discovery
2. Enrich your interactions
3. Anticipate and preempt disruptions
4. Recommend with confidence
5. Scale expertise and learning
6. Detect liabilities and mitigate risk

Some of the most common task's AI performs — and their corresponding subfields — include:

- Extracting information from pictures (computer vision)
- Transcribing or understanding spoken words (speech to text and natural language processing)
- Pulling insights and patterns out of written text (natural language understanding)
- Speaking what's been written (text to speech, natural language processing)
- Autonomously moving through spaces based on its senses (robotics)
- Generally looking for patterns in heaps of data (machine learning)

[17] https://www.interviewbit.com/blog/applications-of-artificial-intelligence/?amp=1
[18] https://www.ibm.com/design/ai/basics/ai

2.6 Application of AI

Artificial Intelligence has various applications in today's society. It is becoming essential for today's time because it can solve complex problems with an efficient way in multiple industries, such as Healthcare, entertainment, finance, education, etc. AI is making our daily life more comfortable and faster.

Following are some sectors which have the application of Artificial Intelligence[19]:

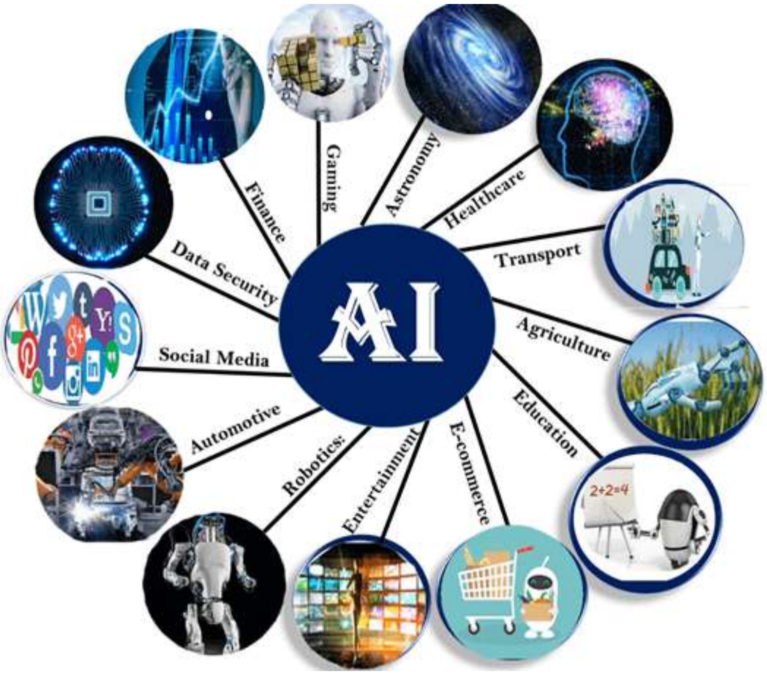


Figure 16:https://static.javatpoint.com/tutorial/ai/images/application-of-ai.png

3 Machine Learning

3.1 Definition

Machine learning is a branch of artificial intelligence (AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy[20].

It is a type of AI that allows software applications to become more accurate at predicting outcomes without being explicitly programmed to do so.

Machine learning uses two types of techniques: supervised learning and unsupervised learning

[19] https://www.techtarget.com/searchenterpriseai/definition/AI-Artificial-Intelligence
[20] https://www.ibm.com/topics/machine-learning

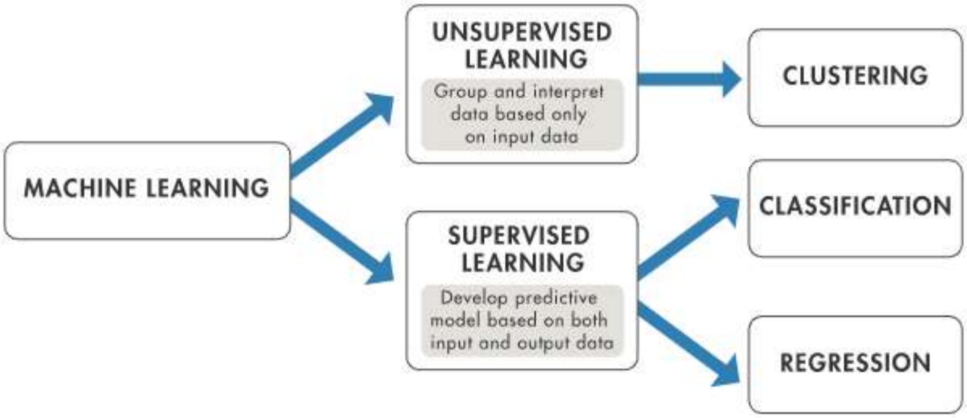

MACHINE LEARNING
UNSUPERVISED LEARNING
Group and interpret data based only on input data
SUPERVISED LEARNING
Develop predictive model based on both input and output data
CLUSTERING
CLASSIFICATION
REGRESSION

Figure 17: machine learning branches

3.2 Unsupervised learning

Unsupervised learning finds hidden patterns or intrinsic structures in data. It is used to draw inferences from datasets consisting of input data without labeled responses[21].

3.2.1 Clustering

is the most common unsupervised learning technique. It is used for exploratory data analysis to find hidden patterns or groupings in data. Applications for cluster analysis include gene sequence analysis, market research, and object recognition.

Clustering is used to identify different groups/segments of customers and to profile the attributes of different groups.

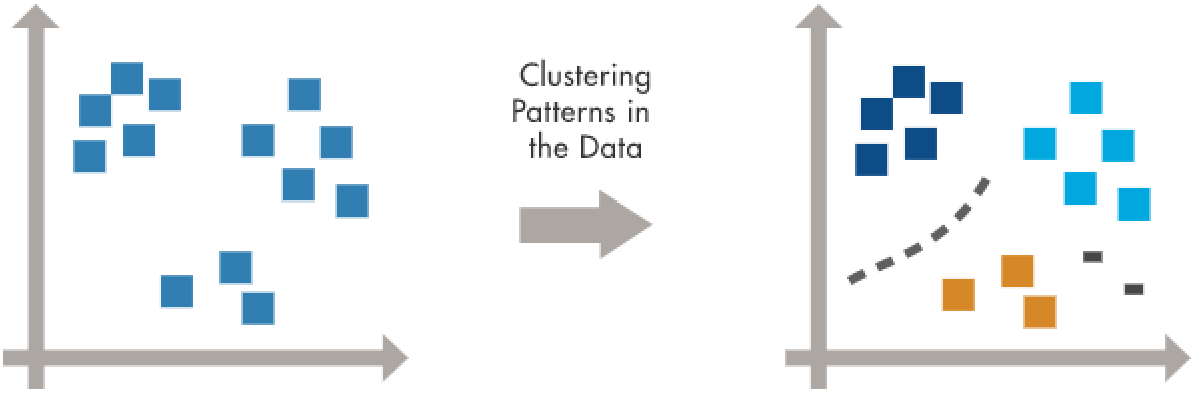


Figure 18: Clustering pattern

There are many algorithms developed to implement this technique but we are going to mention the most popular and widely used algorithms in machine learning[22].

1. K-mean Clustering.
2. Hierarchical Clustering.

[21] https://www.mathworks.com/discovery/machine-learning.html
[22] https://towardsdatascience.com/clustering-unsupervised-learning-788b215b074b

3.2.1.1 K-mean Clustering

1. It starts with K as the input which is how many clusters you want to find. Place K centroids in random locations in your space.
2. Now, using the Euclidean distance between data points and centroids, assign each data point to the cluster which is close to it.
3. Recalculate the cluster centers as a mean of data points assigned to it.
4. Repeat 2 and 3 until no further changes occur.

Now, you might be thinking that how do I decide the value of K in the first step.

One of the methods is called “Elbow” method can be used to decide an optimal number of clusters. Here you would run K-mean clustering on a range of K values and plot the “percentage of variance explained” on the Y-axis and “K” on X-axis.

In the picture below you would notice that as we add more clusters after 3 it doesn't give much better modeling on the data. The first cluster adds much information, but at some point, the marginal gain will start dropping.

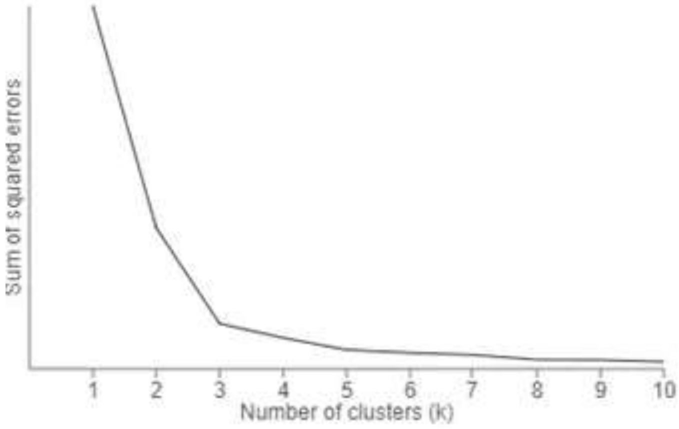


Figure 19: Elbow method

3.2.1.2 Hierarchical Clustering

Unlike K-mean clustering Hierarchical clustering starts by assigning all data points as their own cluster. As the name suggests it builds the hierarchy and in the next step, it combines the two nearest data point and merges it together to one cluster.

1. Assign each data point to its own cluster.
2. Find closest pair of clusters using Euclidean distance and merge them in to single cluster.
3. Calculate distance between two nearest clusters and combine until all items are clustered in to a single cluster.

In this technique, you can decide the optimal number of clusters by noticing which vertical lines can be cut by horizontal line without intersecting a cluster and covers the maximum distance[23].

3.3 Supervised learning

Supervised learning, as the name indicates, has the presence of a supervisor as a teacher. Basically supervised learning is when we teach or train the machine using data that is well-labeled. Which means some data is already tagged with the correct answer. After that, the machine is provided with a new set of examples(data) so that the supervised learning algorithm analyses the training data (set of training examples) and produces a correct outcome from labeled data.

3.3.1 Classification

Classification models in machine learning are used to predict discrete outcomes or assign input data into specific categories. These models are particularly useful in applications such as Email Spam Detection or Medical Imaging for example whether a tumor is cancerous or benign.

There are so many machine learning algorithms for classification problems we like **decision tree, random forest, naïve bayass.**

3.3.1.1 Decision Tree

Decision tree builds tree branches in a hierarchy approach and each branch can be considered as an if-else statement. The branches develop by partitioning the dataset into subsets based on most important features. Final classification happens at the leaves of the decision tree.

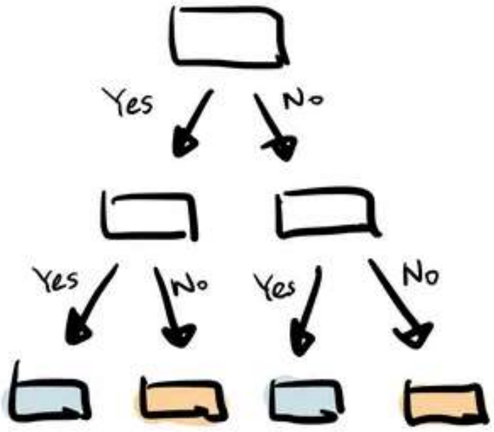


Figure 20: Decision tree

3.3.1.2 Random Forest

Random forest is a collection of decision trees. It is a common type of ensemble methods which aggregate results from multiple predictors. Random forest additionally utilizes bagging technique that allows each tree trained on a random sampling of original dataset and takes the majority vote from trees.

[23] https://towardsdatascience.com/clustering-unsupervised-learning-788b215b074b

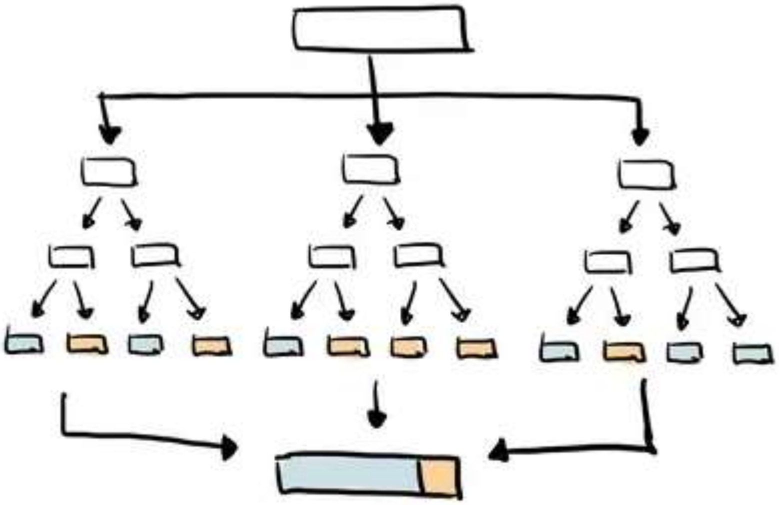


Figure 21: Random Forest

3.3.2 Regression:

As for regression, it is to predict continuous numerical values. It involves creating a mathematical model that establishes a relationship between the input variables (features or independent variables) and the output variable (dependent variable). The main objective of regression is to analyze and quantify the relationship between these variables, enabling the prediction of future values.

In the figure below, each type and the most popular algorithms used in it.

3.4 Semi-supervised learning

Semi-supervised learning deals with the situation where relatively few labeled training examples are available, but a large number of unlabeled points are given, it is directly relevant to a multitude of practical problems where it is relatively expensive to produce labeled data.

Learning problems of this type are challenging as neither supervised nor unsupervised learning algorithms are able to make effective use of the mixtures of labeled and untellable data. As such, specialized semis-supervised learning algorithms are required.[24]

3.5 Reinforcement Learning

In addition to the supervised and the unsupervised types, another important type of machine learning is reinforcement learning.

Reinforcement learning is a type of machine learning that involves an agent learning to make decisions and take actions in an environment to maximize a reward signal. It is particularly useful in situations where there is no labeled dataset available for training, and the agent learns through trial and error.

[24] Olivier Chapelle Semi-Supervised Learning (Adaptive Computation and Machine Learning series)

In the context of pattern recognition, reinforcement learning can be applied to tasks such as image classification, object detection, or even image generation. The agent interacts with the environment, which in this case includes the input images, and learns to make decisions or predictions based on the observed rewards or feedback.[25]

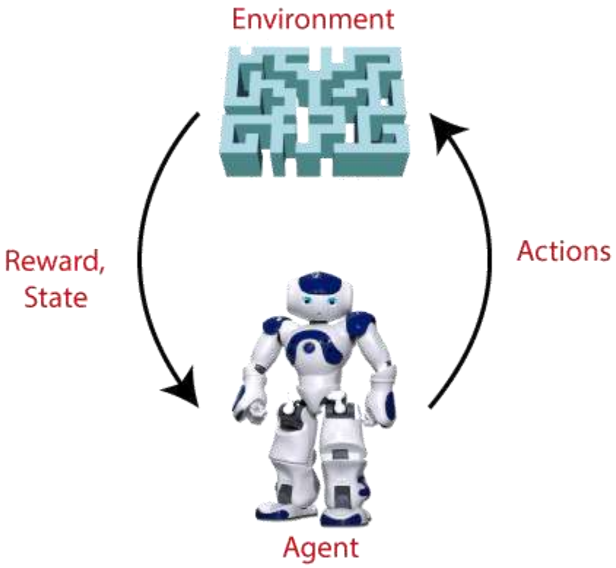


Figure 22: a simple example of reinforcement learning

4 Artificial Neural network

4.1 Definition

Neural network is a series of algorithms that works similarly to the human brain's neural network. A "neuron" in a neural network is a mathematical function that collects and classifies information according to a specific architecture[26].

Neural networks are sometimes described in terms of their depth, including how many layers they have between input and output, or the model's so-called hidden layers.

[25] https://www.coursera.org/specializations/reinforcement-learning
[26] https://www.investopedia.com/terms/n/neuralnetwork.asp by JAMES CHEN.

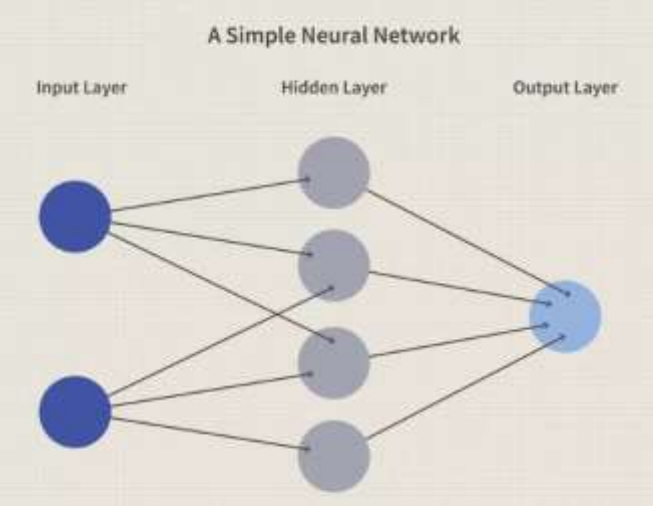


Figure 23 A simple neuron network

4.2 Artificial Neural Networks (ANN) VS Biological Neural Networks (BNN):

4.2.1 Biological Neural Networks

In human (animal) body, neurons are nerve cells. A neuron has branch called the **dendrites** through which the information coming from outwards to the **cell body**. The neuron processes this information and returns the result to the through its **axon**. This signal emitted by the neuron can then be transmitted, through from a **synapse**, to another neuron, or to a muscle or a gland.

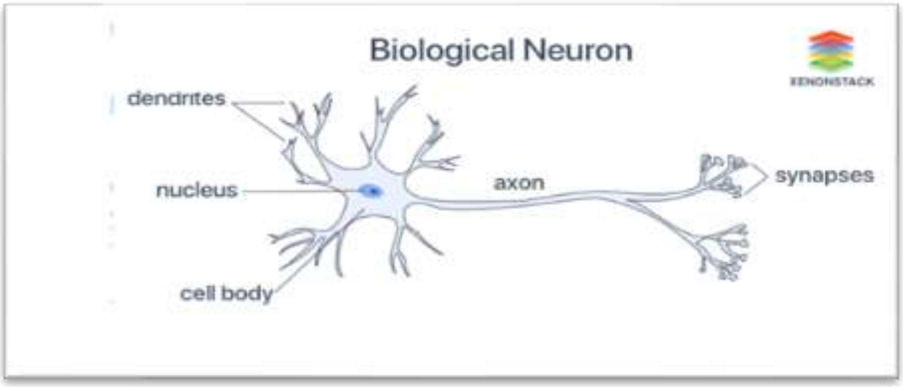


Figure 24: Biological neuron network[27]

Neural Networks resemble the human brain in the following two ways:

- A neural network acquires knowledge through learning.
- A neural network's knowledge is a store within inter-neuron connection strengths known as synaptic weights.

[27] https://www.xenonstack.com/blog/artificial-neural-network-applications.

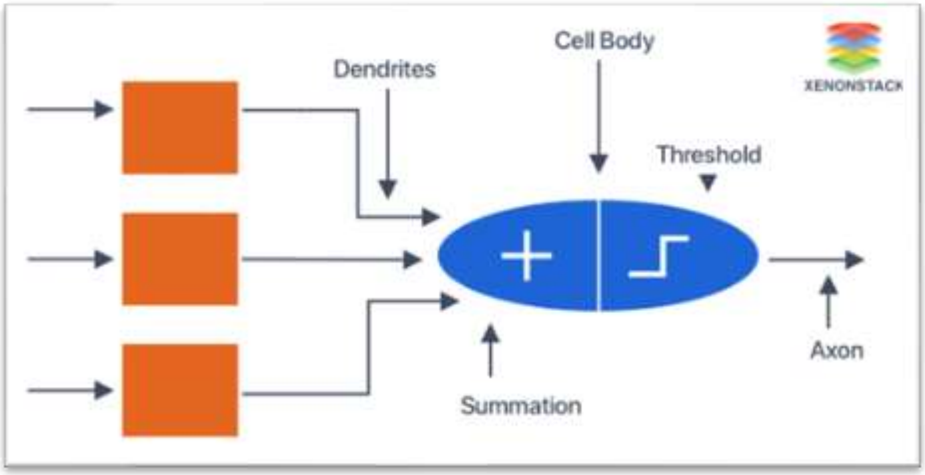


Figure 25: Artificial neuron network

4.3 How Do Neural

Neural network-based machine learning algorithms typically do not require programming with precise rules defining what to anticipate from the input. Instead, the neural network learning algorithm learns by analyzing many labeled examples provided during training and by utilizing this answer key to determine what qualities of the input are required to generate the proper output. The neural network can start processing new, unknown inputs and effectively produce correct results once a sufficient number of examples have been processed[28].

4.4 Neural network types

4.4.1 Recurrent Neural Network (RNN)

Recurrent neural networks are constructed to comprehend temporal or sequential data. RNNs improve their predictions by using additional data points in a sequence. To modify the output, they take in input and reuse the activations of earlier or later nodes in the sequence.

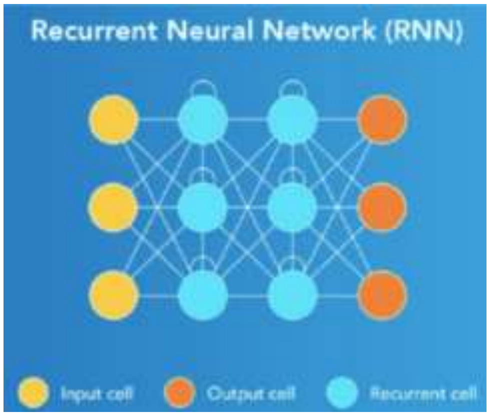


Figure 26: Recurrent Neural Network[29]

4.4.2 Multilayer Perceptron (MLP)

A multilayer perceptron is a fully convolutional network that creates a collection of outputs from a set of inputs. A directed graph connecting the input and output layers of an MLP is made up of multiple layers of input nodes.

[28] https://www.knowledgehut.com/blog/data-science/types-of-neural-networks#what-are-neural-networks.

[29] https://www.knowledgehut.com/blog/data-science/types-of-neural-networks.

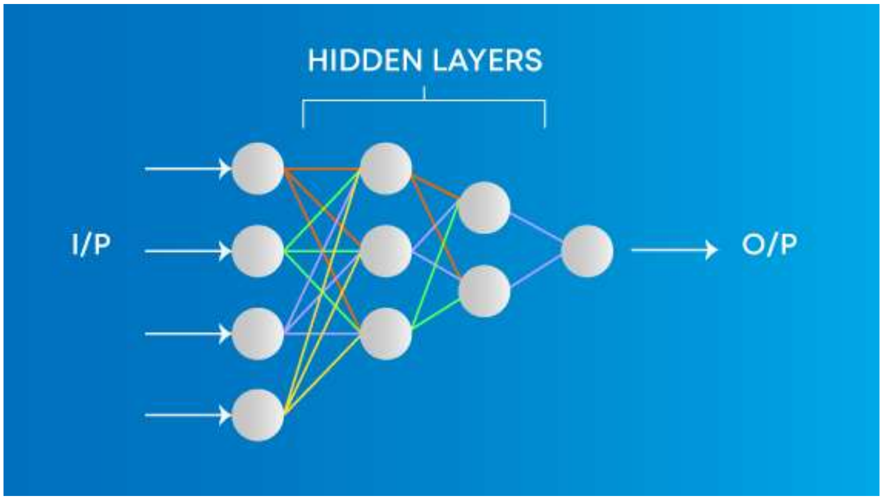


Figure 27: 2 Multilayer Perceptrons (MLP)[30]

4.4.3 Convolutional Neural Network (CNN)

A convolutional neural network (CNN or ConvNet) is another class of deep neural networks. CNNs are most commonly employed in computer vision, given a series of images or videos from the real world, with the utilization of CNN[31].

The neurons in a convolution neural network are arranged in three dimensions rather than the typical two-dimensional array. The convolutional layer refers to the top layer[32].

4.4.3.1 CNN Layers

CNNs typically consist of three main types of layers, which are:

- **Convolutional Layers:**

The convolutional layer is the main building block of a CNN, and it is where the majority of computation occurs. It requires a few items, which are input data, a filter, and a feature map.

Let's suppose that the input will be a color image, which is made up of a matrix of pixels in 3D. This means that the input will have three dimensions a height, width, and depth which correspond to RGB in an image. We also have a filter, which will move across the receptive fields of the image, checking if the feature is present. This process is known as a convolution.

After each convolution operation, a CNN applies a **Rectified Linear Unit (ReLU)** transformation to the feature map, introducing nonlinearity to the model.

ReLU is an activation function commonly used in CNNs. It is a non-linear function applied element-wise to the output of a neuron or a convolutional layer.

- **ReLu Layer:**

Rectified Linear Unit (ReLU) layers are commonly used in Convolutional Neural Networks (CNNs) as activation functions to introduce non-linearity into the network. ReLU is a simple and computationally efficient activation function that has been found to work well in deep learning models, including CNNs.

[30] https://www.mygreatlearning.com/blog/types-of-neural-networks/
[31] https://viso.ai/deep-learning/deep-neural-network-three-popular-types/
[32] https://www.knowledgehut.com/blog/data-science/types-of-neural-networks

The ReLU activation function is defined as

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

Definition of the rectifier activation function

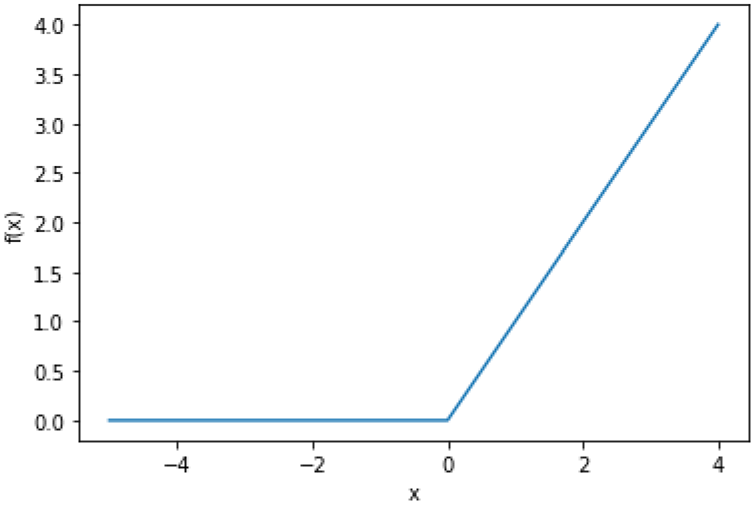


Figure 28: ReLU function

In other words, it returns the input value if it is positive, and zero otherwise. This means that ReLU effectively "turns on" the neurons that have positive input values and "turns off" the neurons with negative input values.

In CNNs, ReLU layers are typically applied element-wise after the convolutional and pooling layers.

Beside of **ReLU** we have another type of function that we can use like:

- **Sigmoid Function**

Sigmoid function is normally used to refer specifically to the logistic function, also called the logistic sigmoid function.

All sigmoid functions have the property that they map the entire number line into a small range such as between 0 and 1, or -1 and 1, so one use of a sigmoid function is to convert a real value into one that can be interpreted as a probability.

Sigmoid functions have become popular in deep learning because they can be used as an activation function in an artificial neural network. They were inspired by the activation potential in biological neural networks[33].

[33] https://deepai.org/machine-learning-glossary-and-terms/sigmoid-function

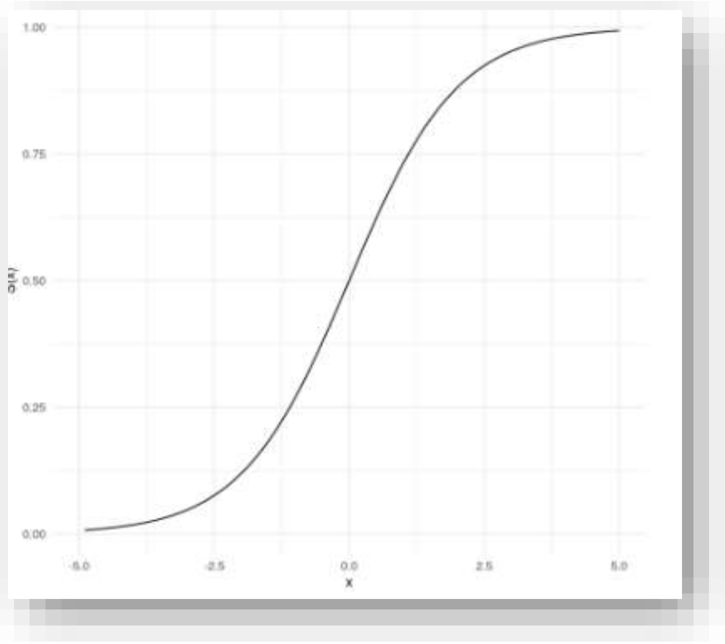


Figure 29 One of the most widely used sigmoid functions

- **Pooling Layers:**

Pooling layers are an essential component in many convolutional neural network (CNN) architectures for image recognition tasks. They help to reduce the spatial dimensions of the input volume while retaining the most important features. Pooling layers operate on individual feature maps and typically involve a down sampling operation.

The two most commonly used types of pooling layers are:

- **Max Pooling**: In this type of pooling, the input feature map is divided into non-overlapping rectangular regions, and the maximum value within each region is selected as the representative value for that region. Max pooling helps to capture the most prominent features while reducing the spatial size of the feature map.
- **Average Pooling:** Unlike max pooling, average pooling computes the average value within each region of the input feature map. It can be useful when you want a more general representation of the features without focusing solely on the most dominant values[34].

- **Fully Connected Layers:**

This layer performs the task of classification based on the features extracted through the previous layers and their different filters. While convolutional and pooling layers tend to use ReLu functions, FC layers usually leverage a softmax activation function to classify inputs appropriately, producing a probability from 0 to 1.

[34] https://www.ibm.com/topics/convolutional-neural-networks

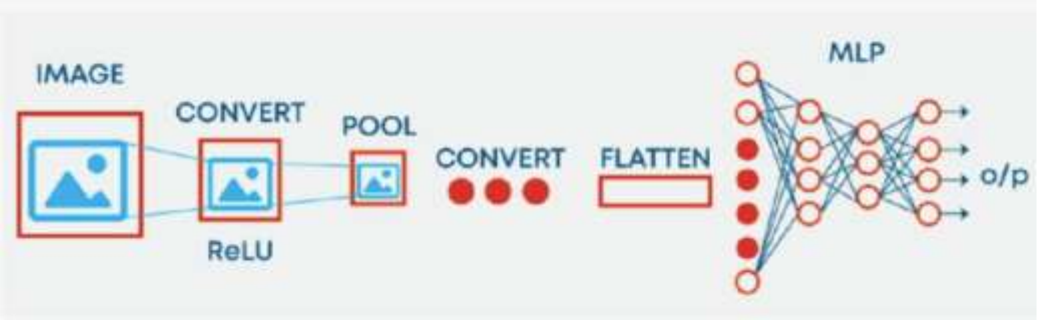


Figure 30 : Convolutional Neural Network

4.4.3.2 Examples of CNN models:

The most common form of CNN architecture stacks a few Conv-ReLU layers, follows them with Pool layers, and repeats this pattern until the input is reduced into a space of a small enough size. At some point, it is common to place fully connected (FC) layers.

There last fully connected layer is connected to the output. Here are some common CNN architectures that follow this pattern:

- INPUT -> CONV -> RELU -> FC
- INPUT -> [CONV -> RELU -> POOL] * 2 -> FC -> RELU -> FC
  Here, there is one layer of **CONV** between each **POOL** layer
- INPUT -> [CONV -> RELU -> CONV -> RELU -> POOL] * 3 -> [FC -> RELU] * 2 -> FC
  Here, there are two stacked **CONV** sofas before each **POOL** layer[35].

5 Deep Learning

5.1 Definition

Deep learning has been used in many tasks and applications today, especially with the amount of data generated every day that allows to use deep learning techniques properly, because it requires a huge amount of data to learn from it, that's what differ deep learning from other approaches.

Deep learning is a subfield of machine learning, inspired from human brain, where nonbiological neural networks used to handle a large amount of data, and learn from it[36].

5.2 Deep Neural Network

Nodes are little components of the system that function similarly to neurons in the human brain. When they are stimulated, a reaction occurs in these nodes. Some are connected and marked, while others are not, although nodes are generally grouped into layers. To complete a task, the system must process layers of data between the input and output. The deeper the network is evaluated, the more layers it must process to obtain the outcome. Credit

[35] Mémoire de fin d'études, diplôme de Master en informatique par Mokri Mohammed Zakaria "Classification des images avec les réseaux de neurons convolutionnels"
[36] Dilmi Abdelbaqi, Chikouche aboubaker, "Failure prediction in cloud environment using deep learning" UNIVERSITY MOHAMED BOUDIAF - M'SILA

Assignment Path (CAP) is a notion that refers to the number of such levels required for the system to finish the assignment. If the CAP index is greater than two, the neural network is considered deep[37].

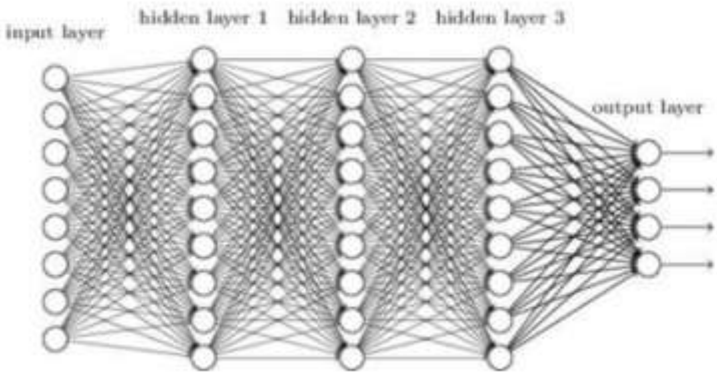


Figure 31: Example of deep neural network

5.3 Advanced techniques in deep neural networks

5.3.1 Inception (GoogLeNet)

The Inception Module is a neural network architecture used in Convolutional Neural Networks (CNNs) to allow for more efficient computation and deeper networks through a dimensionality reduction with stacked 1×1 convolutions. The Inception Module was designed to solve the problem of computational expense, as well as overfitting, among other issues. The Inception Module works by performing a convolution on an input with not one, but three different sizes of filters (1x1, 3x3, 5x5). The Inception Module is incorporated into CNNs as a way of reducing computational expense. The GoogLeNet architecture consists of 22 layers (27 layers including pooling layers), and part of these layers are a total of 9 inception modules. The Inception Module allows the network to choose between multiple convolutional filter sizes in each block. An Inception network stacks these modules on top of each other, with occasional max-pooling layers with stride 2 to halve the resolution of the grid. The Inception-v4 is a more uniform simplified architecture and has more inception modules than Inception-v3[38].

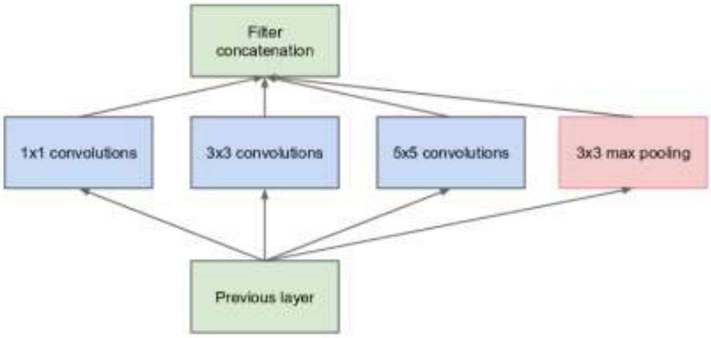


Figure 32: Inception module, naive version[39]

[37] Issame mahdjoubi, Master thesis Automatique UNIVERSITY MOHAMED BOUDIAF - M'SILA

[38] https://deepai.org/machine-learning-glossary-and-terms/inception-module.

[39] Christian Szegedy, "Going deeper with convolutions"

5.3.2 VGGNet

VGGNet is a convolutional neural network architecture that was proposed by A. Zisserman and K. Simonyan of the University of Oxford. The full name of VGG is the Visual Geometry Group, which belongs to the Department of Science and Engineering of Oxford University. VGGNet was developed to increase the depth of CNNs in order to increase the model performance. VGGNet consists of blocks, where each block is composed of 2D Convolution and Max Pooling layers. VGGNet comes in two flavors, VGG16 and VGG19, where 16 and 19 are the number of layers in each of them, respectively. The VGG architecture is very uniform, and the layers are listed as follows: input layer, convolutional layer (64 filters), convolutional layer (64 filters), max-pooling layer, convolutional layer (128 filters), convolutional layer (128 filters), max-pooling layer, convolutional layer (256 filters), convolutional layer (256 filters), convolutional layer (256 filters), max-pooling layer, convolutional layer (512 filters), convolutional layer (512 filters), convolutional layer (512 filters), max-pooling layer, convolutional layer (512 filters), convolutional layer (512 filters), convolutional layer (512 filters), max-pooling layer, fully connected layer (4096 channels), fully connected layer (4096 channels), fully connected layer (1000 channels), output layer. VGGNet is known for its uniform architecture, which makes it easy to understand and implement. VGGNet has fewer convolutional layers than ResNet and GoogleNet, making it work better for a small amount of data. VGGNet-16 model achieves the best result with about 7.0% test error, thereby surpassing a single GoogLeNet by around 0.9%[40].

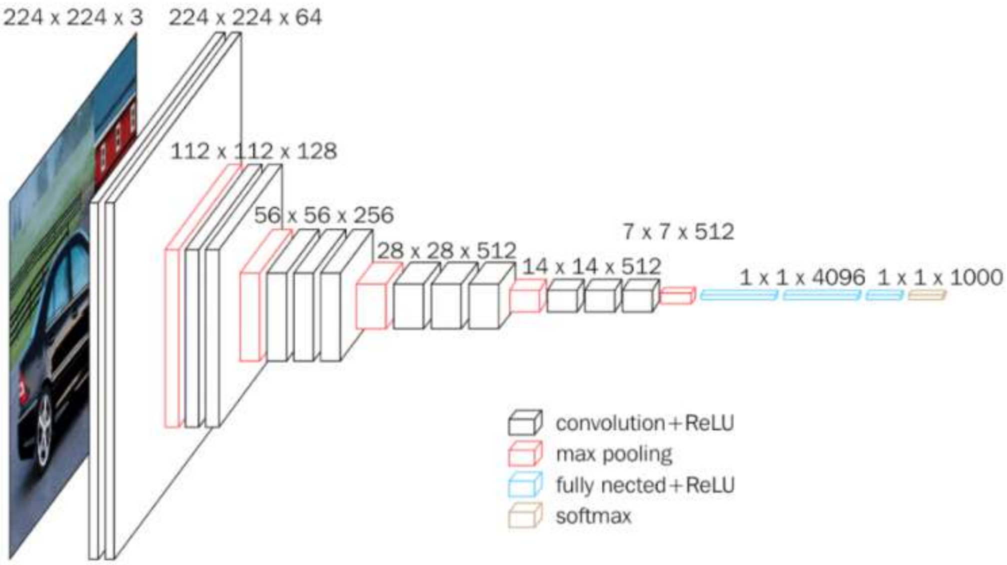


Figure 33: The network structure of VGGNet

5.3.3 ResNet:

ResNet, short for Residual Network, is a type of neural network architecture that was introduced in 2015 by Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. ResNet was designed to overcome the problems of training very deep neural networks, which are networks with many layers. The key idea of ResNet is to add shortcut connections,

[40] https://www.kaggle.com/code/blurredmachine/vggnet-16-architecture-a-complete-guide

or skip connections, between these layers. These skip connections make it possible to train very deep networks without the training process becoming unstable. ResNet is a deep learning algorithm that was developed by Microsoft Research in 2015. It is designed to improve the accuracy of deep neural networks by using a "residual" learning approach.

ResNet network uses a 34-layer plain network architecture inspired by VGG-19 in which then the shortcut connection is added[41].

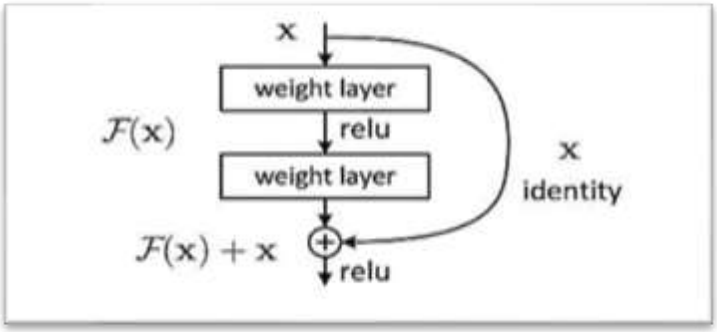


Figure 34: Residual learning : Example[42]

5.4 Transfer Learning

Transfer learning, used in deep learning, is the reuse of a pre-trained model on a new problem. In transfer learning, a machine exploits the knowledge gained from a previous task to improve generalization about another. For example, in training a classifier to predict whether an image contains food, you could use the knowledge it gained during training to recognize drinks.

5.5 Optimization Algorithms

We use optimization Algorithms or methods to change the attributes of the neural network such as weights and learning rate in order to reduce the losses. Optimization algorithms or strategies are responsible for reducing the losses and to provide the most accurate results possible. Understanding the principles of different optimization algorithms is essential for deep learning. Optimization algorithms are particularly important when training a complex deep learning model, which can take hours, days, or even weeks. The performance of the optimization algorithm directly affects the model's training efficiency. Some of the most commonly used optimization algorithms include Gradient Descent, Stochastic Gradient Descent (SGD), AdaGrad, RMSProp, and Adam Optimizer.

- **Gradient Descent:**

This is the most basic and most used optimization algorithm. It is used heavily in linear regression and classification algorithms and also in backpropagation in neural networks. The algorithm works by iteratively adjusting the weights of the neural network to minimize the loss function.

[41] https://www.mygreatlearning.com/blog/resnet/
[42] 'Deep Residual Learning for Image Recognition' Kaiming He et al

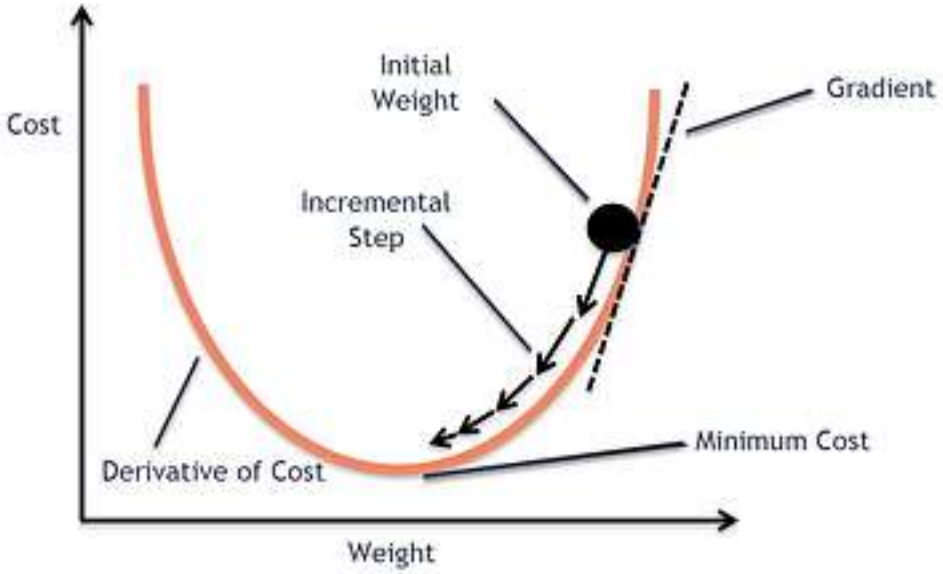


Figure 35:Gradient Descent cost

- **Stochastic Gradient Descent (SGD):**

This algorithm is a variation of Gradient Descent and is commonly used in deep learning. It works by randomly selecting a subset of the training data (a batch) and updating the weights of the neural network based on the error calculated on that batch.

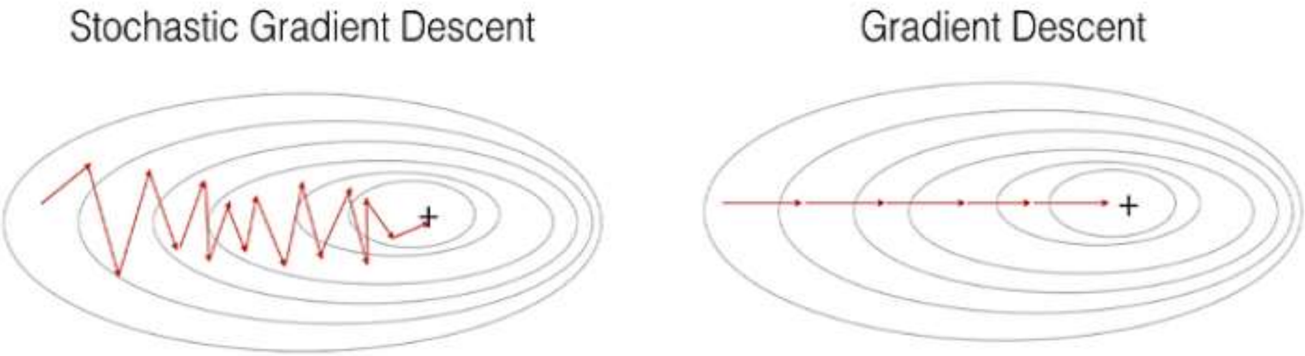


Figure 36:SGD VS GD

- **AdaGrad:**

This algorithm adapts the learning rate of each parameter based on the historical gradient information. It is particularly useful for sparse data.

- **RMSProp:**

This algorithm is a modification of AdaGrad that uses a moving average of the squared gradient to adjust the learning rate.

- **Adam Optimizer:**

This algorithm combines the benefits of both AdaGrad and RMSProp. It uses a moving average of the gradient and the squared gradient to adapt the learning rate.

These optimization algorithms are used to improve the performance of deep learning models by reducing the loss function and providing the most accurate results possible[43].

6 Conclusion

In conclusion, Artificial Intelligence (AI) is indeed a tool that aims to mimic human intelligence and perform tasks more efficiently and quickly than humans. However, the purpose of AI is not necessarily to replace humans entirely but rather to complement and enhance human abilities.

[43] https://keras.io/api/optimizers/adam/?fbclid=IwAR2zeWVUV3UiJ4iiSwLZYNYYb7rM-lVb8SWapQKcyA9BHvqwxTC4iNQ-O2Y

Chapter III
Related work
(state of art)

1 introduction

In the field of image recognition, researchers are actively engaged in exploring and advancing various aspects of the technology. Here is a general introduction to some of the current research trends and areas of focus in image recognition:

1. Deep Learning Architectures: Deep learning has revolutionized image recognition, and researchers continue to explore new architectures and model designs. This includes developing more efficient convolutional neural networks (CNNs) with improved performance, such as residual networks (ResNets), inception networks, and attention mechanisms.
   The method that we are using in this humble work.
2. Transfer Learning and Pretrained Models: Transfer learning involves leveraging pre-trained models on large-scale datasets to accelerate training and improve performance on specific image recognition tasks. Researchers are investigating methods to enhance transfer learning techniques and adapt them to different domains and tasks.
   The paper by Oquab et al. explores the use of transfer learning and pretrained convolutional neural networks (CNNs) for learning mid-level image representations[44].
3. Explain ability and Interpretability: Understanding the reasoning behind image recognition decisions is gaining importance. Researchers are exploring techniques to make deep learning models more interpretable, providing explanations and insights into why certain predictions are made. This includes methods like attention mechanisms, saliency maps, and concept activation vectors.
   The paper by Selvaraju et al. introduces the concept of Grad-CAM (Gradient-weighted Class Activation Mapping), a technique for generating visual explanations from deep neural networks. Grad-CAM highlights the important regions in an image that contribute most significantly to the network's classification decision[45].
4. Adversarial Attacks and Defenses: Adversarial attacks aim to manipulate images in imperceptible ways to deceive image recognition systems. Researchers are developing robust defenses against such attacks, including adversarial training, defensive distillation, and feature squeezing, to enhance the robustness and security of image recognition models.
   The paper by Goodfellow et al introduces the concept of adversarial perturbations[46].
5. Few-Shot and Zero-Shot Learning: Traditional image recognition models often require large labeled datasets for training. However, researchers are focusing on developing techniques that enable learning from limited or even zero labeled samples. Few-shot learning aims to recognize objects with limited labeled examples, while zero-shot learning extends the recognition to unseen classes based on semantic relationships.
6. Multimodal Learning: Combining visual information with other modalities like text or audio is an emerging research direction. Researchers are exploring techniques to fuse multiple modalities to improve image recognition performance, especially in tasks such as image captioning, visual question answering (VQA), and cross-modal retrieval.

[44] Oquab, M., Bottou, L., Laptev, I., & Sivic, J. (2014). Learning and transferring mid-level image representations using convolutional neural networks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, 1717-1724

[45] Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2017). Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization. Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017, 618-626

[46] Goodfellow, I. J., Shlens, J., & Szegedy, C. (2015). Explaining and Harnessing Adversarial Examples. Proceedings of the International Conference on Learning Representations (ICLR), 2015

The paper by Frome et al. introduces the Devise model, which focuses on learning joint representations of visual and semantic information for image recognition tasks[47].

7. Robustness to Domain Shift and Variations: Image recognition models often struggle when applied to different domains or face variations in image quality, lighting conditions, or viewpoints. Researchers are developing methods to improve the robustness of models to handle domain shift, dataset bias, and generalization to unseen variations.

These are just a few examples of the exciting research directions in image recognition. The field is vast, and researchers are continually exploring innovative approaches to improve accuracy, efficiency, interpretability, and robustness, as well as exploring new application domains where image recognition can make a significant impact.

2 Hokuto Kagaya et al Article

They apply a convolutional neural network (CNN) to the tasks of detecting and recognizing food images. Because of the wide diversity of types of food, image recognition of food items is generally very difficult. However, deep learning has been shown recently to be a very powerful image recognition technique, and CNN is a **state-of-the-art** approach to deep learning.

They applied CNN to the tasks of food detection and recognition through parameter optimization.

They constructed a dataset of the most frequent food items in a publicly available food-logging system, and used it to evaluate recognition performance. CNN showed significantly higher accuracy than did traditional support-vector machine-based methods with handcrafted features. In addition, they found that the convolution kernels show that color dominates the feature extraction process. For food image detection, CNN also showed significantly higher accuracy than a conventional method did.

They scaled the dataset images to 80 × 80. When training and testing, cuda-convnet’s python module randomly cropped them to 64 × 64.

they divided the dataset into six sets: four sets were used for training, one for validation, and one for testing.

they conducted a 6-fold cross validation.

For a comparison between CNN and existing techniques, they prepared three methods using SVM-based recognition using handcrafted features: spatial pyramid matching (SPM) using a color histogram + SVM; GIST features + SVM;

The ScSPM package, which is a SIFT–BoW-based method[48].

Table shows the experimental results. The top line shows the average of the 10 trials with the same dataset and hyper parameters. The existing techniques used as baseline methods had about 50–60% accuracy, whereas the CNNs achieved more than 70%. The two-layer CNN performed better than did the three-layer and four-layer CNNs. With respect to kernel size, 5 × 5 was the best.

[47] Frome, A., Corrado, G. S., Shlens, J., Bengio, S., Dean, J., Ranzato, M., & Mikolov, T. (2013). Devise: A Deep Visual-Semantic Embedding Model. Proceedings of the Neural Information Processing Systems (NeurIPS), 2013

[48] Hokuto Kagaya, Kiyoharu Aizawa, Makoto Ogawa, “Food Detection and Recognition Using Convolutional Neural Network”

Table 1:Accuracy of CNNs for various hyper-parameter values compared with three existing techniques.

Layers	Number of kernels	Size of kernels	Normalization (LRN)	Dataset	Avg.	SD
2	32-32	9-7	1 time	Same dataset and parameters	72.39%	1.48%
*2	32-32	9x9, 7x7	1 time	6-fold cross validation	71.17%	0.83%
2	32-64	9x9, 7x7	1 time	6-fold cross validation	72.94%	0.68%
2	64-64	9x9, 7x7	1 time	6-fold cross validation	70.07%	2.98%
3	32-32-64	5x5, 5x5, 5x5	2 times	6-fold cross validation	69.82%	0.73%
4	32-32-32-64	7x7, 7x7, 5x5, 2x2	3 times	6-fold cross validation	66.57%	0.91%
2	32-32	7x7, 5x5	1 time	6-fold cross validation	72.86%	1.35%
2	32-32	5x5, 3x33	1 time	6-fold cross validation	72.88%	1.68%
2	32-32	5x5, 5x5	1 time	6-fold cross validation	73.69%	1.28%
2	32-32	5x5, 5x5	1 time	6-fold cross validation	**73.70%**	0.81%
SPM + Color + SVM				6-fold cross validation	54.63%	N.A.
GIST + SVM				6-fold cross validation	52.53%	N.A.
ScSPM				6-fold cross validation	60.47%	N.A.

3 Pishva and al article

Proposed a bread recognition system which can deal with 73 kinds of bread artisanal with a classification rate of 95% the images of their dataset are taken by a special setting of the fixed camera in order to let the center of the bread adapt to the center of an image, and they used a uniform background to separate the bread regions from backgrounds easily. On the other hand, we deal with images of food taken by many people in various contexts. In fact, in the experiment, we use food images collected from the web[49].

4 Taichi Joutou & Keiji Yanai article

As food image recognition, D. Pishva et al. proposed a bread recognition system which can treat with 73 kinds of hand-made bread with the 95% classification rate. However,

images in their dataset are taken by a special fixed camera setting in order to let the center of bread fit to the center of an image, and they used uniform background to separate bread regions from backgrounds easily. On the other hand, **Taichi Joutou & Keiji Yanai** treat with food images taken by many people in various settings. In fact, in the experiment, we use food images gathered from the Web.

In this paper, they realize image recognition of many kinds of foods with high accuracy by introducing a MKL-based feature fusion method into food image recognition. In our recognition, they prepared 50 kinds of food categories as shown in Figure, and classify an unknown food image into one of the categories. There has been no

[49] D. Pishva, A. Kawai, K. Hirakawa, K. Yamamori, and T. Shiino,“Bread Recognition Using Color Distribution Analysis,”

systems which can handle such many kinds of food categories so far.

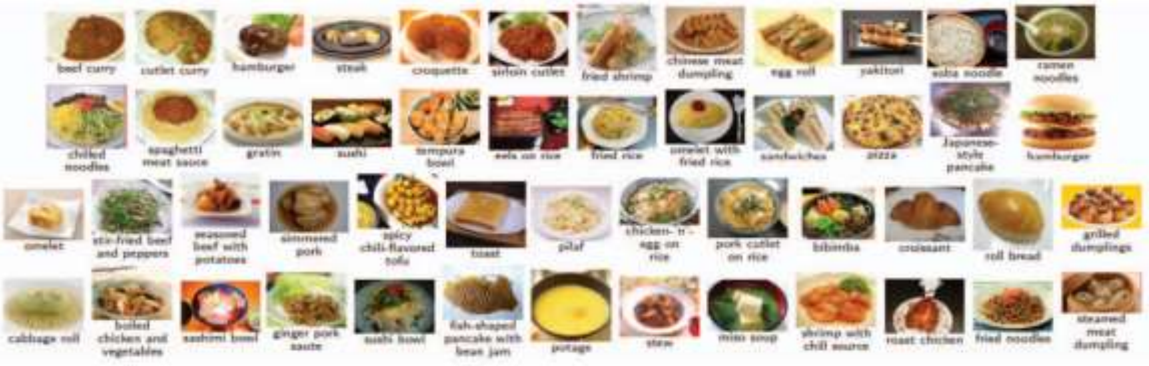


Figure 37:50 kinds of food images which are recognition targets in the paper

In the training step, they extract various kinds of image features such as bag-of-features (BoF), color histogram and Gabor texture features from the training images, and they train a MKL-SVM with extracted features. In the classification step, they extract image features from a given image in the same way as the training step, and classify it into one of the given food categories with the trained MKL SVM[50].

general comparison

Table 2:general comparison

	methode	dataset	Classification rate	Type of Dataset
Hokuto Kagaya et al	CNN	publicly available food-logging system	73.70%	various types of food
Pishva and al	MKL	bread images using a fixed camera	95%	73 kinds of artisanal bread
Taichi Joutou & Keiji Yanai	MKL-SVM	food images gathered from the web	80.05%	various types of food

5 Conclusion

In conclusion, research in the field of image recognition, particularly in the context of food images, aims to improve the accuracy and reliability of models for food recognition tasks. Techniques such as deep learning, transfer learning are being explored to enhance the performance of food image recognition systems. Researchers are also investigating methods for recognizing food items with limited labeled data, enabling more effective food classification and identification. By leveraging multiple modalities, such as visual and textual information, researchers are striving to develop robust and interpretable models for accurate food image recognition. These advancements have the potential to benefit various applications related to food analysis, dietary monitoring, and recipe recommendation systems.

[50] Taichi Joutou and Keiji Yanai "A FOOD IMAGE RECOGNITION SYSTEM WITH MULTIPLE KERNEL LEARNING"

Chapter IV
Implementation and Experiences

1 Introduction

In this chapter, we will highlight our design and show how to create our own classifier and create our model. Then we will evaluate and create 03 models with different architectures and then we will apply these models to the dishes downloaded from Kaggle. For this we will work with the python programming language and libraries like Tensorflow and Keras for training and compilation, and to improve the performance of the models we will use simple but effective techniques like Dropout.

2 Software and libraries Used in the implementation

2.1 Python

Python is an easy to learn, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming. Python’s elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms [51].

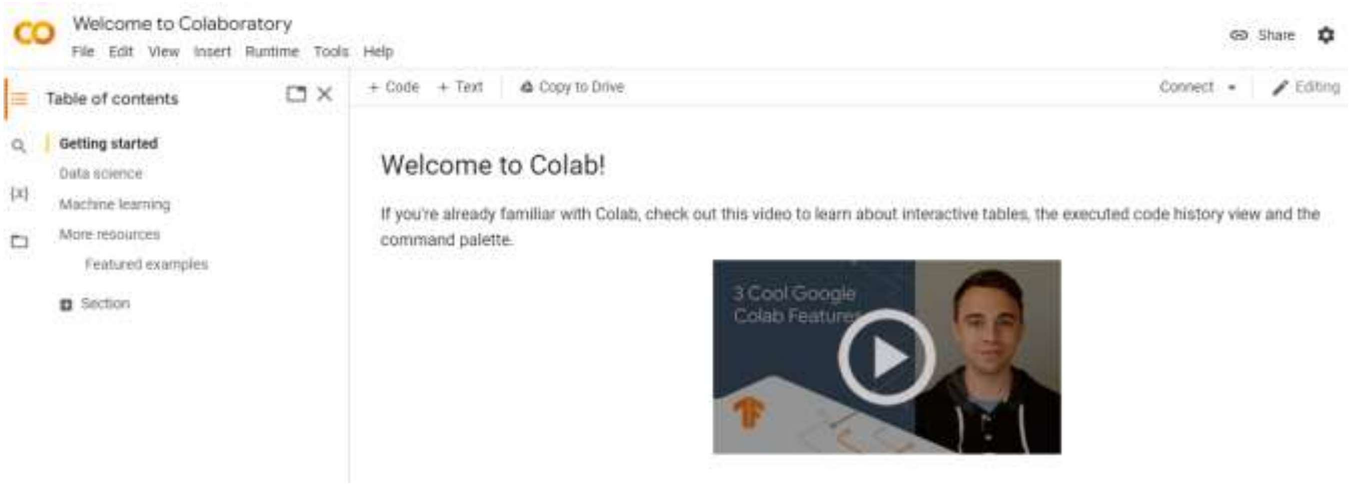
Figure 38:The Python Logo

2.2 Google colab

Collaboratory, or “Colab” for short, is a product from Google Research. Colab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education. More technically, Colab is a hosted Jupyter notebook service that requires no setup to use, while providing access free of charge to computing resources including GPUs [52].

[51] https://docs.python.org/fr/3/tutorial/

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



Welcome to Colab!

If you're already familiar with Colab, check out this video to learn about interactive tables, the executed code history view and the command palette.

Figure 39:Google colab

You can write Python codes using Colab on your Google Chrome or Mozilla Firefox web browsers. You can also execute those codes on the browser without needing any runtime environment or command line interface.

Furthermore, you can give your Python project notebook a professional look by adding mathematical equations, graphs, tables, images, and other graphics. Additionally, you can code data visualizations in Python, and Colab will render the code in a visual asset [53].

2.3 Navigator Anaconda

Conda is an open-source package management system and environment management system that runs on Windows, macOS, and Linux. Conda quickly installs, runs and updates packages and their dependencies. Conda easily creates, saves, loads and switches between environments on your local computer. It was created for Python programs, but it can package and distribute software for any language.

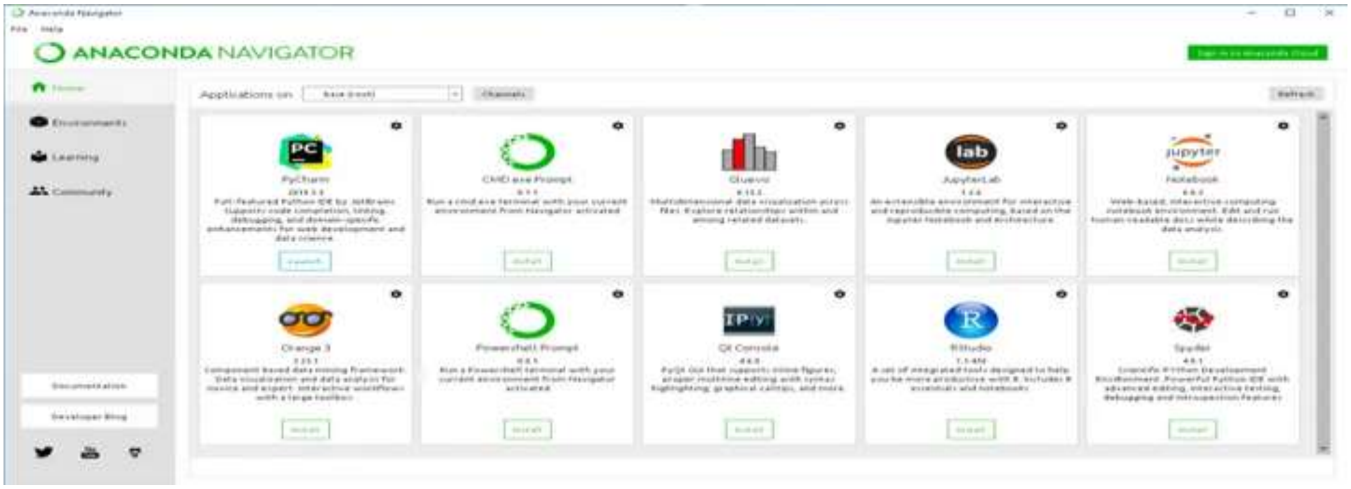


Figure 40:Navigateur Anaconda

[53] https://geekflare.com/google-colab/

Conda as a package manager helps you find and install packages. If you need a package that requires a different version of Python, you do not need to switch to a different environment manager, because conda is also an environment manager. With just a few commands, you can set up a totally separate environment to run that different version of Python, while continuing to run your usual version of Python in your normal environment [54].

2.4 Visual Studio Code

Visual Studio Code is a lightweight but powerful source code editor which runs on your desktop and is available for Windows, macOS and Linux. It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages and runtimes (such as C++, C#, Java, Python, PHP, Go, .NET)[55].

Figure 41:Visual Studio Code

2.5 Keras

Keras is a deep learning API written in Python, running on top of the machine learning platform TensorFlow. It was developed with a focus on enabling fast experimentation. Being able to go from idea to result as fast as possible is key to doing good research [56].

Figure 42:Logo Keras

2.6 TensorFlow

Tensorflow is an open-source library for numerical computation and large-scale machine learning that ease Google Brain TensorFlow, acquiring data, training models, serving predictions, and refining future results.

[54] https://docs.conda.io/en/latest/
[55] https://code.visualstudio.com/docs
[56] https://keras.io/about/

Tensorflow bundles together Machine Learning and Deep Learning models and algorithms. It uses Python as a convenient front-end and runs it efficiently in optimized C++.

Figure 43: Logo Tensorflow

Tensorflow allows developers to create a graph of computations to perform. Each node in the graph represents a mathematical operation, and each connection represents data. Hence, instead of dealing with low details like figuring out proper ways to hitch the output of one function to the input of another, the developer can focus on the overall logic of the application[57].

2.7 Scikit-learn

Scikit-learn is an open-source machine learning library that supports supervised and unsupervised learning. It also provides various tools for model fitting, data preprocessing, model selection, model evaluation, and many other utilities[58].

Figure 44: Logo Scikit-learn

2.8 OpenCV

OpenCV (Open-Source Computer Vision Library) is an open-source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to

[57] https://www.mygreatlearning.com/blog/what-is-tensorflow-machine-learning-library-explained/

[58] https://scikit-learn.org/stable/getting_started.html

accelerate the use of machine perception in the commercial products. Being an Apache 2 licensed product, OpenCV makes it easy for businesses to utilize and modify the code [59].

Figure 45:Logo OpenCV

2.9 NumPy

NumPy is a Python library used for working with arrays.

It also has functions for working in domain of linear algebra, Fourier transform, and matrices.

NumPy was created in 2005 by Travis Oliphant. It is an open source project and you can use it freely.

NumPy stands for Numerical Python [60].

Figure 46:Logo NumPy

2.10 Flask

Flask is an API of Python that allows us to build up web-applications. It was developed by Armin Ronacher. Flask's framework is more explicit than Django's framework and is also easier to learn because it has less base code to implement a simple web-Application. A Web-Application Framework or Web Framework is the collection of modules and libraries that helps the developer to write applications without writing the low-level codes such as

[59] https://opencv.org/about/
[60] https://www.w3schools.com/python/numpy/numpy_intro.asp

protocols, thread management, etc. Flask is based on WSGI (Web Server Gateway Interface) toolkit and Jinja2 template engine [61].

Figure 47:Logo Flask

3 database used

3.1 Food Images (Food-101)

The dataset contains a number of different subsets of the full food-101 data. The idea is to make a more exciting simple training set for image analysis than CIFAR10 or MNIST. For this reason, the data includes massively downscaled versions of the images to enable quick tests. The data has been reformatted as HDF5 and specifically Keras HDF5Matrix which allows them to be easily read in. The file names indicate the contents of the file 62.

Figure 48:kaggle datasets download -d kmader/food41

3.2 Our database

We created a small dataset where we took a previously studied database containing ("baghrir", "baklawa", "couscous", "mhajeb", "pizza" and "spaghetti") and added 3 categories of

Food_101 database ("borak", "cigar", and "fritte"). So, we have a database of nine chapters, each chapter containing 400 images.

[61] https://www.geeksforgeeks.org/python-introduction-to-web-development-using-flask/
[62] https://www.kaggle.com/datasets/kmader/food41?resource=download

Each dish type contains 300 practice and 100 dish for the exam.

Note: The training images have not been cleaned up and therefore still contain a certain amount of noise.

4 Architecture of our network

During our experiments, we created three models (model 1, model 2 and model 3) with different architectures.

In the following we present the architecture of the three models.

4.1 Model Vgg16

The full name of VGG is the **Visual Geometry Group**, which belongs to the Department of Science and Engineering of Oxford University. It has released a series of convolutional network models beginning with VGG, which can be applied to face recognition and image classification, from VGG16 to VGG19. The original purpose of VGG's research on the depth of convolutional networks is to understand how the depth of convolutional networks affects the accuracy and accuracy of large-scale image classification and recognition. -Deep-16 CNN), in order to deepen the number of network layers and to avoid too many parameters, a small 3x3 convolution kernel is used in all layers [63].

4.1.1 VGG16 Architecture

VGG16, as its name suggests, is a 16-layer deep neural network. VGG16 is thus a relatively extensive network with a total of 138 million parameters—it's huge even by today's standards. However, the simplicity of the VGGNet16 architecture is its main attraction.

The VGGNet architecture incorporates the most important convolution neural network features [64] .

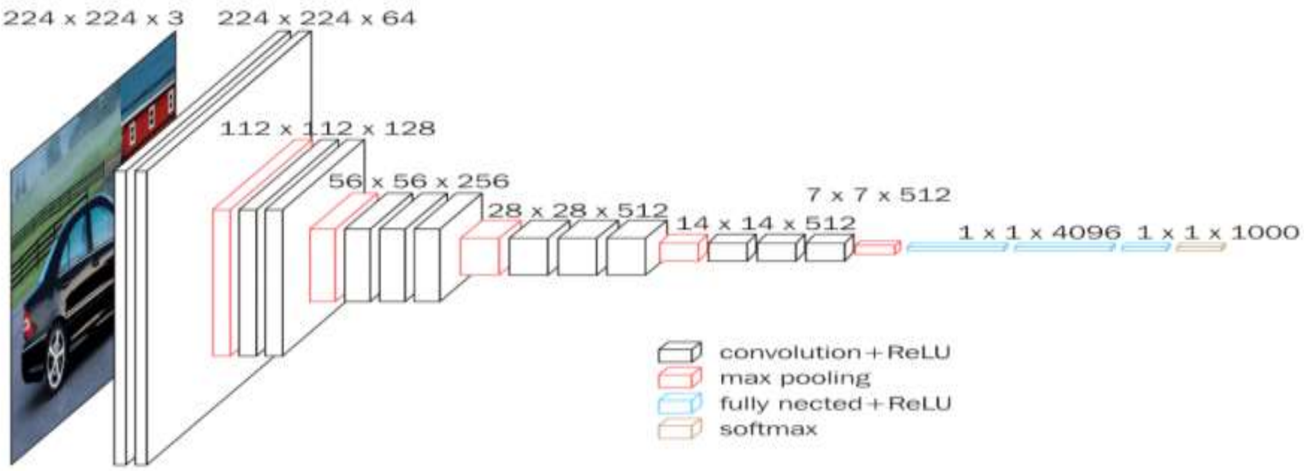


Figure 49:VGG16 Architecture

[63] https://www.kaggle.com/code/blurredmachine/vggnet-16-architecture-a-complete-guide
[64] https://datagen.tech/guides/computer-vision/vgg16/

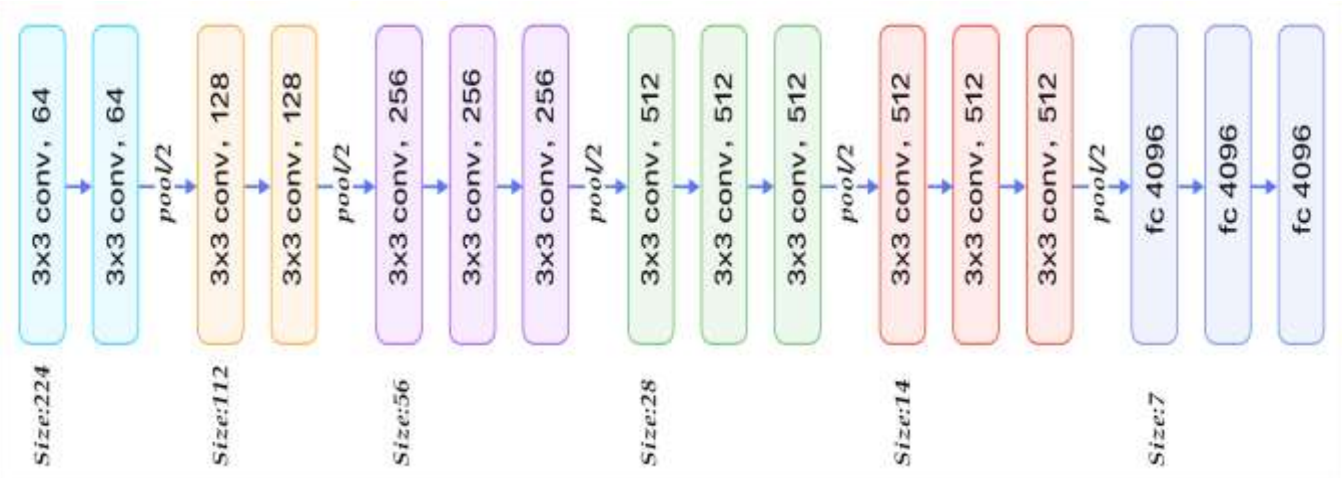
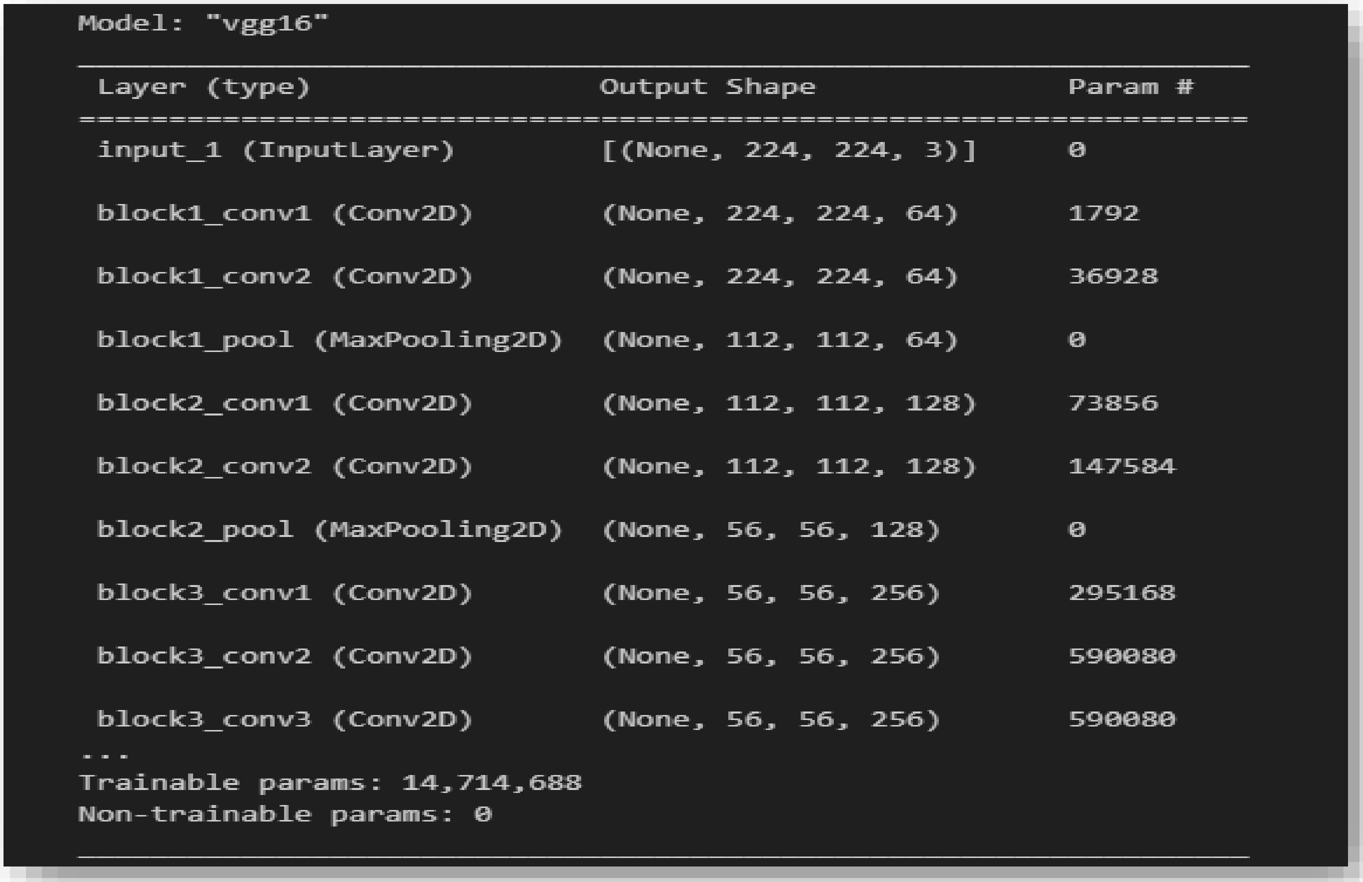


Figure 50:Network Vgg16

This is the configuration VGG16:



```
Model: "vgg16"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_1 (InputLayer)        [(None, 224, 224, 3)]     0

 block1_conv1 (Conv2D)       (None, 224, 224, 64)      1792

 block1_conv2 (Conv2D)       (None, 224, 224, 64)      36928

 block1_pool (MaxPooling2D)  (None, 112, 112, 64)      0

 block2_conv1 (Conv2D)       (None, 112, 112, 128)     73856

 block2_conv2 (Conv2D)       (None, 112, 112, 128)     147584

 block2_pool (MaxPooling2D)  (None, 56, 56, 128)       0

 block3_conv1 (Conv2D)       (None, 56, 56, 256)       295168

 block3_conv2 (Conv2D)       (None, 56, 56, 256)       590080

 block3_conv3 (Conv2D)       (None, 56, 56, 256)       590080
...
Trainable params: 14,714,688
Non-trainable params: 0
_________________________________________________________________
```

Figure 51:Network Configuration_VGG16

4.2 Model inception v3

Inception v3 mainly focuses on burning less computational power by modifying the previous Inception architectures. This idea was proposed in the paper Rethinking the Inception Architecture for Computer Vision, published in 2015. It was co-authored by Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, and Jonathon Shlens.

4.2.1 Inception V3 architecture

The architecture of an Inception v3 network is progressively built, step-by-step, as explained below:

1. Factorized Convolutions: this helps to reduce the computational efficiency as it reduces the number of parameters involved in a network. It also keeps a check on the network efficiency.

2. Smaller convolutions: replacing bigger convolutions with smaller convolutions definitely leads to faster training. Say a 5 × 5 filter has 25 parameters; two 3 × 3 filters replacing a 5 × 5 convolution has only 18 (3*3 + 3*3) parameters instead.

3. Asymmetric convolutions: A 3 × 3 convolution could be replaced by a 1 × 3 convolution followed by a 3 × 1 convolution. If a 3 × 3 convolution is replaced by a 2 × 2 convolution, the number of parameters would be slightly higher than the asymmetric convolution proposed.

4. Auxiliary classifier: an auxiliary classifier is a small CNN inserted between layers during training, and the loss incurred is added to the main network loss. In Google Net auxiliary classifiers were used for a deeper network, whereas in Inception v3 an auxiliary classifier acts as a regularized.

5.Grid size reduction: Grid size reduction is usually done by pooling operations. However, to combat the bottlenecks of computational cost.

All the above concepts are consolidated into the final architecture [65]:

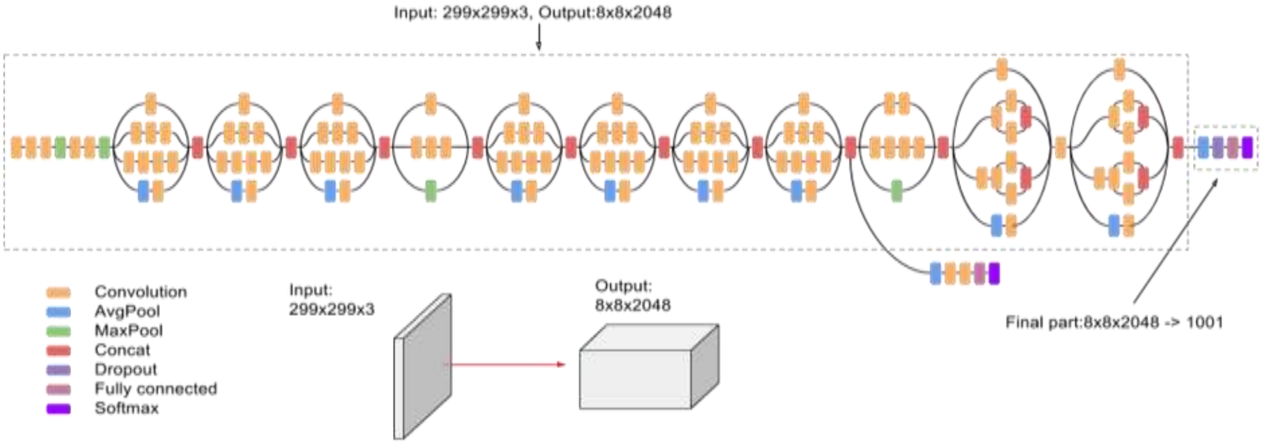
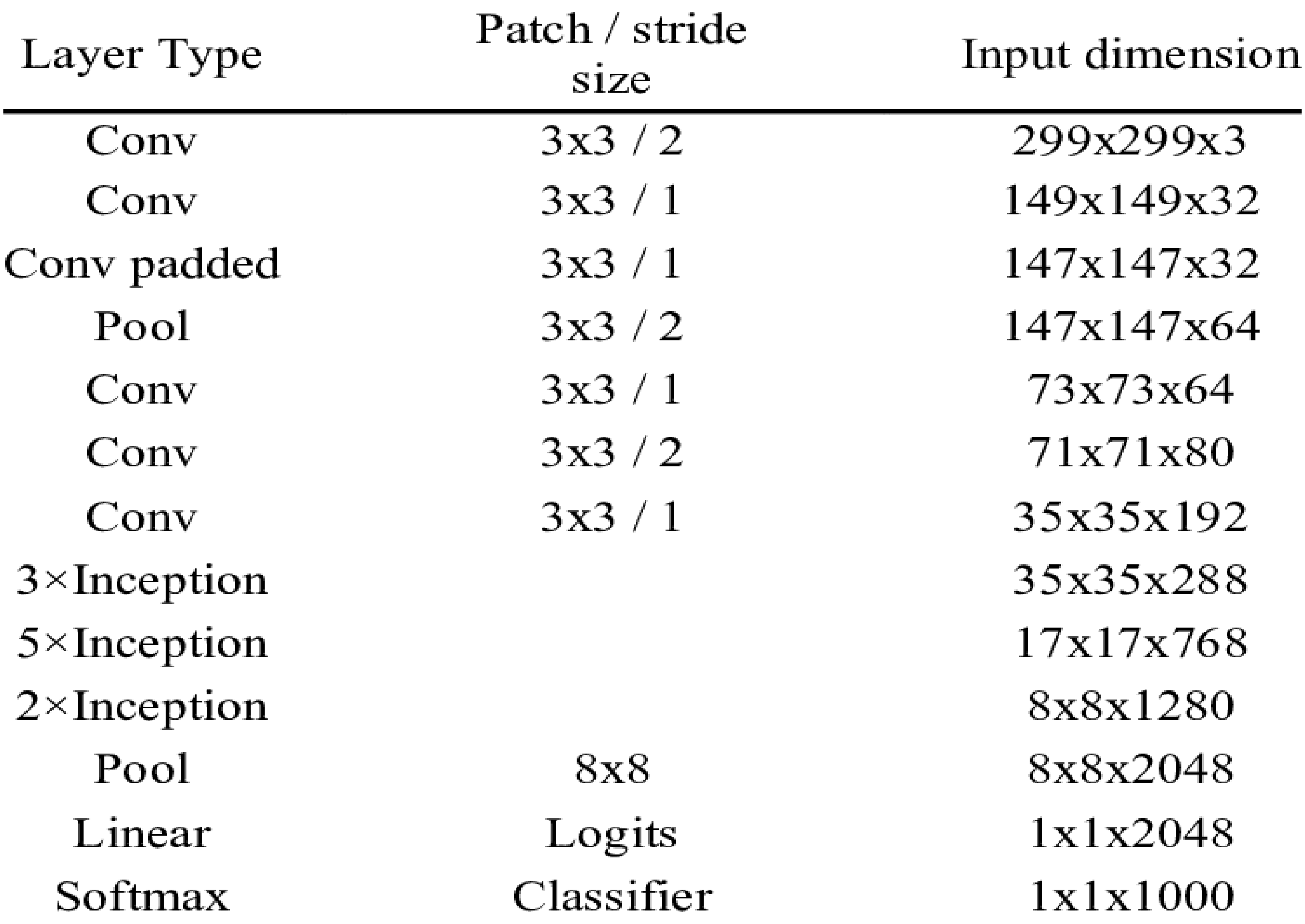


Figure 52:inception V3

[65] https://blog.paperspace.com/popular-deep-learning-architectures-resnet-inceptionv3-squeezenet/

This is the configuration inception v3:



Layer Type	Patch / stride size	Input dimension
Conv	3x3 / 2	299x299x3
Conv	3x3 / 1	149x149x32
Conv padded	3x3 / 1	147x147x32
Pool	3x3 / 2	147x147x64
Conv	3x3 / 1	73x73x64
Conv	3x3 / 2	71x71x80
Conv	3x3 / 1	35x35x192
3×Inception		35x35x288
5×Inception		17x17x768
2×Inception		8x8x1280
Pool	8x8	8x8x2048
Linear	Logits	1x1x2048
Softmax	Classifier	1x1x1000

Figure 53:confuration inception v3

4.3 ResNet

ResNet (short for "Residual Network") is a deep neural network architecture that was introduced by Microsoft researchers in 2015. ResNet is designed to address the problem of vanishing gradients in very deep neural networks, which can make it difficult to train them effectively [66] .

4.3.1 ResNet Architecture

The key innovation of ResNet is the use of residual connections, which allow information to be passed directly from one layer to another, without being transformed by intermediate layers. This facilitates the training of very deep networks by allowing the gradients to flow more easily through the network, since the residual connections provide a shortcut for the gradients to bypass the intermediate layers .

[66] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR) (pp. 770-778). IEEE.

The approach behind this network is instead of layers learning the underlying mapping, we allow the network to fit the residual mapping. So, instead of say H(x), initial mapping, let the network fit:

$F(x) := H(x) - x$ which gives $H(x) := F(x) + x$.

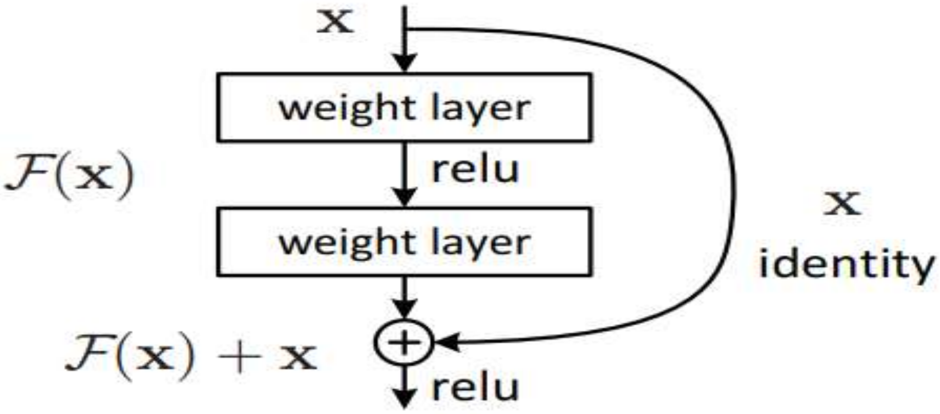

x
weight layer
$\mathcal{F}(\mathbf{x})$
relu
x
identity
weight layer
$\mathcal{F}(\mathbf{x}) + \mathbf{x}$
relu

Figure 54:Residual-Block

The basic building block of a ResNet is called a "residual block", which consists of two or more convolutional layers with a shortcut connection that bypasses the convolutional layers. The output of the residual block is the sum of the input and the output of the convolutional layers. This allows the network to learn residual functions, which capture the difference between the input and the desired output.

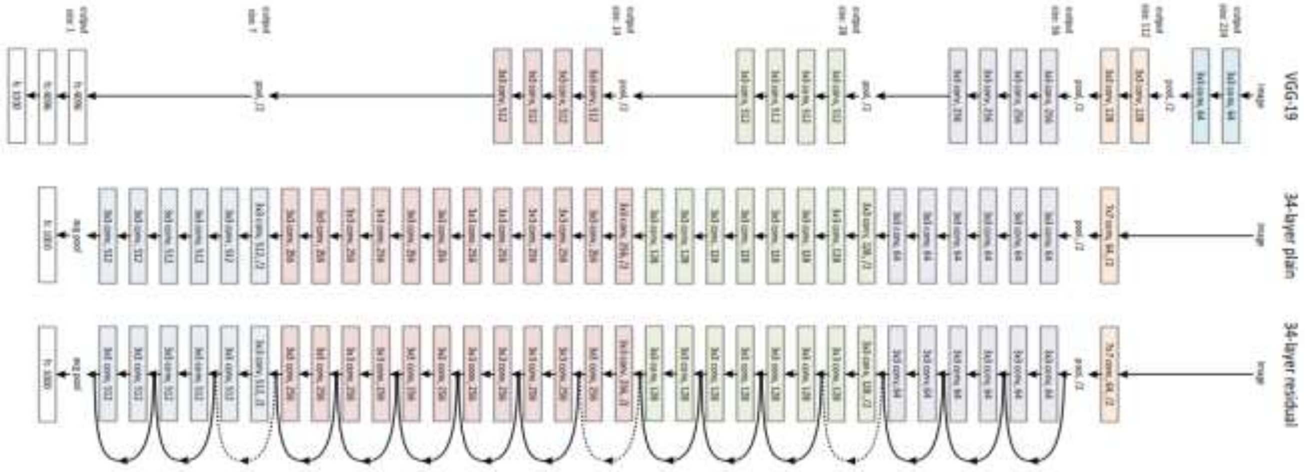
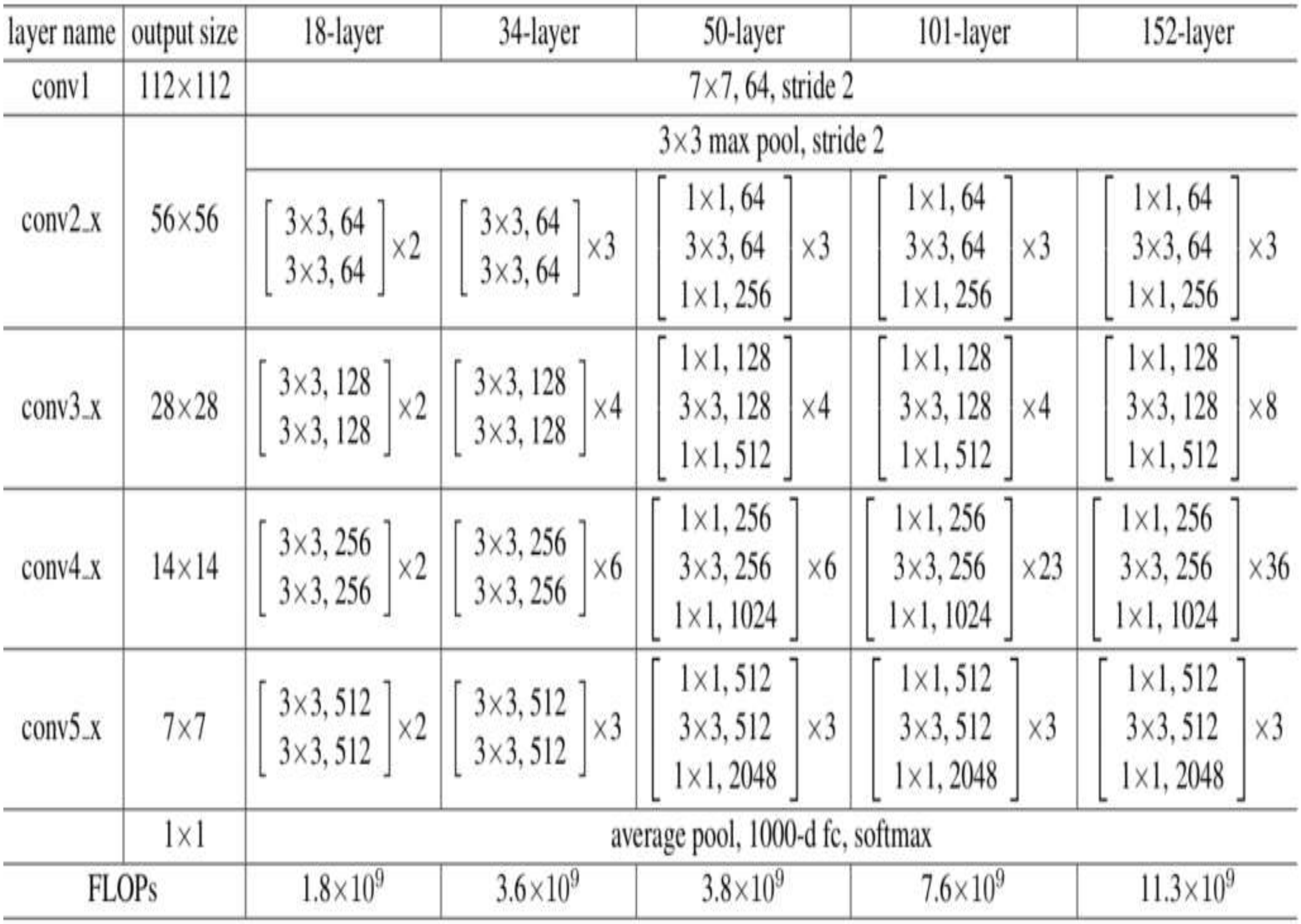


Figure 55:ResNet -34 architecture

ResNet comes in several variants, including ResNet-18, ResNet-34, ResNet-50, ResNet-101, and ResNet-152, which differ in the number of layers and the number of residual blocks in each layer. ResNet-50, for example, has 50 layers and uses a bottleneck architecture to reduce the number of parameters, while ResNet-152 is a very deep network with 152 layers[67].

[67] https://www.geeksforgeeks.org/residual-networks-resnet-deep-learning/

And This is the configuration ResNet:



layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		[3×3, 64 3×3, 64] ×2	[3×3, 64 3×3, 64] ×3	[1×1, 64 3×3, 64 1×1, 256] ×3	[1×1, 64 3×3, 64 1×1, 256] ×3	[1×1, 64 3×3, 64 1×1, 256] ×3
conv3_x	28×28	[3×3, 128 3×3, 128] ×2	[3×3, 128 3×3, 128] ×4	[1×1, 128 3×3, 128 1×1, 512] ×4	[1×1, 128 3×3, 128 1×1, 512] ×4	[1×1, 128 3×3, 128 1×1, 512] ×8
conv4_x	14×14	[3×3, 256 3×3, 256] ×2	[3×3, 256 3×3, 256] ×6	[1×1, 256 3×3, 256 1×1, 1024] ×6	[1×1, 256 3×3, 256 1×1, 1024] ×23	[1×1, 256 3×3, 256 1×1, 1024] ×36
conv5_x	7×7	[3×3, 512 3×3, 512] ×2	[3×3, 512 3×3, 512] ×3	[1×1, 512 3×3, 512 1×1, 2048] ×3	[1×1, 512 3×3, 512 1×1, 2048] ×3	[1×1, 512 3×3, 512 1×1, 2048] ×3
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

Figure 56:configuration ResNet

5 Results obtained and discussion

In order to show the results obtained for the three models, we explain in the following the results in terms of:

Accuracy, error, and confusion matrix for each of the three models.

5.1 Results obtained for Vgg16

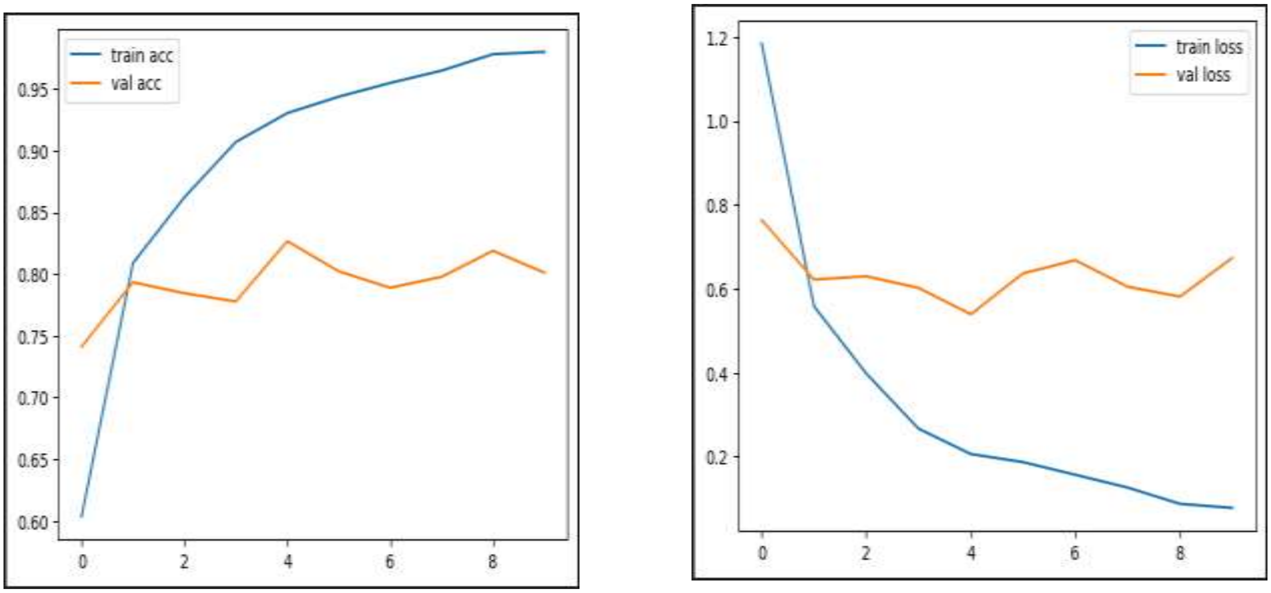


Figure 57: Accuracy and loss of Vgg16

After analyzing the obtained results, we make the following observations:

According to the Figure 57, the accuracy of learning and testing increases with the number of epochs,

This reflects that with each epoch, the model learns more information. If the accuracy decreases

We will then need more information to train our model, and so should we

Increase the number of epochs and vice versa.

Likewise, the learning and validation error decreases with the number of epochs.

We got an error rate of 7.79% on train and 67.35% on check,

The accuracy rate is 98.00% on train and 80.11% on validation.

5.2 Results obtained for inception V3

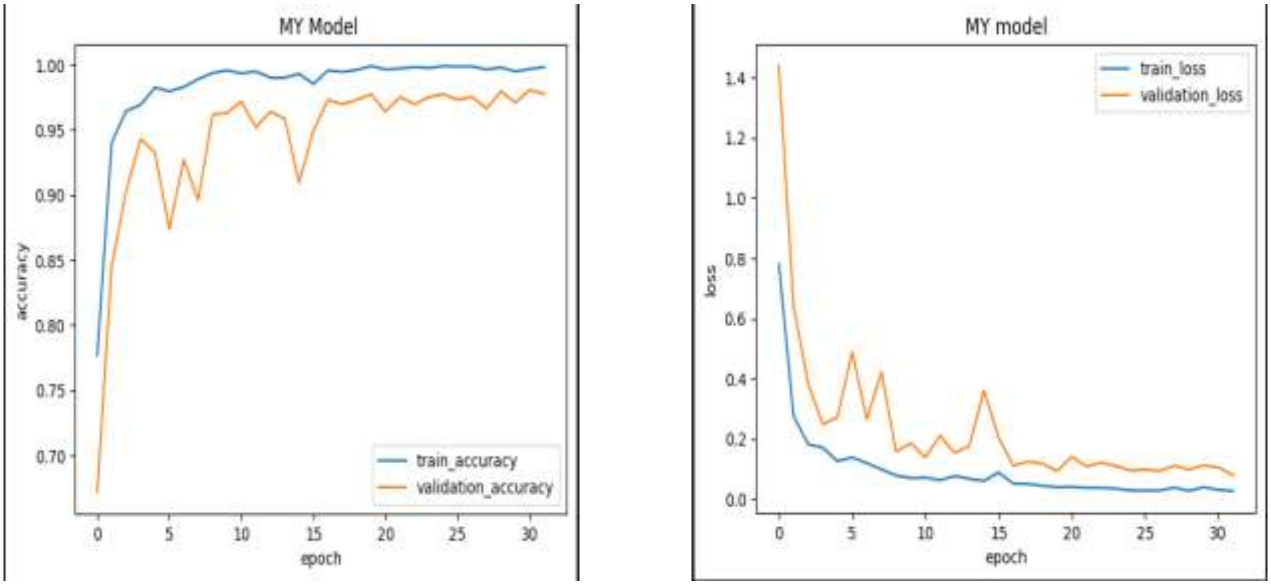


Figure 58:Accuracy and loss of inception V3

After analyzing the results obtained, we make the following observations:

From Figure 58 the accuracy of training and validation increases with the increase in the number

This reflects that with each epoch, the model learns more information. If the accuracy

Next, we will need more information to make our model learn so

We must increase the number of epochs and vice versa.

Likewise, the learning and validation error decreases with the number of epochs

We got an error rate of 48,89% on train and 48,89% on validation,

The accuracy rate is 99,85% in train and 97,77% in validation.

5.3 Results obtained for ResNet

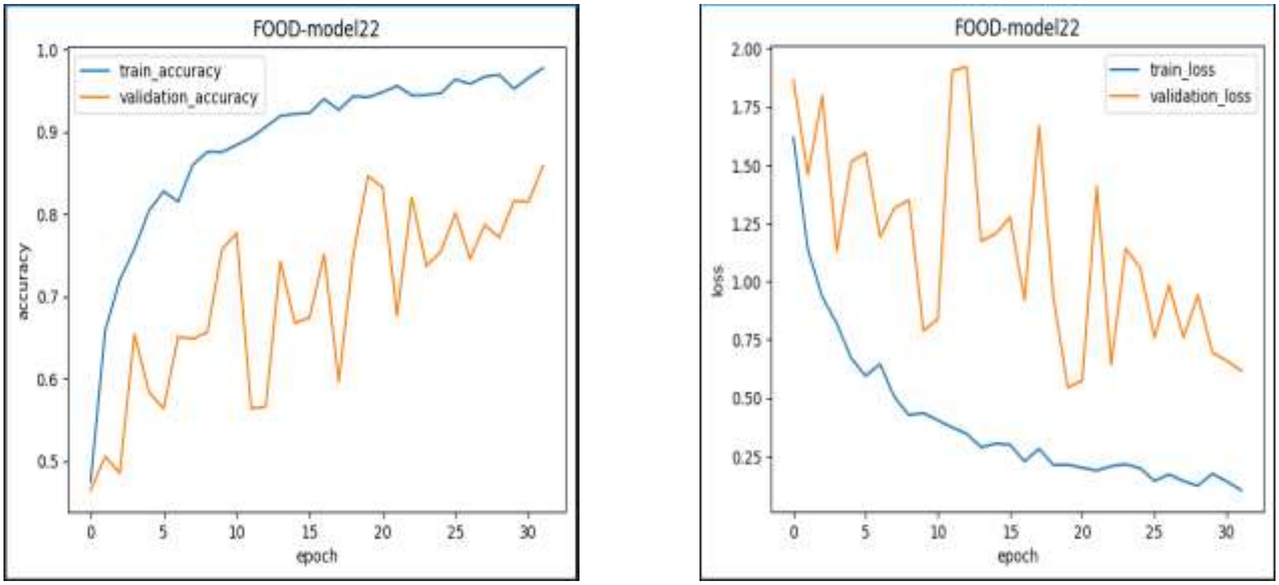


Figure 59:Accuracy and loss of ResNet

After analyzing the results obtained, we make the following observations:

From Figure 59 the accuracy of learning and validation increases with the number of epochs

this reflects that in each epoch the model learns more information. If the accuracy is reduced, we will need more information to make our model learn, and therefore we must increase the number of epochs and vice versa.

Likewise, the learning and validation error decreases with the number of epochs.

We got an error rate of 10,34% on train and 61,70% on validation,

The accuracy rate is 97,73% in train and 85,83% in validation.

6 Results comparison table

Table 3: Different results obtained on the three models

	Architecture used			Numbers epoch	Accuracy	Validation-Accuracy	Val_loss
	Convolution Layer	Pooling Layer	Fully Connected				
Vgg16	13	05	03	10	98,00%	80,11%	67,35%
Inception V 3	48	02	01	32	99,85 %	97,77 %	07,97 %
ResNet	50	02	01	15	97,73 %	85,83 %	61,70 %

The table shows the architecture used in each model as well as the number of eras. The obtained results are expressed in terms of learning accuracy, validation, trial and error, and finally execution time. Implementation time is very expensive. This is due to the large size of the base which requires the use of the graphics processing unit (GPU) instead of the central processing unit (CPU). Inception V3 provided the best results found.

In general, a large and deep convolutional neural network gives good results and the performance of our network deteriorates if the convolutional layer is removed. Therefore, depth is essential to achieve good results.

For example, according to the table, deleting one of the two middle layers leads to a loss of about 5% in network performance. Therefore, depth is essential to achieve good results.

The results obtained improved as our network deepened and the number of epochs increased. Learning base is also a crucial component of convolutional neural networks, it is necessary to have a large learning base to achieve the best results.

7 Predict classes for new images using the best model (Inception V3)

We brought 4 new photos that the model did not know before

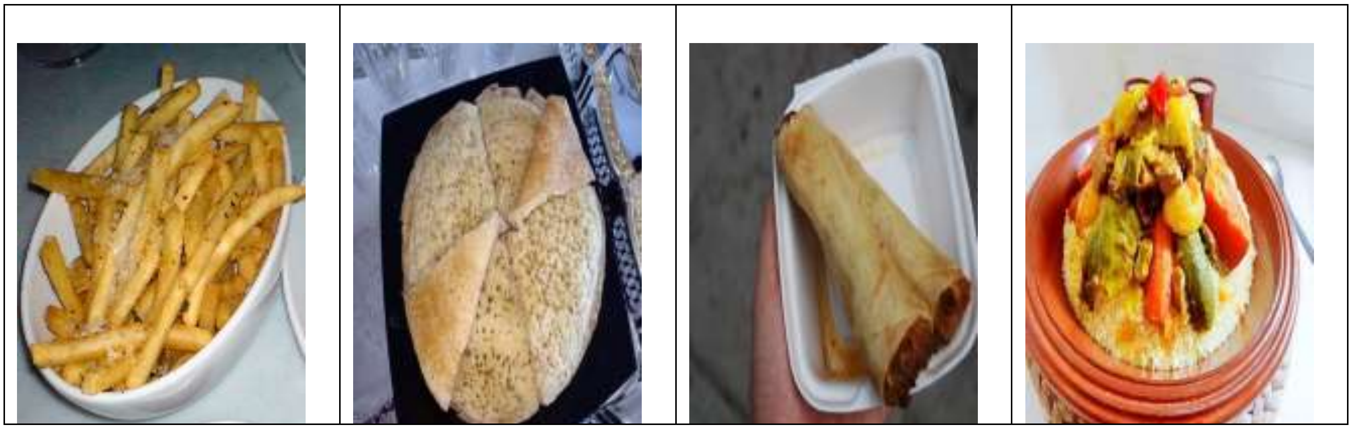
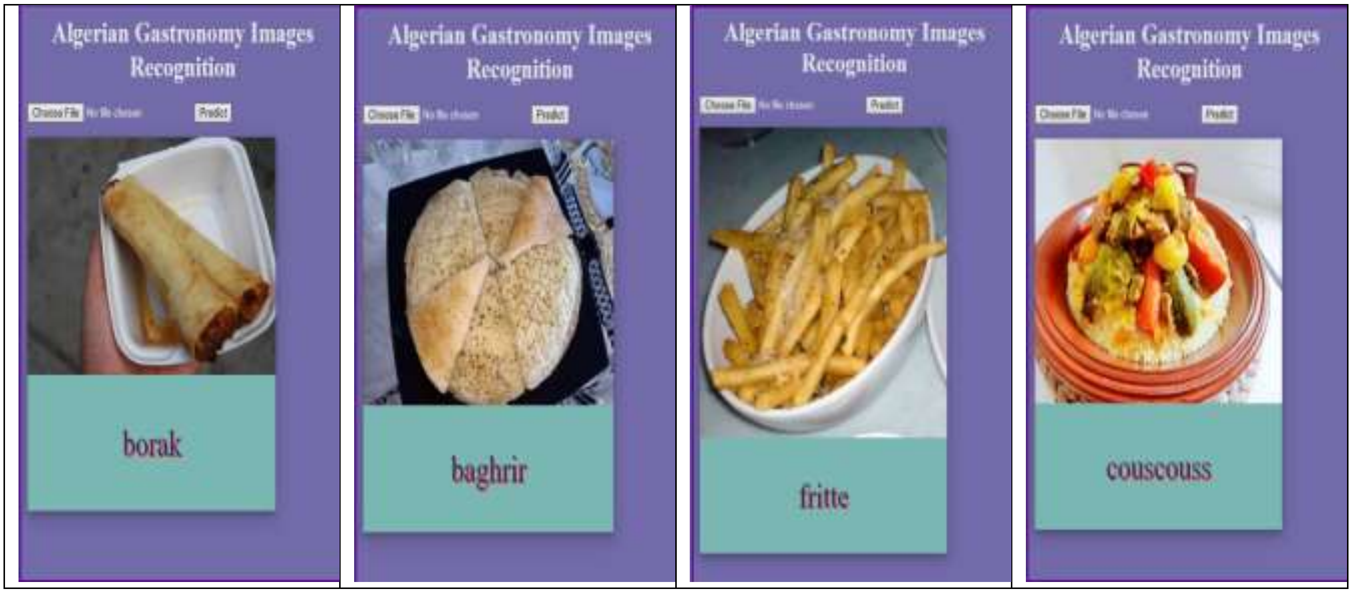


Figure 60:Pictures of new dishes

We used Flask to apply the model and test it on the new images



Algerian Gastronomy Images Recognition

Choose File No file chosen
Predict

borak

Algerian Gastronomy Images Recognition

Choose File No file chosen
Predict

baghrir

Algerian Gastronomy Images Recognition

Choose File No file chosen
Predict

fritte

Algerian Gastronomy Images Recognition

Choose File No file chosen
Predict

couscouss

Figure 61:Test result

8 Conclusion

In this chapter we introduced a classification approach based on convolutional neural networks, so we used three models with different architectures and showed the different results obtained in terms of accuracy and error. A comparison of the results found showed that the number of eras, the size of the base and the depth of the networks are important factors for obtaining better results

General conclusion

In conclusion, recognizing Algerian dishes using deep learning represents a significant advancement in the field of information technology and artificial intelligence. This innovative approach allows us to develop models capable of distinguishing and accurately classifying different dishes, contributing to a deeper understanding and appreciation of Algeria's rich culinary heritage.

By leveraging deep neural networks, we can analyze images and extract the intricate features that differentiate each dish from others. With this precise recognition, we can harness the power of deep learning to develop food recommendation systems, enhance cooking and culinary experiences, and promote Algerian cuisine locally and globally. Additionally, it can aid in preserving culinary traditions and facilitating cultural exchanges.

Undoubtedly, deep learning technology will continue to evolve and improve, opening up further discoveries and applications in the field of recognizing Algerian dishes and related domains. Research and development in this field will continue, contributing to enhanced performance, accuracy, and expanding the knowledge base of Algerian cooking.

In summary, utilizing deep learning for the recognition of Algerian dishes represents an exciting and promising advancement in the field of information technology, providing opportunities for growth and exploration in the realm of Algerian cuisine

Source list

1. Mohamed SANDELI, traitement d'images par des approches bio-inspirées application à la segmentation d’images, diplôme de Magister en informatique, université Constantine 2. 2014.
2. http://thesis.univbiskra.dz/2271/6/Chapitre%2003.pdf
3. Digital Imaging: A Practical Approach
4. Digital Imaging: A Practical Approach
5. Digital Imaging: A Practical Approach
6. Classification Automatique des images de Nourritures, master Réseaux et télécommunication
7. Mémoire Master Professionnel,Domaine : Informatique et Technologie de l'Information,Spécialité : Réseaux convergence et sécurité,Présenté par : Sarra BENFRIHA et Asma HAMEL
8. Noura Zouad, “Généralités sur le traitement d’images”
9. Mr. Karim Mezzoug, “TRAITEMENT ET ANALYSE DES IMAGES NUMERIQUES”,
10. Mémoire Master Professionnel,Domaine : Informatique et Technologie de l'Information,Spécialité : Réseaux convergence et sécurité,Présenté par : Sarra BENFRIHA et Asma HAMEL
11. Deep Learning for Vision Systems by Mohamed Elgendy
12. A Review on Image Processing and Image Segmentation, Jiss Kuruvilla, Dhanya Sukumaran, Anjali Sankar, Siji P Joy
13. Computer Vision: Algorithms and Applications De Richard Szeliski
14. https://www.red-gate.com/simple-talk/development/data-science-development/introduction-to-artificial-intelligence/
15. https://en.wikipedia.org/wiki/History_of_artificial_intelligence
16. https://www.javatpoint.com/types-of-artificial-intelligence
17. https://www.interviewbit.com/blog/applications-of-artificial-intelligence/?amp=1
18. https://www.ibm.com/design/ai/basics/ai
19. https://www.techtarget.com/searchenterpriseai/definition/AI-Artificial-Intelligence
20. https://www.ibm.com/topics/machine-learning
21. https://www.mathworks.com/discovery/machine-learning.html
22. https://towardsdatascience.com/clustering-unsupervised-learning-788b215b074b
23. https://towardsdatascience.com/clustering-unsupervised-learning-788b215b074b
24. Olivier Chapelle Semi-Supervised Learning (Adaptive Computation and Machine Learning series)
25. https://www.coursera.org/specializations/reinforcement-learning
26. https://www.investopedia.com/terms/n/neuralnetwork.asp by JAMES CHEN.
27. https://www.xenonstack.com/blog/artificial-neural-network-applications.
28. https://www.knowledgehut.com/blog/data-science/types-of-neural-networks#what-are-neural-networks.
29. https://www.knowledgehut.com/blog/data-science/types-of-neural-networks.
30. https://www.mygreatlearning.com/blog/types-of-neural-networks/

31. https://viso.ai/deep-learning/deep-neural-network-three-popular-types/
32. https://www.knowledgehut.com/blog/data-science/types-of-neural-networks
33. https://deepai.org/machine-learning-glossary-and-terms/sigmoid-function
34. https://www.ibm.com/topics/convolutional-neural-networks
35. Mémoire de fin d’études, diplôme de Master en informatique par Mokri Mohammed Zakaria “Classification des images avec les réseaux de neurons convolutionnels”
36. Dilmi Abdelbaqi, Chikouche aboubaker, “Failure prediction in cloud environment using deep learning” UNIVERSITY MOHAMED BOUDIAF - M'SILA
37. Issame mahdjoubi, Master thesis Automatique UNIVERSITY MOHAMED BOUDIAF - M'SILA
38. https://deepai.org/machine-learning-glossary-and-terms/inception-module.
39. Christian Szegedy, “Going deeper with convolutions”
40. https://www.kaggle.com/code/blurredmachine/vggnet-16-architecture-a-complete-guide
41. https://www.mygreatlearning.com/blog/resnet/
42. ‘Deep Residual Learning for Image Recognition’ Kaiming He et al
43. https://keras.io/api/optimizers/adam/?fbclid=IwAR2zeWVUV3UiJ4iiSwLZYNYYb7rM-lVb8SWapQKcyA9BHvqwxTC4iNQ-O2Y
44. Oquab, M., Bottou, L., Laptev, I., & Sivic, J. (2014). Learning and transferring mid-level image representations using convolutional neural networks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, 1717-1724
45. Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2017). Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization. Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017, 618-626
46. Goodfellow, I. J., Shlens, J., & Szegedy, C. (2015). Explaining and Harnessing Adversarial Examples. Proceedings of the International Conference on Learning Representations (ICLR), 2015
47. Frome, A., Corrado, G. S., Shlens, J., Bengio, S., Dean, J., Ranzato, M., & Mikolov, T. (2013). Devise: A Deep Visual-Semantic Embedding Model. Proceedings of the Neural Information Processing Systems (NeurIPS), 2013
48. Hokuto Kagaya, Kiyoharu Aizawa, Makoto Ogawa, “Food Detection and Recognition Using Convolutional Neural Network”
49. D. Pishva, A. Kawai, K. Hirakawa, K. Yamamori, and T. Shiino,“Bread Recognition Using Color Distribution Analysis,”
50. Taichi Joutou and Keiji Yanai “A FOOD IMAGE RECOGNITION SYSTEM WITH MULTIPLE KERNEL LEARNING”
51. https://docs.python.org/fr/3/tutorial/
52. https://research.google.com/colaboratory/faq.html
53. https://geekflare.com/google-colab/
54. https://docs.conda.io/en/latest/

55. https://code.visualstudio.com/docs
56. https://keras.io/about/
57. https://www.mygreatlearning.com/blog/what-is-tensorflow-machine-learning-library-explained/
58. https://scikit-learn.org/stable/getting_started.html
59. https://opencv.org/about/
60. https://www.w3schools.com/python/numpy/numpy_intro.asp
61. https://www.geeksforgeeks.org/python-introduction-to-web-development-using-flask/
62. https://www.kaggle.com/datasets/kmader/food41?resource=download
63. https://www.kaggle.com/code/blurredmachine/vggnet-16-architecture-a-complete-guide
64. https://datagen.tech/guides/computer-vision/vgg16/
65. https://blog.paperspace.com/popular-deep-learning-architectures-resnet-inceptionv3-squeezenet/
66. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR) (pp. 770-778). IEEE.
67. https://www.geeksforgeeks.org/residual-networks-resnet-deep-learning/