UniversityofKasdiMerbah- Ouargla

FacultyofNewTechnologiesofInformationandCommunicationDepartmen tofElectronicandTelecommunication

FINALSTUDYDISSERTATION

IntheaimofobtainingMASTERDegreeACADEMIC

Domain: Science and Technology

Option:Automatic and Systems

Specialty:Automatic

Presentedby:

Naimi Mohamed Chouaib Naimi Mohamed Amine Benatallah Mohammed Tewfik

Topic:

Medical Image Classification with Convolutional Neural Network

WasPubliclyDebatedon:16.09.2021 in front of The ExaminingCommittee

ComposedFrom:

Academic year:2020/2021

Abstract

Image patch categorization is a critical job in a wide range of medical imaging applications. We created a customized Convolutional Neural Network (CNN) with a shallow convolution layer to categorize lung image patches with interstitial lung disease in this study (ILD). Despite the fact that numerous feature descriptors have been developed in recent years, they can be extremely complex and domain-specific. Our customized CNN framework, on the other hand, can learn the intrinsic image characteristics from lung image patches that are most appropriate for classification automatically and effectively. The same architecture may be used to classify medical images or textures in various ways.

CNN: convolutional neural network **ANN** : Artificial neural network

DEEP LEARNING: is a type of machine learning and artificial intelligence

MACHINE LEARNING: is a type of artificial intelligence

ARTIFICIAL INTELLIGENCE: is intelligence demonstrated by machines

Résumé

La catégorisation des patchs d'images est un travail essentiel dans un large éventail d'applications d'imagerie médicale. Nous avons créé un réseau neuronal convolutif (CNN) personnalisé avec une couche de convolution peu profonde pour catégoriser les patchs d'image pulmonaire avec une maladie pulmonaire interstitielle dans cette étude (ILD). Malgré le fait que de nombreux descripteurs de caractéristiques ont été développés ces dernières années, ils peuvent être extrêmement complexes et spécifiques à un domaine. Notre cadre CNN personnalisé, d'autre part, peut apprendre les caractéristiques intrinsèques de l'image à partir des patchs d'image pulmonaire qui sont les plus appropriés pour une classification automatique et efficace. La même architecture peut être utilisée pour classer des images ou des textures médicales de diverses manières.

CNN : réseau de neurones convolutifs **ANN** : Réseau neuronal artificiel

L'APPRENTISSAGE EN PROFONDEUR: est un type d'apprentissage automatique et d'intelligence artificielle

APPRENTISSAGE MACHINE : est un type d'intelligence artificielle

INTELLIGENCE ARTIFICIELLE : l'intelligence est-elle démontrée par les machines

Acknowledgments

First of all, wethank ALLAH for havinggiven us the courage and patience to arrive at thismodestwork.

Wewould like to thankoursupervisor Dr. LATI Abdelhai, who has been alwaysgenerousduring all phases of the research, his follow-up and advice throughout thisperiod.

Our mostgratefulthanks to all thosewho have contributed to the realization of this memory.

Finally, wewould like to express ourtremendous gratitude to ourfamily and ourfriends for their support duringtheseyears.

DEDICATION

Wededicatethismodestworktothosewhoarethesourceofouri nspirationandourcourage.

Tomydearmother,whoalwaysgivesmehopetoliveandwhoha sneverstoppedprayingforme.

Tomydearfather,forhisencouragementandsupport,andabov eallforhissacrificesothatnothingwillhinderthecourseofmystu dies.

Toalltheprofessorsandteacherswhohavefollowedme throughout myschoolingandwhohaveallowedmetosucceedinmystudies .

To mydearbrothers

To mydearsisters

All myfriends

Contents

Liste of figures

General Introduction

The security and safety of individuals, properties and information need to be guaranteed , actually one of the major concerns of our societies, especially after the great spread of terrorism around the world. In fact, people willing to cross boundaries must prove their identities using their passports, people willing to cross buildings or academic institution must validate their access cards, people desiring access to banking services must login using a login and a password. Nevertheless, these traditional methods show great weaknesses for identity verification. Indeed, the identity of a person is directly related to that they possess (such as passport, access card, etc.) or/and that they know (password, PIN codes, etc.). Nonetheless, PIN codes and passwords may be forgotten or compromised and access cards may be falsified or duplicated which lead to identity sponge. In this respect, experts are looking for a technology which resolves these problems by giving more convenience to persons and ensuring a highly secured access, by relating the identity of a person to what they are and not to what they possess or know. Biometry is the most suitable technology for identity verification and/or person identification by employing their physiological features including biological, morphological and behavioral characteristics. This technology makes identity data theft more difficult and thus, increases user confidence as the physical presence is necessary during identification.[1] In our work, we have chosen facial recognition as an average of identification compared to other methods because this identification is naturally used by a human being, so this type of recognition does not stop with the identification of the face, but can apply to the location of an individual in a crowd, unlike other methods, and does not require very complex acquisition equipment that is to say a simple camera can acquire the shape of an individual’s face and then remove certain features. The essential features for face recognition are eyes, mouth, face, nose, etc. Depending on the system used, 1 the individual must be positioned in front of the camera where they may be moving at a distance. The biometric data that is obtained is compared to the reference file. The software must be able to identify an individual despite various physical devices (mustache, beard, glasses, etc.).[2] This thesis deals with a topic of identification. An identification system is intended to answer the question; ”who is this person?” You have to check the biometrics against all others in the database. Which is simplified to 1: Many It is, therefore, responsible for discovering the identity of an unknown person in a dataset. Several methods have been developed in the literature for face recognition. In our work, we have opted for a technique based on neural networks called the Convolutional Neural Networks (CNN) which is a type of neural network with deep learning, or Deep Neural Network. The latter has several hidden layers. CNN consists of two very distinct parts, part of extraction that can be used to simplify an input image, reducing its size, and part of classification that classifies this data.[2]

We chose to articulate our study around four main chapters. The first chapter is devoted to the general presentation of biometrics. It describes the operating principle of biometric systems and then defines the tools used to evaluate their performance. Then, the place of facial recognition among the other biometric techniques is analyzed. Through this chapter, we want to position the problem of facial recognition and present its issues and interests to other techniques. Finally, we highlight the difficulties faced by face recognition systems. In the second chapter, we will discuss the state of the art of face recognition techniques. We present just the most popular face recognition and face detection algorithms, and quote some of the most used databases for face recognition. The third chapter is composed of two parts. We will first take on the artificial neural network basics (ANN), which is the heart of the recognition system. Then we talk about recognition techniques based on deep neural network (Deep Learning) of the convolutional neural network-type (CNN) in the second part. In the fourth chapter, we present the experimental results obtained by methods of face recognition that we choose and analyze their performance, followed by a discussion with the interpretation of the results. Finally, the general conclusion will summarize the results obtained by our approach.

CHAPTER I

I.1.Introduction

Gone are the days, when health-care data was small. Due to the tremendous advancement in image acquisition devices, the data is quite large (moving to big data), that makes it challenging and interesting for image analysis. This rapid growth in medical images and modalities requires extensive and

Medical images have varied characteristics depending on the target organ and the suspected diagnosis. Common modalities used for medical imaging include X-ray, computed tomography (CT), diffusion tensor imaging (DTI), positron emission tomography (PET), magnetic resonance imaging (MRI), and functional MRI (fMRI) [1–4]. In the past thirty years, these radiological image acquisition technologies have enormously improved in terms of acquisition time, image quality, resolution [5–8] and have become more affordable. Despite improvements in hardware, all radiological images require subsequent image analysis and diagnosis by trained human radiologists [9]. Besides the significant time and economic costs involved in training radiologists

Applications of image

- Comparison of Different Features and Classifiers for Driver Fatigue Detection Based on a Single EEG Channel

Traffic accidents aremore and more increasing, resulting in avery large number of casualties. Safety driving is fundamentalto public health, and fatigue driving can be life threatening.It is crucial and necessary to develop some technologies fordetecting driver fatigue. There are many methods thathave been proposed in the past fewyears, such as vehicle drivingparameters by using various sensors, driver behaviorcharacteristics by using video imaging techniques, driver physiological parameters by using acquisition andanalysis of electrocardiogram (ECG), electrooculogram (EOG), electromyogram (EMG), and EEG. As a kind of direct indicator of the brain status, EEG isconsidered as the “gold” method to identify driver fatigue.

EEG is an objective method for the evaluation of brainstate and function, which is often used in auxiliary diagnosisof illness such as epilepsy and seizure. The advantages ofEEG are sensitivity for analysis and being relativelycheapfor acquisition. Various computational approaches based onEEG signals have been developed foranalyzing anddetectingdriver fatigue.

Fu et al. proposed a fatigue detection model based on Hidden Markov Model and fused physiological and contextual knowledge to assess probabilities of fatigue. They achieved highest accuracy of 92.5% based on EEG signals from two channels (O1 and O2) and other physiological signals. Li et al. collected 16 channels of EEG data and computed12 types of energy parameters. The number of significant electrodes is reduced using Kernel PrincipalComponent Analysis (KPCA). The experimental results from two channels (FP1 and O1) achieved the highest accuracy of 91.5%. Wali et al. used Discrete Wavelet Transforms to process the EEG signal for fatigue detection and yielded the highest accuracy of 85%. Using Fast Fourier Transform, Simon etal. proposed EEG alpha spindle measures for assessing driver fatigue. Charbonnier et al. made use of the Frobeniusdistance between the EEG spatial covariance matrices of 6 brain regions, and experimental results had shown that the index based on the alpha band can accurately assess fatigue. Apker et al. predicted driver performance using power spectral density and the linear regression, providing a confidence estimate for the stable driving model. Hajinorooziet al.'s experimental results showed that channel-wise convolutional neural network achieved robust and improved performance for detection of driver fatigue. Zhao et al. studied an automatic measurement of driving mental fatigue, using a KPCA-SVM classifier and their accuracy was quite high, up to 98.7%. Kong et al. analyzed EEG signalsby using Granger-Causality-based brain effective networks and found a significant difference in terms of strength of Granger-Causality in the frequency domain and some changes were more significant over the frontal brain. Zhao etal. observed that coherence was significantly increased in the frontal, central, and temporal brain regions, as well as significant increases in the clustering coefficient and the character path length.

Recently, entropy has been broadly applied in the analysis of EEG signals, considering the fact that it is a complex, unstable, and nonlinear signal.Xiong et al. combined features of AE and SE with Support Vector Machine (SVM)classifier to detect driver fatigue, achieving highest accuracy of 91.3% at channel P3. Chai et al. present independent component by entropy rate bound minimization analysis for the source separate, autoregressive (AR) modeling for the features extraction and Bayesian neural network for the classificationalgorithm. They achieved an accuracy of 88.2% and the highest value of area under the receiver operating curve(AUC) is 0.93. Zhang et al. extracted wavelet entropy andSE of EEG and wavelet entropy of EOG and AE of EMG to estimate the driving fatigue stages, and their accuracy wasquitehigh, which is about 96.5%–99.5% using artificial neural network. Kar et al. used five types of entropies, that is, Shannon's entropy, R´enyi entropy of order 2, R´enyientropyof order 3, Tsallis wavelet entropy, and Generalized Escort-Tsallis entropy, along with alpha band relative energy for estimation of fatigue level [28]. However, few studies have been conducted for using optimal combination of entropy methods and classifiers based on EEG to study driver fatigue detection.

Multichannel EEG acquisition system, such as the 32-channel EEG system used in my experiment, is relatively complex equipment, which can only be available in laboratories or hospitals. It requires well-trained technicians to locate electrodes, since all the electrodes have to be placed in the proper location. And it is time-consuming. All these reasons remaking the system difficult to apply in real life.Thereforworthwhile EEG system with fewer channels or even one channel for estimating driver fatigue has to be a portable system that is cheaper, simpler, and easier to use.

Although many EEG-based methods have been proven to detect driver fatigue, the optimal method has not yet been determined. Furthermore, the EEG with more channels usually restricts its application in the detection of driver fatigue. Using the data from 12 subjects, the detection model for driver fatigue was developed with a single channel. Four types of entropies were deployed in this work: SE, FE, AE,and PE. The classification procedure was implemented by ten classifiers: K-Nearest Neighbors (KNN), SVMwithline AR kernel (LS), SVM with RBF kernel (RS), Gaussian Process (GP), Decision Tree (DT), RF, Multilayer Perceptron (MLP), AdaBoost (AB), GaussianNaïve Bayes (GNB), andQuadraticDiscriminantAnalysis (QDA).

Figure I.1: Snapshot of the experimental setup.

- Topological Measurements of DWI Tractography for Alzheimer's Disease Detection

Novel approach to characterize the brain organization from a topological perspective is presented. In particular, because of the well-known and stereotyped pattern characterizing AD, we chose to use this pathology as a benchmark. A topological score and a weighted variant have been defined and used to train support vector machines on a mixed NC/AD cohort. Results showed that topological information was able to efficiently detect diseased patterns (AUC = 95%, 95% confidence interval 92%–99%).

We also addressed in this study the problem of quantifying the effect of combining MRI-based features with topological ones. We found that their combination can improve classification accuracy; nonetheless, this is strictly related to the quality of structural features used. In fact, when using all MRI features available the classification performance decreased; on the contrary, it was slightly raised using hippocampal volumes whose association with AD is well known. A subtle effect should be better investigated on larger cohorts.

The performance obtained is comparable with best results reported in the literature so far, but possible improvements could include a more refined study of weighted networks, instead of their binary version; nevertheless, this cannot be considered a limitation of the present study, whose main goal was to investigate the brain topology and understandwhether the topological measures proposed were suitable for clinical purposes.

The presented methodology is general, even if in this case it has been tailored on Alzheimer's disease. For future work, we propose to investigate the application of thismethodology to mixed cohorts including alsoMCI subjects, trying to tacklethe discrimination problem between subjects converting toAD or not, and the early diagnosis of AD. Patients affected by neurodegenerative diseases incur a cognitive impairment which could be effectively diagnosed and monitored by these measurements, a useful trait for technological innovations in the e-health field, for example, for remote medicine applications, or for pharmacological industries, aiming at the development of drug therapies and clinical trials. Further investigations could be aimed at diseases affecting the brain organization with less stereotyped patterns.

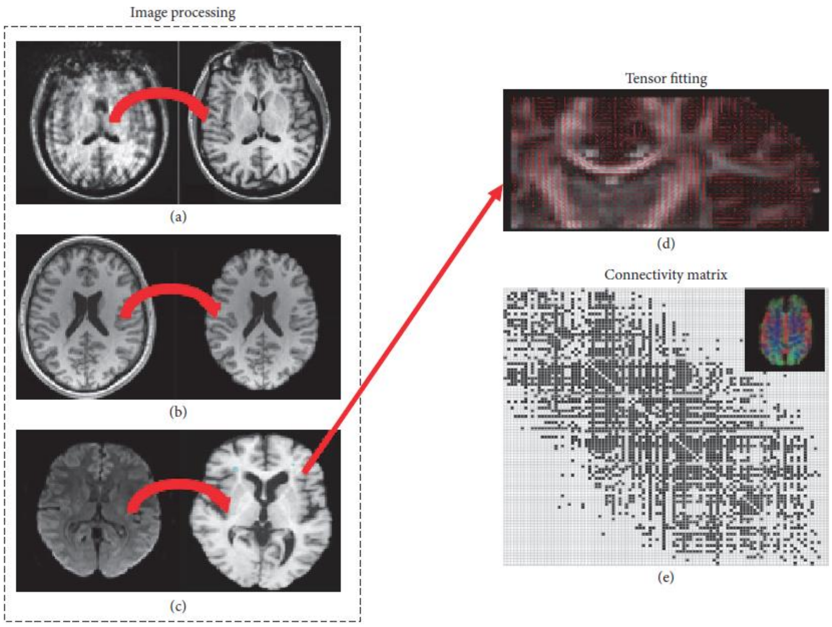


Figure I.2: the processing pipeline underwent by brain DWIscans.

The dotted box includes the dedicated image processing steps: (a) eddy correction, (b) brain extraction, and (c) affine registration. For each voxel the diffusion tensor was estimated, (d) thus allowing the probabilistic fiber reconstruction. Using the Harvard-Oxford atlas, the connectivity matrix derived from tractography was computed foreach subject.

- A Fusion-Based Approach for Breast Ultrasound Image Classification Using Multiple-ROI Texture and Morphological Analyses

An effective approach for BUS image classification is proposed. Texture analysis is carried out by dividing the tumor into a set of nonoverlapping ROIs and processing each ROI individually to estimate its tumorclassindicator. The tumor class indicators of all ROIs inside the tumor are combined using a majority voting mechanism to estimate the posterior tumor class likelihood. In addition to the multiple-ROI texture analysis, morphological analysis is used to estimate the posterior tumor class likelihood. A probabilistic approach is employed to fuse the posterior tumor class likelihoods obtained using the texture and morphological analyses. The proposed approach has been employed to classify 110 BUS images. The classification results indicate that the proposed approach achieved classification performance that outperforms conventional texture and morphological analyses. In particular, fusing the multiple-ROI texture analysis and morphological analysis enabled classification accuracy of 98.2%, specificity of 98.4%, and sensitivity of 97.8%. These results suggest that the proposed approach has the potential to provide the radiologists within accurate second opinion to reduce the rate of expendable biopsy and minimize BUS image misdiagnosis.

- A Fusion-Based Approach for Breast Ultrasound Image Classification Using Multiple-ROI Texture and Morphological Analyses

Lung cancer is becoming one of the main threats to human health at present in the world. The number of deaths caused due to lung cancer is more than prostate, colon, and breast cancers [1]. Early detection of solitary pulmonary nodules (SPNs) is an important clinical indication for early-stage lung cancer diagnosis because SPNs have high probabilities to become malignant nodules [2, 3]. SPNs refer to lung tissue abnormalities that are roughly spherical with round opacity and a diameter of up to 30 mm.

It is therefore an important task to develop computer aided detection (CAD) systems that can aid/enhance radiologist workflow and potentially reduce false-negative findings. CADis a scheme that automatically detects suspicious lesions (i.e., nodule, polyps, and masses) in medical images of certain body parts and provides their locations to radiologists. CAD has become one of the major research topics inmedical imaging and diagnostic radiology and has been applied to various medical imaging modalities including computed tomography (CT), magnetic resonance imaging (MRI), and ultrasound imaging. Generally, typical CAD systems for cancer detection and diagnosis (i.e., breast, lung, and polyp) cover four stages as depicted in Figure 1(a), including candidate nodule ROI (Region of Interest) detection, feature extraction, and nodule classification.Thestagesof feature extraction and nodule classification belong to the false-positive reduction step. Current CAD schemes for nodule characterization have achieved high sensitivity levels and would be able to improve radiologists' performance in the characterization of nodules in thin-section CT, whereas current schemes for nodule detection appear to report many false positives. It is because detection algorithms have high sensitivity that some nonnodular structures (e.g., blood vessels) are labeled as nodules inevitably in the initial nodule identification step. Since the radiologists must examine each identified object, it is highly desirable to eliminate these false positives (FPs) as much as possible while retaining the true positives (TPs). Therefore, significant effort is needed in order to improve the performance levels of current CAD schemes for nodule detection in thin-section CT.

The purpose of false-positive reduction is to remove these false positives (FPs) as much as possible while retaining a relatively high sensitivity. It is a binary classificationHindawiPublishing Corporation between the nodule and non-nodule, aiming to develop new methods in order to accurately distinguish suspicious regions, leading to significant reduction of FPs with machinelearningtechniques. The false-positive reduction step, or classification step, the aim of which is to learn a system capable of the prediction of the unknown output class of a previously unseen suspicious nodule with a good generalization ability, is a critical part in the lung nodule detection system. Classification plays an important role in the reduction of false positives in lung computer aided detection and diagnosis methods. Deep learning can be used for both classification and feature learning in various fields such as computer vision and speech. In our work, a deep convolutional neural network proposed for pulmonary nodule classification using theLIDC database. The method used in CAD system replaces the two components of feature extraction and classification. The input of deep convolution neural networks in this work is ROI pixel data directly without feature extraction and selection. Compared with the traditional methods, the approach in our work has four advantages as follows.

(i) The representation of nodule ROI is critical for discrimination between true nodule and false nodule However, it is difficult to obtain good feature representations by human efforts. Our method can learn good feature representation of ROI without feature extraction and selection.

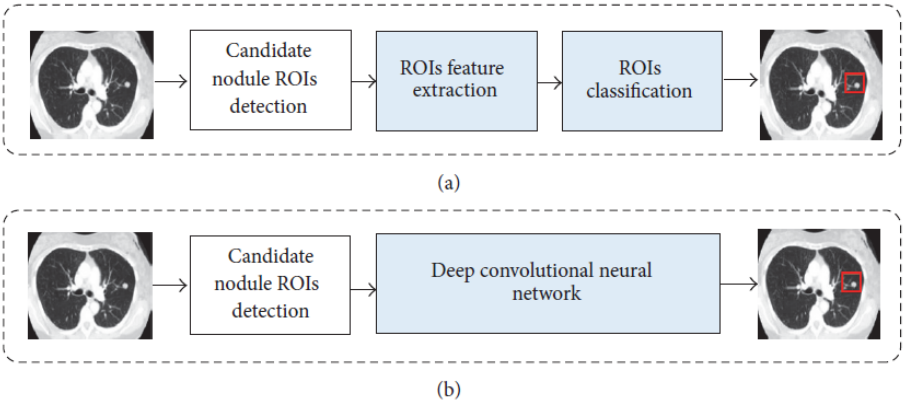
(ii) Our method takes advantage of the relationships between the internal region and external region of ROI, so as to learn more discriminative knowledge for false-positive reduction.

(iii) Our method can be executed based on the Centre of the ROI rather than the whole ROI region.Therefore, there is no necessity to obtain the exact margin of the nodules detected in the first step of CAD system.

(iv) The neural networks are trained by large scale ROIsdata with nodules and no nodules more than 60thousand which are the largest in our knowledge. Soothe neural network is capable of recognizing a wide range of representations of nodules.

- Pulmonary Nodule Classification with Deep Convolution al Neural Networks on Computed Tomography Images

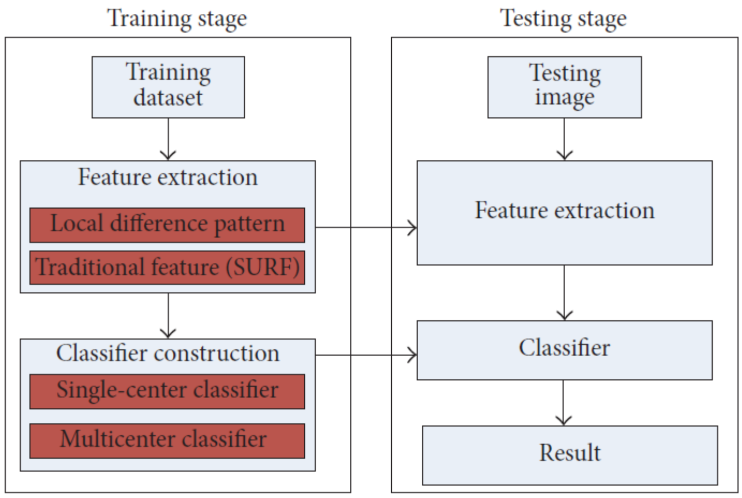
In this paper, a method of pulmonary nodule recognition using deep convolution neural networks is presented. The deep convolution neural network can take advantage of the training dataset to enable the algorithm to automatically select the best representation as the feature representation of the image. Through the training of the training dataset, the approach obtains much more general characteristics of pulmonary nodules and higher accuracy while retaining relatively better robustness. We plan to extend the proposed method to be capable of benign and malignant classification in the future. The algorithm will be accelerated by GPUcomputing for convolution operation.



FigureI.3: The main components in a general CAD system (a) and the main components in our work (b).

- Lung Nodule Image Classification Based on Local Difference Pattern and Combined Classifier

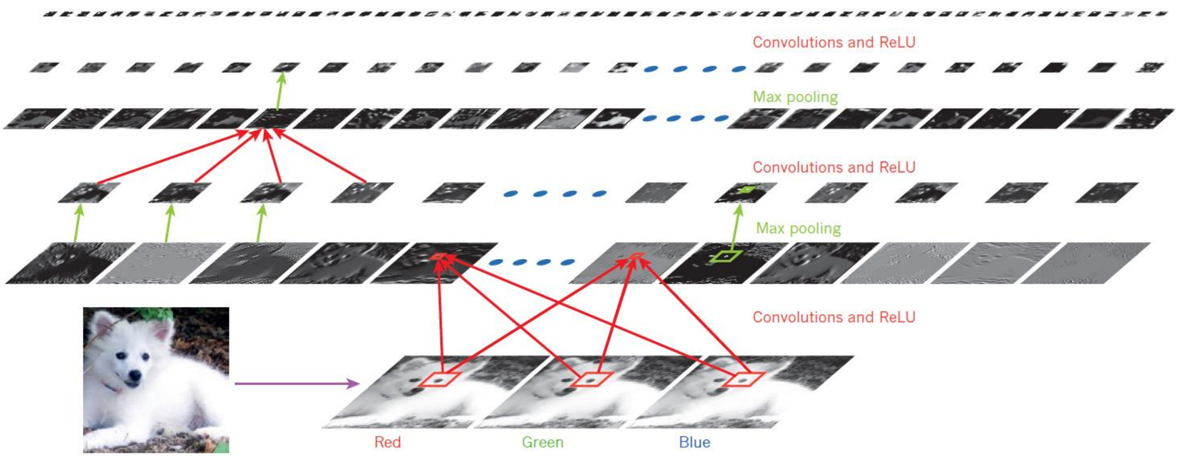
a method for lung nodule image classification. First, a novel local feature representation, LocalDifferencePattern, is designed, which can catch more information from the lung nodule and its neighbor regions. And a singlecenterclassifier is constructed according to LDP and SVM.Then, a multicenter classifier is designed by clustering theSURF feature of lung nodule image and computing the similarity between testing image andmultiple centers. Finally, the two classifiers are combined to implement the classification.The proposed method aims to extract more useful feature and decrease the gap between high variance intraclass and high similarity interclass. Evaluation on public dataset shows that our proposed method outperforms other methods for lung nodule image classification. Our future works will focus on designing more accurate feature representation methods for lung nodule image, such as autoencoder and convolutional neural network.



FigureI.4: Framework of the method.

Deep learning

Deep learning allows computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction. These methods have dramatically improved the state-of-the-art in speech recognition, visual object recognition, object detection and many other domains such as drug discovery and genomics. Deep learning discovers intricate structure in large data sets by using the backpropagation algorithm to indicate how a machine should change its internal parameters that are used to compute the representation in each layer from the representation in the previous layer. Deep convolutional nets have brought about breakthroughs in processing images, video, speech and audio, whereas recurrent nets have shone light on sequential data such as text and speech.



CHAPTER II Biometry and image classification

II.1 Introduction

Biometry is a growing technology which has become increasingly used in our daily life. It aims to establish the identity of a person as reliable as possible using their biological features in order to guarantee the safety of people in public places. In this chapter, we introduce firstly, the identity of a biometric system, structure and the different biometric modalities. Eventually, we will showcase one of the most efficient modalities to identify a subject.

II.2 Biometry

II.2.1 Definition of biometry

Biometry is the process of verifying an individual's identification using his biological features, which are divided into two categories. The first is physical characteristics, which are the most often utilized and rely on physical features of persons such as the iris, fingerprint, palmprint, face, and so on. The second type of feature is behavioral characteristics, which are less commonly employed and rely on specific acts or behaviors such as movement, speech, dynamic signature, and so on. Biometric modalities [1] are physical and behavioral features that allow people to be identified.

Biometry is the study of determining how to quantify these unique characteristics and how to utilize them to differentiate people. Biometrics researchers are working to automate these procedures and make them more appropriate. A biometric system [3] will execute it on a computer or a device.

II.2.2 Biometric modalities

There are many different biometric modalities that are used to get information on human personal qualities, and they are categorized into three basic groups (biological, behavioral, and morphological). Today, the most used modalities are fingerprint, face, iris, and voice. These are the biometric modalities that, as of now, best satisfy the criteria for uniqueness, permanence, and consistency, as well as the simplicity with which they may be captured utilizing sensing equipment. This section addresses a variety of biometric modalities that are based on biological, behavioral, or morphological study.

• **Biological: This** classification is based on an examination of an individual's biological features. The idea of this sort of analysis is that each individual's biological data is a personal signature.

Odor, DNA, and physiological signals are all examples of biological analysis [4].

However, DNA analysis is not currently employed in biometrics for automatic user authentication for two reasons. First, extraction of DNA sequences still necessitates biochemical processing, which is currently not entirely automated and takes a long time.

The second reason is that organic material containing DNA is readily lost. As a result, it may readily be collected and re-used by other people, for as by taking a sample of a lost hair from a brush or spit from a glass [5].

• **Behavioral:** The examination of an individual's behavior, such as signature dynamics, demarche, typing, and voice [4], is the basis for this category.

It is primarily defined by three types of individual characteristics: the biological structure of the organs that create behavior, the acquired qualities of how to produce behavior, and the goal or intention of which action to be produced. Various aspects of the biological construction of the mouth, vocal cords, and glottis, for example, influence the individual sound characteristics of speech generation in speech-based biometrics. Learned features, on the other hand, include linguistic aspects such as voice tones, pronunciation, and speech pace, which are significantly impacted by how speaking ability is acquired [5].

• **Morphological:** This category is based on the individual's usage of physical characteristics that are distinctive and permanent. The face, the fingerprint, the shape of the hand, the iris, and other modalities have all been utilized to extract this information [4]. Physiological characteristics of people are biological structures that are unique to each person and may be obtained without collecting physical samples, such as by optical methods. These can be regarded as tangible consequences that are visible or at the very least quantifiable, since they are organically grown according to the genetic building code. The structure of ridges on fingers, for example, has been shown to be unique and permanent in most humans [5].

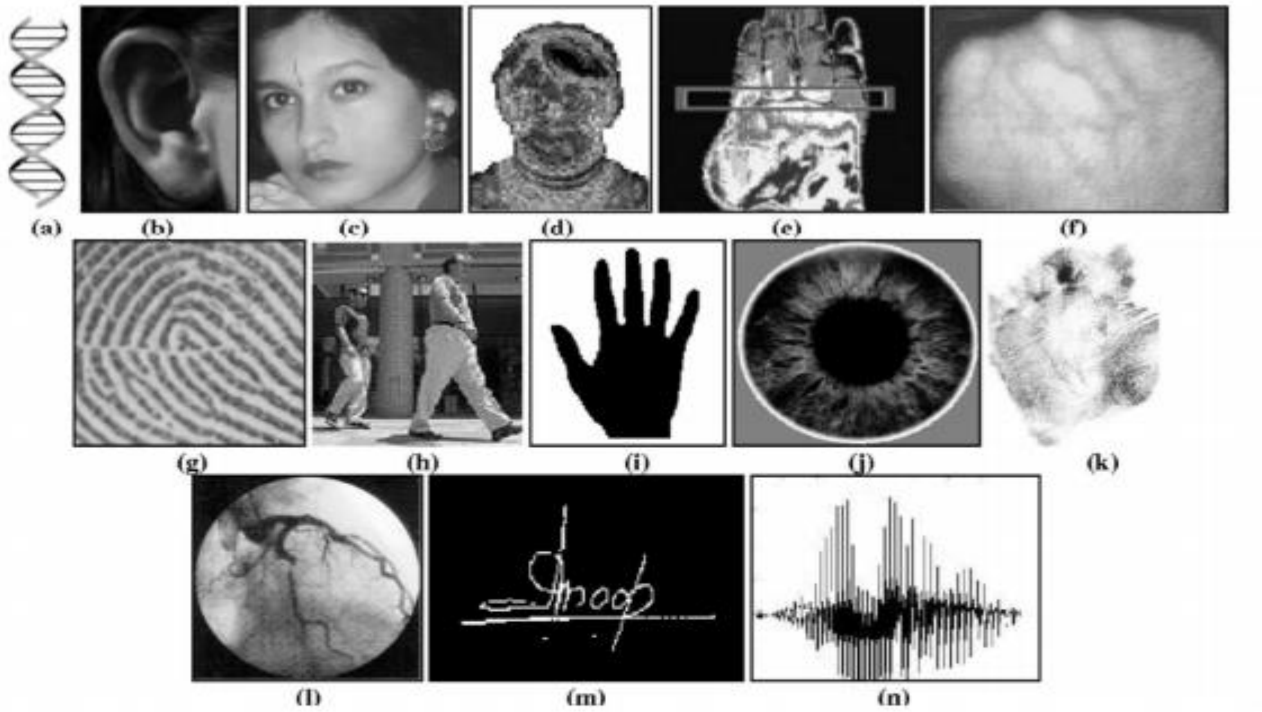


Figure II.1: Examples of biometric characteristics (a) DNA, (b) ear, (c) face, (d) facial thermogram, (e) hand thermogram, (f) hand vein, (g) fingerprint, (h) gait, (i) hand geometry, (j) iris, (k) palmprint, (l) retina, (m) signature, and (n) voice. [6]

Figure II.1: (a) DNA, (b) ear, (c) face, (d) facial thermogram, (e) hand thermogram, (f) hand vein, (g) fingerprint, (h) gait, I hand geometry, (j) iris, (k) palmprint, (l) retina, (m) signature, and (n) voice are examples of biometric traits. [6]

II.2.3 Biometric systems

A biometric system is simply a pattern recognition system that works by collecting biometric data from a person, extracting a feature set from that data, and comparing that feature set to a database template set (see Fig. I.2). A biometric system can function in either verification or identification mode, depending on the application environment.

• In verification mode, the system verifies a person's identification by comparing biometric data recorded with her own biometric template(s) stored in the system database. In such a system, a person who wants to be recognized claims his or her identity, usually through a personal identification number (PIN), a user name, or a smart card, and the system performs a one-to-one comparison to see if the claim is true or not (for example, "Does this biometric data belong to Bob?”). Typically, identity verification is used for positive recognition, with the goal of preventing multiple people from using the same identity.

The following is a formal formulation of the verification problem: Determine if (I, XQ) belongs to class w1 or w2 given an input feature vector XQ (extracted from biometric data) and a claimed identity I, where w1 indicates that the claim is true (a real user) and w2 indicates that the claim is incorrect (an impostor). To determine its category, XQ is usually compared to XI, the biometric template that corresponds to user I. Thus

$$(I, XQ) \in (w1, \text{ if } S(XQ, X1) \geq t \text{ } w2, \text{ otherwise } (I.1)$$

where S is the function that compares feature vectors XQ and XI, and t is a preset threshold. A similarity or matching score between the user's biometric measures and the claimed identity is defined as S (XQ, X1). As a result, depending on the variables XQ, I, XI, and t, and the function S, each claimed identity is categorized as w1 or w2. It's worth noting that biometric measures (such as fingerprints) collected at multiple times on the same person are virtually never similar. This is why the threshold t was introduced.

• In the **identification mode, where** S is the function that compares feature vectors XQ and XI, and t is a preset threshold. A similarity or matching score between the user's biometric measures and the claimed identity is defined as S (XQ, X1). As a result, depending on the variables XQ, I, XI, and t, and the function S, each claimed identity is categorized as w1 or w2. It's worth noting that biometric measures (such as fingerprints) collected at multiple times on the same person are virtually never similar. This is why the threshold t was introduced. For convenience, identification can also be utilized in positive recognition (the user does not have to assert an identity). While traditional means of personal identification such as passwords, PINs, keys, and tokens may work for positive identification, biometrics are the sole way to establish negative identification.

On the other hand, the identification problem may be phrased as follows. Determine the identity IK, K1, 2....N, N + 1 given an input feature vector XQ. The identities enrolled in the system are I1, I2, IN, and IN+1 denotes the reject situation, in which no acceptable identity for the user can be found. Hence

XQ ∈ (IK, if maxk {S (XQ, XIK)} ≥ t, K = 1, 2, ..., N IN+1, otherwise (I.2)

where XIK is the biometric template corresponding to identity IK, and t is a predefined threshold [6].

Before we go into the layout of the biometric system, it's important to understand that in order to identify/verify a subject, we'll need a database of individual templates, which will be filled up throughout the enrollment phase:

• **Enrollment** Both verification and identification mechanisms have this property. It is the preliminary phase in which a user's biometric data is initially recorded in the system. One or more biometric modalities are collected and saved as templates in the database during this step. This stage is critical since it has an impact on the entire recognition process later on. In reality, because gathered data are used as references for the individual, the quality of enrolled data is critical for further identification phases. To account for the variability of a person's biometric modality, a set of samples should be collected [1].

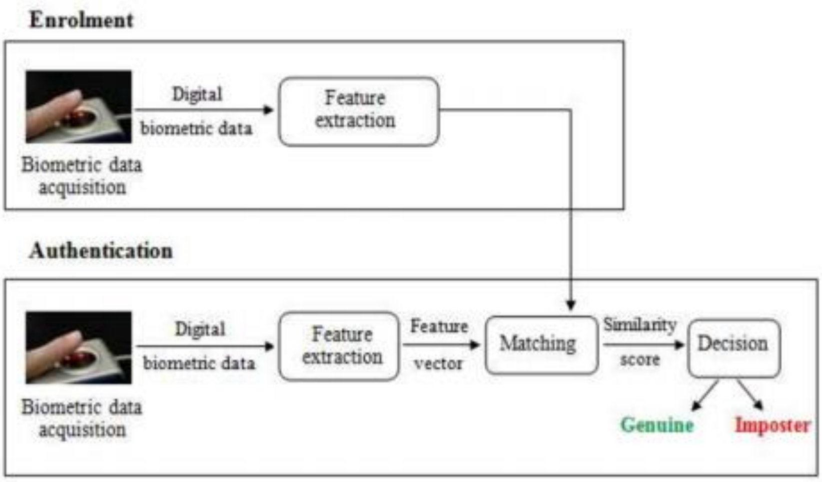


Figure II.2: Block diagrams of enrollment, verification, and identification [1]

II.2.4 Structure of a biometric system

The structure of a biometric system is composed of four modules. A biometric system is designed using the following four main modules (see Fig. I.6).

• **Sensor module** which records an individual's biometric data A fingerprint sensor, for example, captures the ridge and valley structure of a user's finger.

• **Feature extraction module** in which biometric data is gathered and processed to extract a collection of prominent or distinguishing characteristics. A fingerprint-based biometric system's feature extraction module, for example, extracts the position and orientation of minutiae points (local ridge and valley singularities) in a fingerprint picture.

• **Matcher module** in which matching scores are generated by comparing the characteristics collected during recognition to the stored templates. The number of matching minutiae between the input and template fingerprint pictures is determined and a matching score is provided in the matching module of a fingerprint-based biometric system, for example. The matcher module also includes a decision-making module, which uses the matching score to confirm (verification) or establish (identification) a user's stated identity.

• **System database module** which is utilized by the biometric system to store the enrolled users' biometric templates. The enrollment module is in charge of adding people to the biometric system database. During the enrolling step, a biometric scanner scans an individual's biometric characteristic to create a digital representation of the feature. Depending on the application, data collection throughout the enrollment process may or may not be overseen by a human. A quality check is usually carried out to verify that the obtained sample can be processed reliably by subsequent steps. The input digital representation is then processed by a feature extractor to create a compact yet expressive representation, referred to as a template, in order to enable matching. The template may be kept in the biometric system's central database or recorded on a smart card given to the person, depending on the application. To account for changes in the biometric characteristic, several templates of an individual are usually kept, and the templates in the database may be changed over time [6].

II.2.5 Performance of biometric systems

There are two sorts of faults to look for when evaluating the performance of a biometric system:

• **False Acceptance Rate (FAR):**which is when the system erroneously recognizes two different samples as samples from the same source F AR = number of accepted imposters (False Accept) total number of imposters' accesses (I.3)

• **False Rejection Rate (FRR): which** is when the system erroneously recognizes two samples from the same source as samples from different sources. F RR = number of rejected clients (False Reject) total number of client accesses (I.4)

We may compute the Equal Error Rate (EER) after computing the FAR and FRR. This rate is generated from the first two criteria and serves as a benchmark for current performance. This position corresponds to the point at which FRR = FAR, i.e., the optimum compromise between false rejections and false acceptances [7].

EER = number of false acceptances + number of false rejection total number of accesses (I.5)

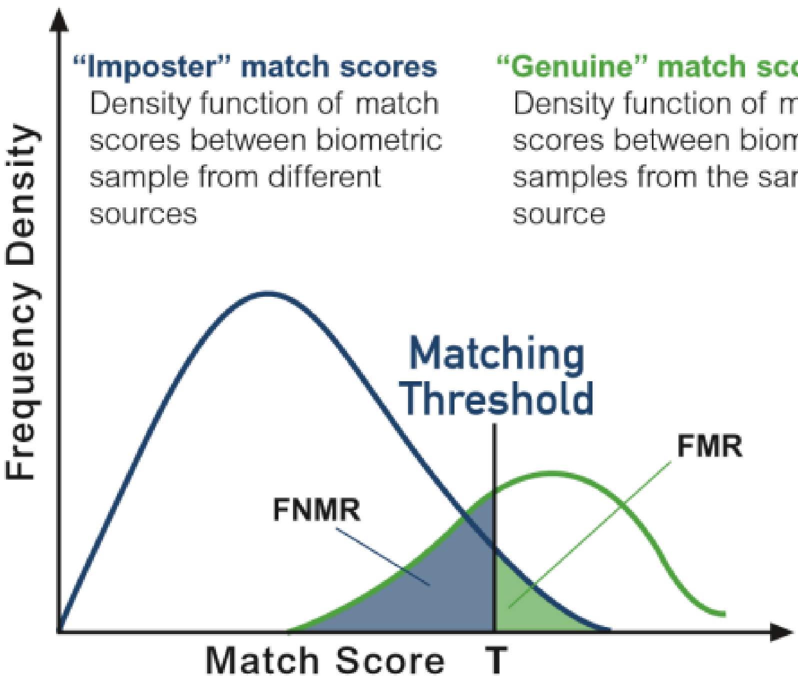


Figure II.3: FAR and FRR diagram

A **Receiver Operating Characteristic (ROC)** curve can be used to show the system's performance at all operating points (thresholds). A plot of FMR versus or FNMR for various threshold values is known as a ROC curve [6]. The more this curve resembles the form of the mark, the more efficient the system is with a high Recognition Rate (RR) [1].

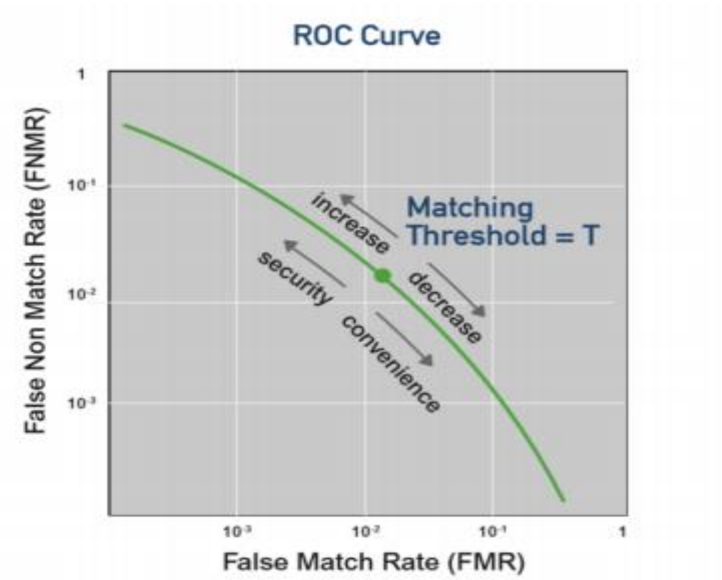


Figure II.4: A ROC curve for a given biometric matching system [8]

II.3 Conclusion

We have mostly discussed the broad background of biometry in this chapter by defining the various biometric modalities and their characteristics. We discussed the structure of a biometric system as well as how to calculate its performance. Then we looked at one of the modalities for identifying persons, the face, and demonstrated why it is one of the most often utilized modalities today, as well as how a facial recognition system is built.

CHAPTER III

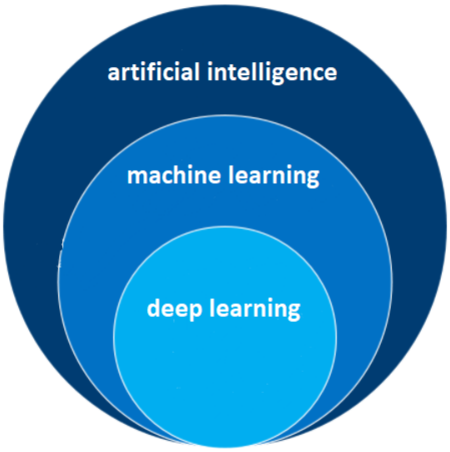
Classifications of deep learning

III.1 Introduction

- **Deep Learning**

Deep learning is a branch of machine learning that is primarily concerned with developing algorithms that enable a computer to learn to perform difficult tasks that require a deep understanding of data and the nature of its work.

It is based mainly on Artificial Neural Networks, which we will learn about later. The following figure shows the relationship of deep learning to machine learning and artificial intelligence



What distinguishes deep learning algorithms in particular is their ability to learn and automate tasks without explicit programming. By explicit programming here we mean writing specific commands and conditional tools to test the data in order to reach a certain result or extracting data features manually by data scientists.

Deep learning algorithms can automatically extract the most important features and recurring patterns of data by looking at a lot of the input data and then analyzing it to find direct or indirect links and relationships between the input data and the required output. This is in contrast to previous machine learning algorithms that required a deep understanding of the data and a great effort to manually elicit its features and patterns by data scientists.

- **Deep Learning Applications**

Deep learning is convenient to use in identification applications such as face recognition, text translation, voice recognition and advanced driver assistance systems.
Some examples of deep learning at work:

_ Fraud Detection: An ATM rejects a counterfeited bank card.
_ Autonomous Cars: A self-driving car capable of slowing down as it approaches a pedestrian crosswalk.
_ Virtual Assistants: A virtual assistant that can understand natural language voice commands and complete tasks for the user.
_ Facial Recognition Systems: A system that can recognize human faces and matching them against a database of faces.

- **What Makes Deep Learning cutting edge?**

Accuracy is what makes deep learning state-of-the-art, advanced tools and techniques drastically enhanced deep learning algorithms to the point where they can outperform humans in specific tasks such as classifying images, beating world champion in chess or enable a virtual assistant like amazon echo or google home to find that dress you like.
Three technology enablers make this degree of accuracy possible:

1- Easy access to huge sets of classified data

Data sets such as ImageNet and PASCAL VoC are available for free, and are useful for training on many different types of objects.

2- Increased computing power

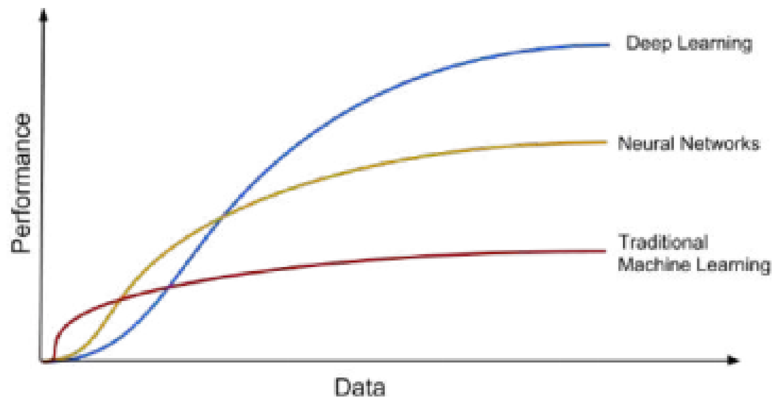
High-performance GPUs accelerate the process of training huge amounts of data needed for deep learning, reducing training time from days to hours.

3-Pretrained models built by experts

Models that can be retrained to perform new recognition tasks using a technique called transfer learning such as AlexNet.

- **Difference between deep learning and machine learning**

The major difference between deep learning and machine learning is its execution as the size of data increases. Deep learning algorithms need a large of data, when the data is small those algorithms don’t perform that well.

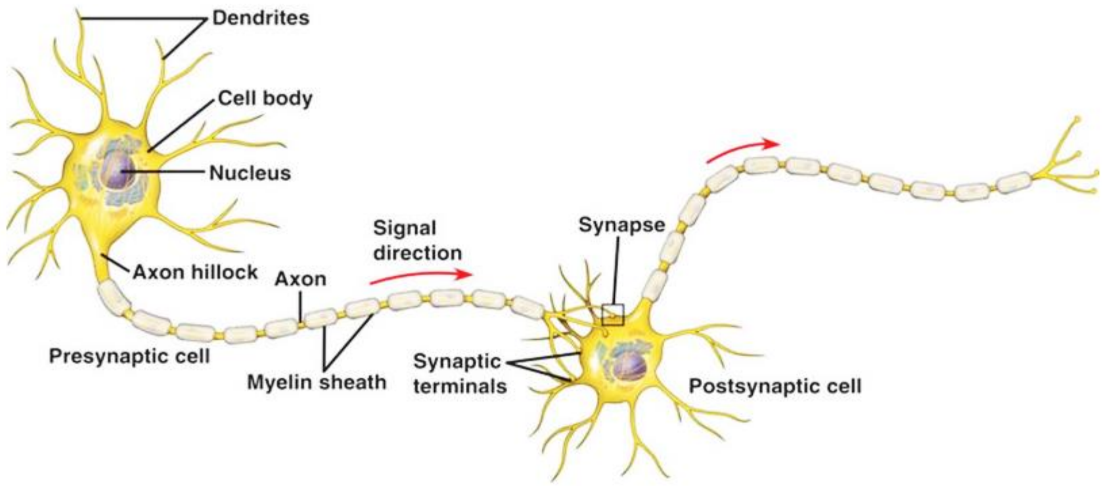


FigureIII.1: Machine learning vs deep learning

III.2 ARTIFICIAL NEURAL NETWORKS:

III.2.1 Human neural network

The principal work of artificial neural networks is similar to the human neural network, it is good to know how it works (Figure 1).



FigureIII.2: A diagram of the neuron highlighting the chain structure between the axon and dendrite. (1.)

Usually, when a neuron is triggered by a stimulus (2) it fires signals to the axon of that neuron and another neuron's dendrites through a synapse. After that, the new neuron might fire and cause another neuron to fire, repeating the process in the whole system.

III.2.2 Artificial neural network

An artificial neural network (ANN) is a variety of layers of neurons. When the ANN is fully connected each unit in a layer is connected to each unit in the next layer (Figure 2).

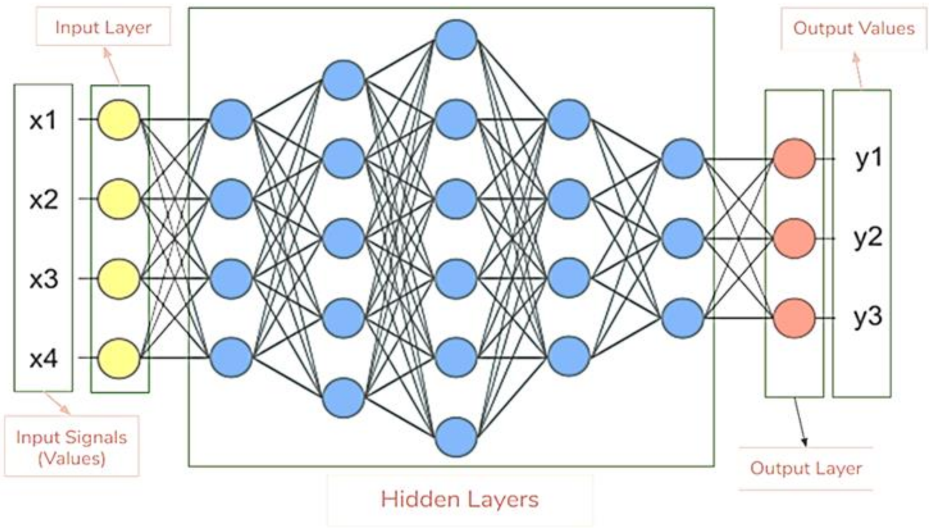


Figure III.3: The artificial neural network architecture (ANN)

To classify images there is an input layer, where the network takes all the information needed. Each hidden layer between the input layer and output layer is used to detect a different set of features in an image, from less to more detailed. For example, the first hidden layer detects edges, the second layer detects forms, and the third layer detects certain image elements, for example, a face or a ball. (6.) The output layer is where the network makes predictions. The predicted image categories are compared to the labels provided by humans. If they are incorrect, the network uses a technique called backpropagation, to correct its learning, so it can make guesses more correctly in the next iteration. (7.) After enough learning, a network can make classifications automatically without human help. (5.)

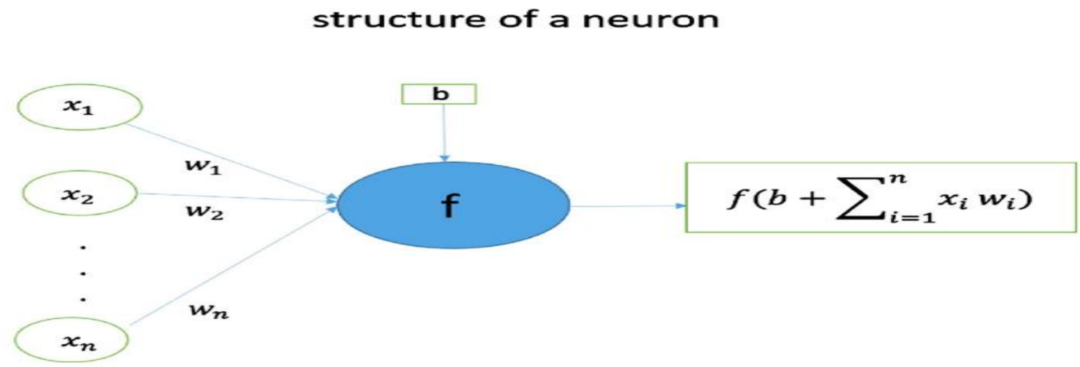
III.2.3 Artificial neuron

An artificial neuron is a connection point (unit or node) in an artificial neural network and can process input signals and produce an output signal. (8.)

III.3 Weights, biases, and activation functions

III.3.1 Weights and structure of a neuron

The connections between the units in a neural network are weighted, which means that the weight indicates the influence of the input from a previous unit that has on the output of the next unit. (9.) To mathematically compute an artificial neuron, all the products of all the inputs (x_1 to x_n) and their corresponding weights (w_1 to w_n) are added, then a bias (b) is added to that sum, then to form the output, the resulting value is poured into an activation function



FigureIII.4: A diagram to show the work of a neuron: input x, weights w, bias b activation function

III.3.2 Biases

A bias (b) is an extra input to a neuron (Figure 3) and it is technically the number 1 multiplied by a weight. (9.) The bias makes it possible to move the activation function curve left or right on the coordinate graph, enabling the neuron to create the required output value (Figure 4). (1 1.)

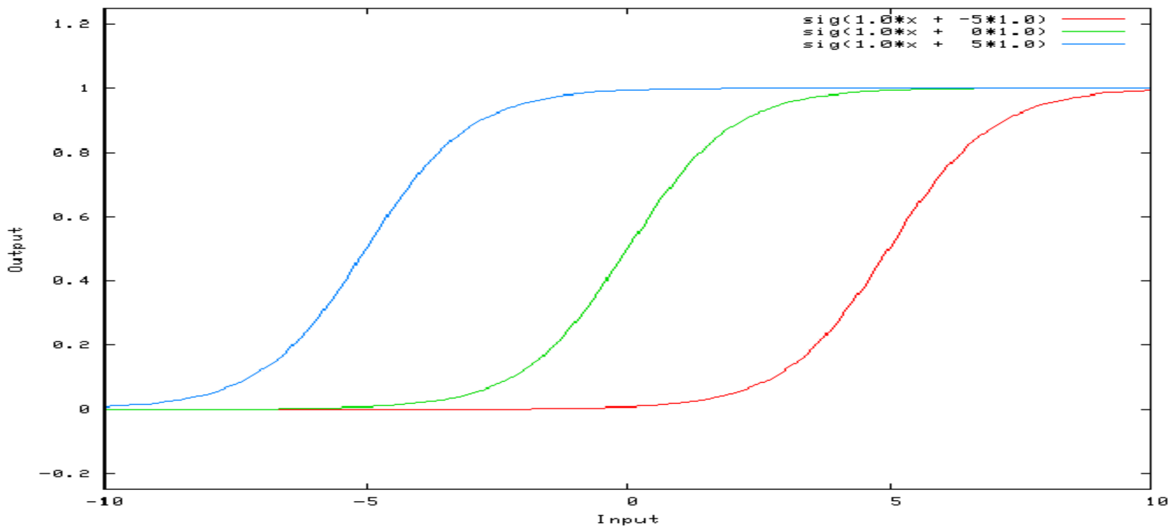
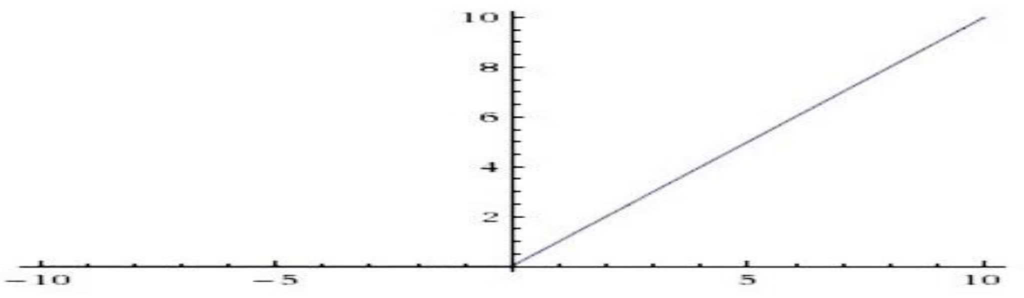


Figure III.5: A bias value allows the activation function to shift to the left or right.

To explain Figure 4, a bias value of −5 enables the Sigmoid activation function to output 0 when the input x is 2.

III.3.3 Activation functions and the ReLu (Rectified Linear unit)

In a neural network, the activation function is responsible for transforming the summed weighted input from the node into the activation of the node or output for that input.



FigureIII.5: The ReLufunction.

III.3.4Backpropagation

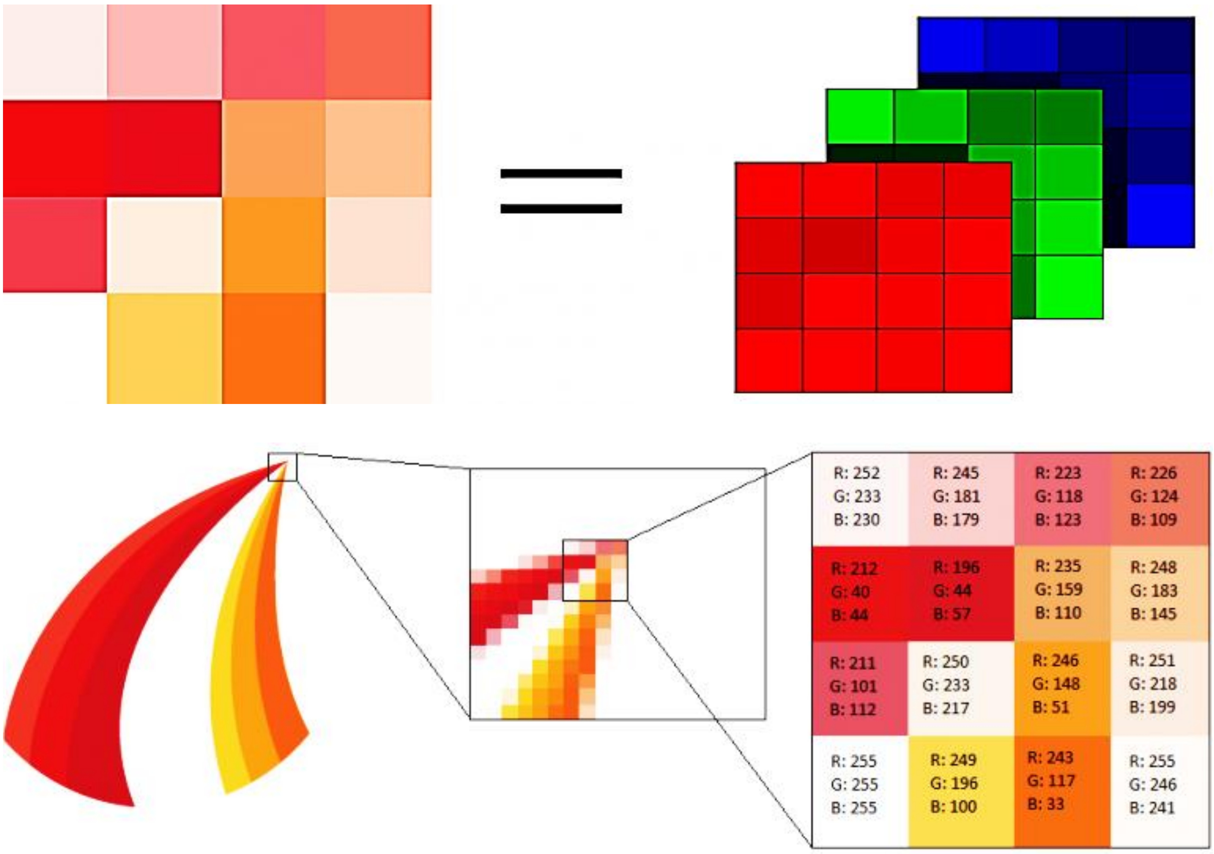
Backpropagation is an important mathematical tool for improving the accuracy of predictions in data mining and machine learning. Essentially, backpropagation is an algorithm used to calculate derivatives quickly.

III.4 CONVOLUTIONAL NEURAL NETWORKS

III.4.1 How computers see images

An image is comprised of pixels. In the RGB model, each pixel has three color elements, red, green, and blue. Most commonly each element can range from 0 (no color) to 255 (full saturation) in value**.**

In digital image processing, a colored image is a three-layered matrix of pixels, where each layer is a two-dimensional matrix representing red, green, or blue pixel values



FigureIII.6: A three-dimensional RGB matrix. Each layer of the matrix is a two-dimensional matrix of red, green, or blue pixel values.

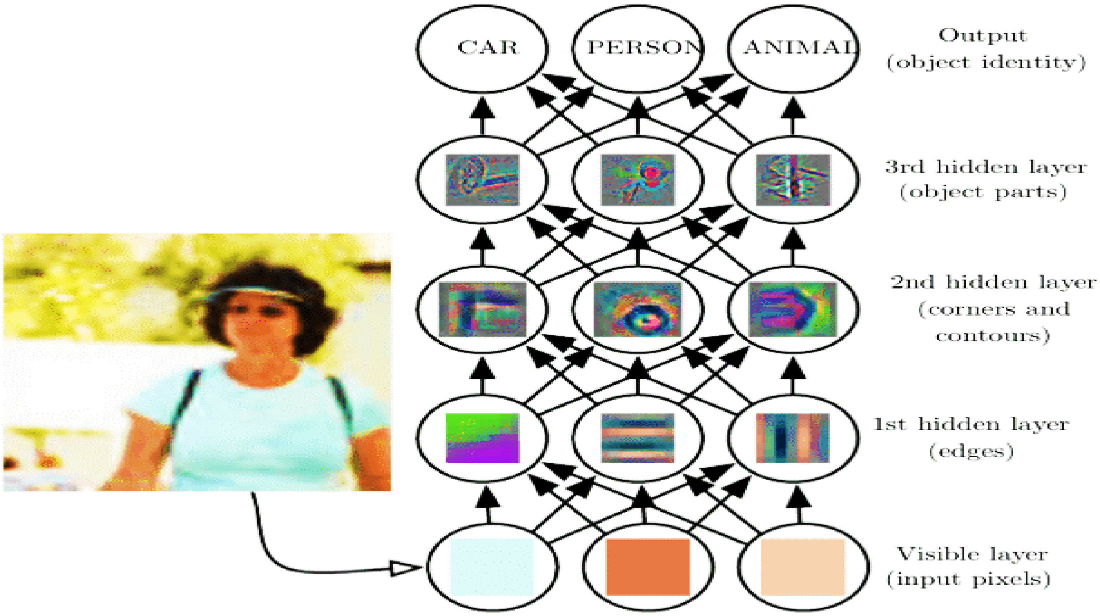
III.4.2 Convolutional Neural Networks

A convolutional neural network, or CNN, is a deep learning neural network designed for processing structured arrays of data such as images. Convolutional neural networks are widely used in computer vision and have become the state of the art for many visual applications such as image classification, and have also found success in natural language processing for text classification.

A convolutional neural network is a feed-forward neural network, often with up to 20 or 30 layers. The power of a convolutional neural network comes from a special kind of layer called the convolutional layer.

Convolutional neural networks contain many convolutional layers stacked on top of each other, each one capable of recognizing more sophisticated shapes. With three or four convolutional layers it is possible to recognize handwritten digits and with 25 layers it is possible to distinguish human faces.

The usage of convolutional layers in a convolutional neural network mirrors the structure of the human visual cortex, where a series of layers process an incoming image and identify progressively more complex features.



FigureIII.7: The CNN architecture.

III.4.3 Architecture

All CNN models found are the same architecture (Figure III.8).

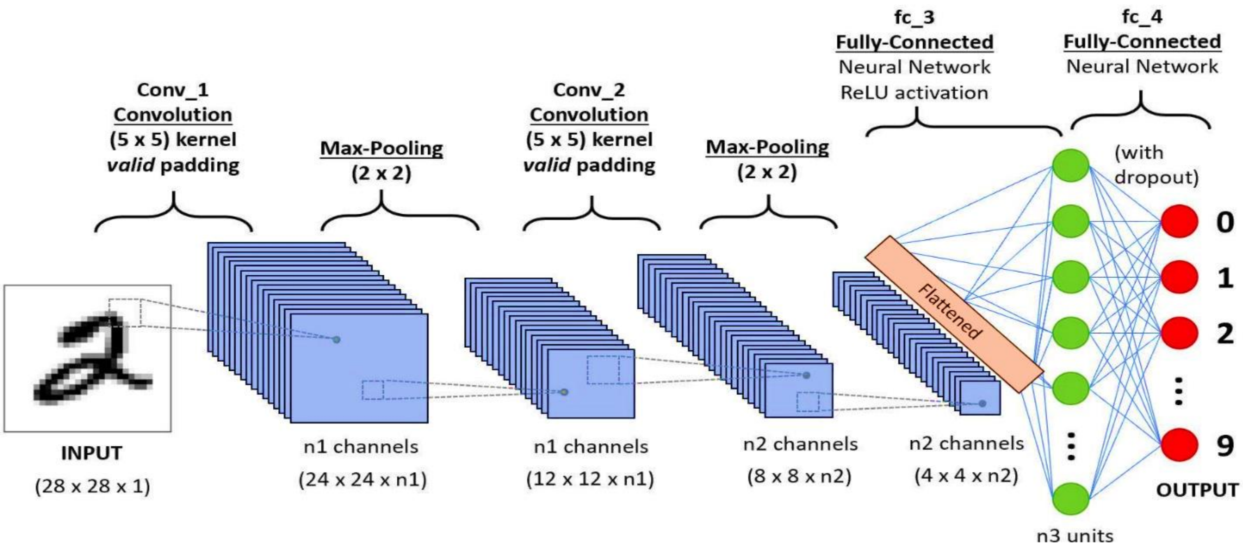
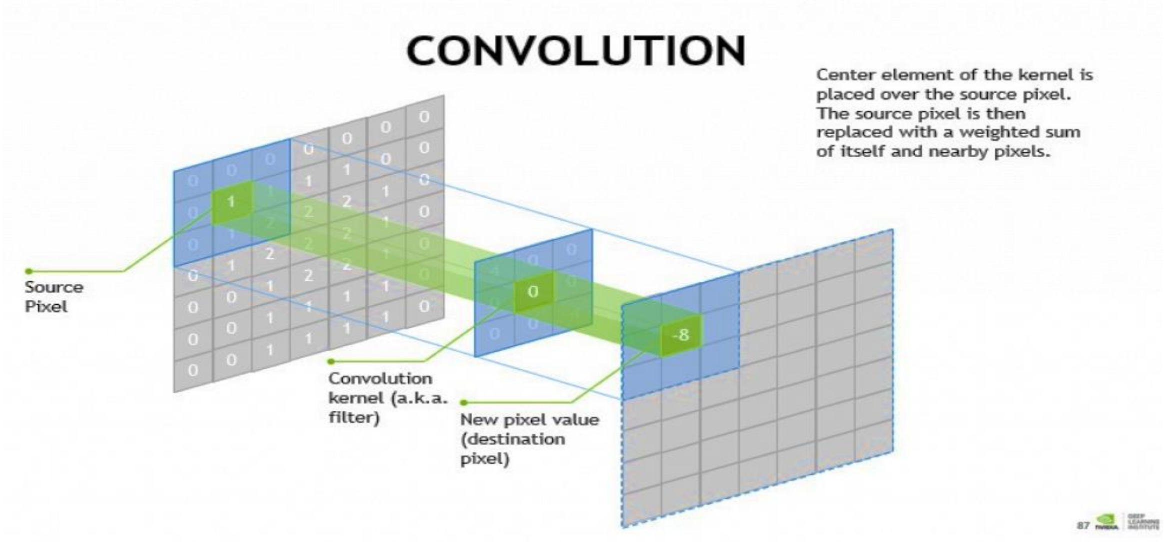


Figure III.8: The CNN architecture

III.5Convolutional layers

The term “convolution” refers to the mathematical combination of two functions to form a third function. When that happens, two sets of information are merged. In the context of CNNs, a convolutional layer (called filter or kernel) is applied to the input data to then produce a feature map



FigureIII.9: The filter slides over the input and performs its output on the new layer.

In Figure III.9, a dot product multiplication is made between a 3x3 sized filter matrix and a 3x3 sized area of the input image’s matrix. The elements of the resulting matrix are summed and the sum is the output value (“destination 16 pixels”) on the feature map. The filter then slides over the input matrix, repeats the dot product multiplication with every remaining combination of 3x3 sized areas, and completes the feature map.

III.5.1Pooling layers

Pooling layers are responsible for reducing the dimensionality of feature maps, specifically the height and width, preserving the depth. (21.) Doing so is beneficial because it decreases the required computational power to process the data while extracting the dominant features in feature maps. (25.) There are two types of pooling layers: max pooling and average pooling (Figure III.10).

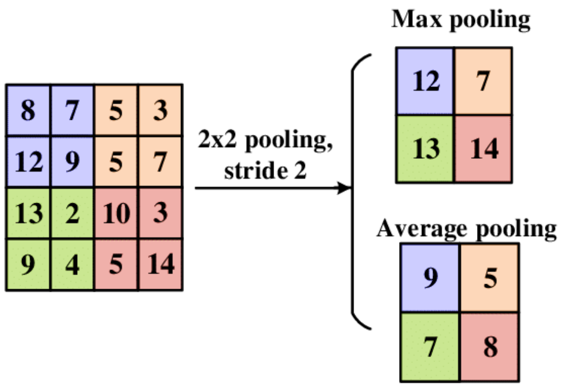


Figure III.10: Types of pooling.

The Max pooling outputs the maximum value of the elements in the portion of the image covered by the filter, while average pooling returns the average value. Max pooling is better at extracting dominant features and is therefore considered more performant.

III.5.2Fully connected layers

Fully connected layers are where classification happens. The input matrix is flattened into a column vector and is fed into a set of fully connected layers which are the same as the fully connected ANN architecture that was previously discussed in chapter 2.2. (21.) Each fully connected layer (called the Dense layer) is passed through an activation function (e.g., Tanh or ReLu), but the output Dense layer is passed through SoftMax. In the SoftMax multiclass classification, the loss function used is Cross-Entropy (*categoricalcrossentropy*inKeras). (26.) The output of the SoftMax function is an N-dimensional vector, where N is the number of classes the CNN has to choose from. Each number in this N-dimensional vector *represents* the probability that the image belongs to each certain class. For example, if the output vector is [0 .1 .1 .75 0 0 0 0 0 .05], then there is a 10% probability that this image belongs to class 2, 10% probability that it belongs to class 3, 75% probability that this image belongs to the class 4, and 5% probability that it belongs to the class 10. (27.)

III.5.3GoogLeNet/Inception (2014)

GoogLeNet is developed based on the idea that several connections between layers are ineffective and have redundant information due to the correlation between them. Accordingly, it uses an “Inception module”, a sparse CNN, with 22 layers in a parallel processing workflow, and benefits from several auxiliary classifiers within the intermediate layers to improve the discrimination capacity in the lower layers. In contrast to conventional CNNs such as AlexNet and VGG, wherein either a convolutional or a pooling operation can be used at each level, the Inception module could benefit from both at each layer. Furthermore, filters (convolutions) with varying sizes are used at the same layer, providing more detailed information and extracting patterns with different sizes. Importantly, a 1 x 1 convolutional layer, the so-called bottleneck layer, was employed to decrease both the computational complexity and the number of parameters. To be more precise, 1 x 1 convolutional layers were used just before a larger kernel convolutional filter (e.g., 3 x 3 and 5 x 5 convolutional layers) to decrease the number of parameters to be determined at each level (i.e., the pooling feature process). In addition, 1 x 1 convolutional layers make the network deeper and add more non-linearity by using ReLU after each 1 x 1 convolutional layer. In this network, the fully connected layers are replaced with an average pooling layer. This significantly decreases the number of parameters since the fully connected layers include a large number of parameters. Thus, this network is able to learn deeper representations of features with fewer parameters relative to AlexNet while it is much faster than VGG.[38]

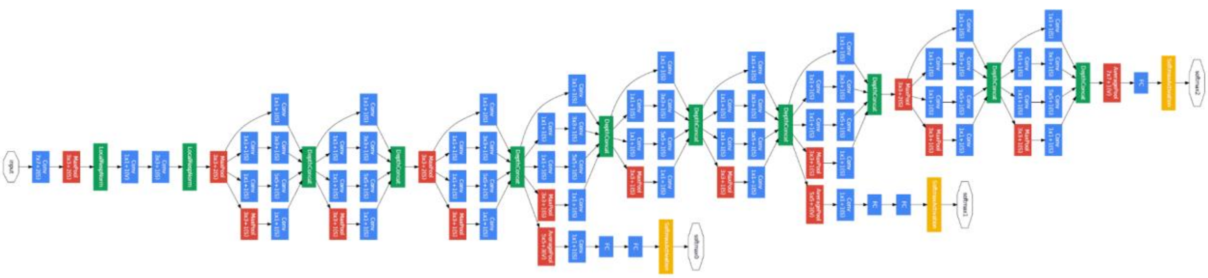


Figure III.11: Compressed view of the Architecture of GoogLeNet (version 3) [38]

III.5.4ResNet (2015)

Residual Neural Network (ResNet) by Kaiming He et al introduces an architecture which consists of 152 layers with skip connections (gated units or gated recurrent units) and features heavy batch normalization. The whole idea of ResNet is to counter the problem of vanishing gradients. By preserving the gradients, Vanishing gradients is the problem that occurs in networks with high number of layers as the weights of the first layers cannot be updated correctly through the backpropagation of the error gradient (the chain rule multiplies error gradient values lower than one and then, when the gradient error comes to the first layers, its value goes to zero).[35] The deep ResNet configuration addresses the vanishing gradient problem by employing a deep residual learning module via additive identity transformations. Specifically, the residual module uses a direct path between the input and output and each stacked layer fits a residual mapping rather than directly fitting a desired underlying mapping. [38]

CHAPTER IV
Experimentation

IV.1Introduction:

The implementation of aimage application with convolution neural networks is discussed in this chapter. GoogLeNet and AlexNet are the two primary libraries utilized in the implementation.

The program will be able to recognize the faces of the subjects on which it has been trained. As input, the program will get a picture including a human and animals (cats & dogs).

It will recognize that person's face and send it to our CNN expert.

The program will return the image of the subject if the model is used to identify the subject. Individual bearing the expected topic name and likelihood value of being accurate

IV.2 Hardware and software used

The software used was MATLAB. This is a programming environment used predominantly in Computer Vision research work. Having prebuilt functions and programming languages for the simulation of a neural network, it leads to faster completion of the setting up the environment stage.

The hardware used is laptop has:

- CPU intel i5 2.5Ghz
- 8Gb RAM
- Storage 120Gb SSD.
- OS: Windows 10 Pro

The deep learning framework used is GoogLeNetAlexNet Resnet.

IV.3 Some definitions

IV.3.1 Train, validation and test

The train dataset is used to train the model with. In the case of neural networks, the model learns its weights and biases.

The validation dataset is what the model uses for evaluation after every set of predictions. It helps the model tune its hyperparameters.

The test dataset is used to evaluate the model after it has been completely trained. (28.)

IV.3.2 Overfitting and underfitting

Overfitting occurs when the model captures the noise of the data. Intuitively, it fits the data too well, or in other words it is too dependent on the data used for training.
Conversely, underfitting occurs when the model cannot capture the underlying trend of the data, or intuitively it does not fit the data well enough.
Overfitting and underfitting both result in poor predictions in new datasets. (29.)

IV.3.3Batch size

Most of the time the whole dataset cannot be fed into the neural network at once, so it has to be divided into parts, or batches. The batch size indicates the number of training samples in a single batch. (30.)

IV.3.4 Epoch

One epoch is when the entire dataset (i.e., every training sample) is fed forward and backward through the neural network only once. (30.)

IV.3.5 Dropout

Dropout is a technique used to reduce overfitting. The term dropout refers to randomly dropping out units and their connections during training. (31, abstract.)

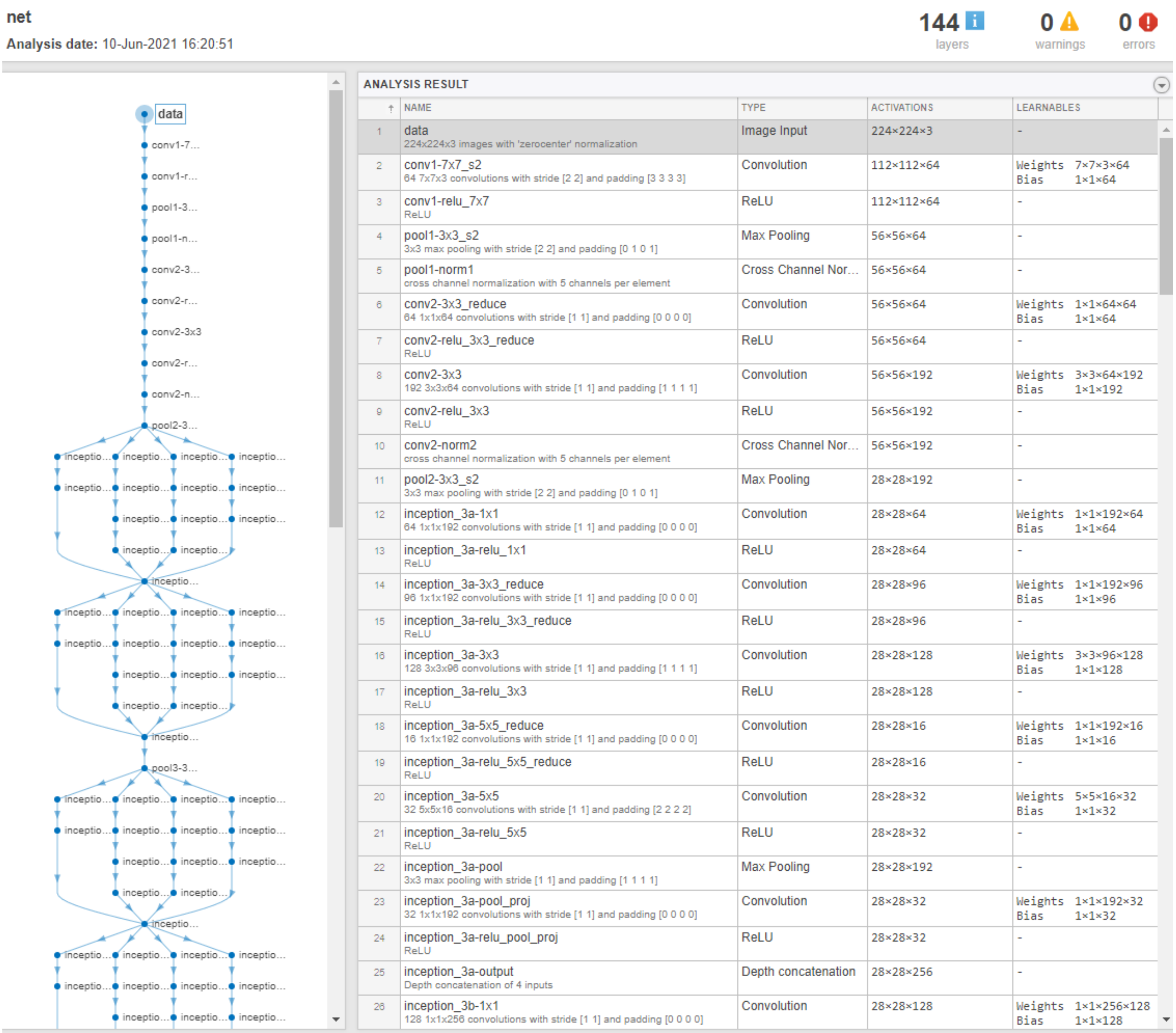
IV.3.6 Batch normalization

Batch normalization is also a method for reducing overfitting. It normalizes the input layer by adjusting and scaling the activations. (32.) The mathematics behind batch normalization is out of the scope of this thesis.

IV.4 Models and results

IV.4.1 First model

This model is based on the Convolutional Neural Network (CNN) tutorial of GoogLeNet.



	NAME	TYPE	ACTIVATIONS	LEARNABLES
1	data 224x224x3 images with 'zerocenter' normalization	Image Input	224×224×3	-
2	conv1-7x7_s2 64 7x7x3 convolutions with stride [2 2] and padding [3 3 3 3]	Convolution	112×112×64	Weights 7×7×3×64 Bias 1×1×64
3	conv1-relu_7x7 ReLU	ReLU	112×112×64	-
4	pool1-3x3_s2 3x3 max pooling with stride [2 2] and padding [0 1 0 1]	Max Pooling	56×56×64	-
5	pool1-norm1 cross channel normalization with 5 channels per element	Cross Channel Nor...	56×56×64	-
6	conv2-3x3_reduce 64 1x1x64 convolutions with stride [1 1] and padding [0 0 0 0]	Convolution	56×56×64	Weights 1×1×64×64 Bias 1×1×64
7	conv2-relu_3x3_reduce ReLU	ReLU	56×56×64	-
8	conv2-3x3 192 3x3x64 convolutions with stride [1 1] and padding [1 1 1 1]	Convolution	56×56×192	Weights 3×3×64×192 Bias 1×1×192
9	conv2-relu_3x3 ReLU	ReLU	56×56×192	-
10	conv2-norm2 cross channel normalization with 5 channels per element	Cross Channel Nor...	56×56×192	-
11	pool2-3x3_s2 3x3 max pooling with stride [2 2] and padding [0 1 0 1]	Max Pooling	28×28×192	-
12	inception_3a-1x1 64 1x1x192 convolutions with stride [1 1] and padding [0 0 0 0]	Convolution	28×28×64	Weights 1×1×192×64 Bias 1×1×64
13	inception_3a-relu_1x1 ReLU	ReLU	28×28×64	-
14	inception_3a-3x3_reduce 96 1x1x192 convolutions with stride [1 1] and padding [0 0 0 0]	Convolution	28×28×96	Weights 1×1×192×96 Bias 1×1×96
15	inception_3a-relu_3x3_reduce ReLU	ReLU	28×28×96	-
16	inception_3a-3x3 128 3x3x96 convolutions with stride [1 1] and padding [1 1 1 1]	Convolution	28×28×128	Weights 3×3×96×128 Bias 1×1×128
17	inception_3a-relu_3x3 ReLU	ReLU	28×28×128	-
18	inception_3a-5x5_reduce 16 1x1x192 convolutions with stride [1 1] and padding [0 0 0 0]	Convolution	28×28×16	Weights 1×1×192×16 Bias 1×1×16
19	inception_3a-relu_5x5_reduce ReLU	ReLU	28×28×16	-
20	inception_3a-5x5 32 5x5x16 convolutions with stride [1 1] and padding [2 2 2 2]	Convolution	28×28×32	Weights 5×5×16×32 Bias 1×1×32
21	inception_3a-relu_5x5 ReLU	ReLU	28×28×32	-
22	inception_3a-pool 3x3 max pooling with stride [1 1] and padding [1 1 1 1]	Max Pooling	28×28×192	-
23	inception_3a-pool_proj 32 1x1x192 convolutions with stride [1 1] and padding [0 0 0 0]	Convolution	28×28×32	Weights 1×1×192×32 Bias 1×1×32
24	inception_3a-relu_pool_proj ReLU	ReLU	28×28×32	-
25	inception_3a-output Depth concatenation of 4 inputs	Depth concatenation	28×28×256	-
26	inception_3b-1x1 128 1x1x256 convolutions with stride [1 1] and padding [0 0 0 0]	Convolution	28×28×128	Weights 1×1×256×128 Bias 1×1×128

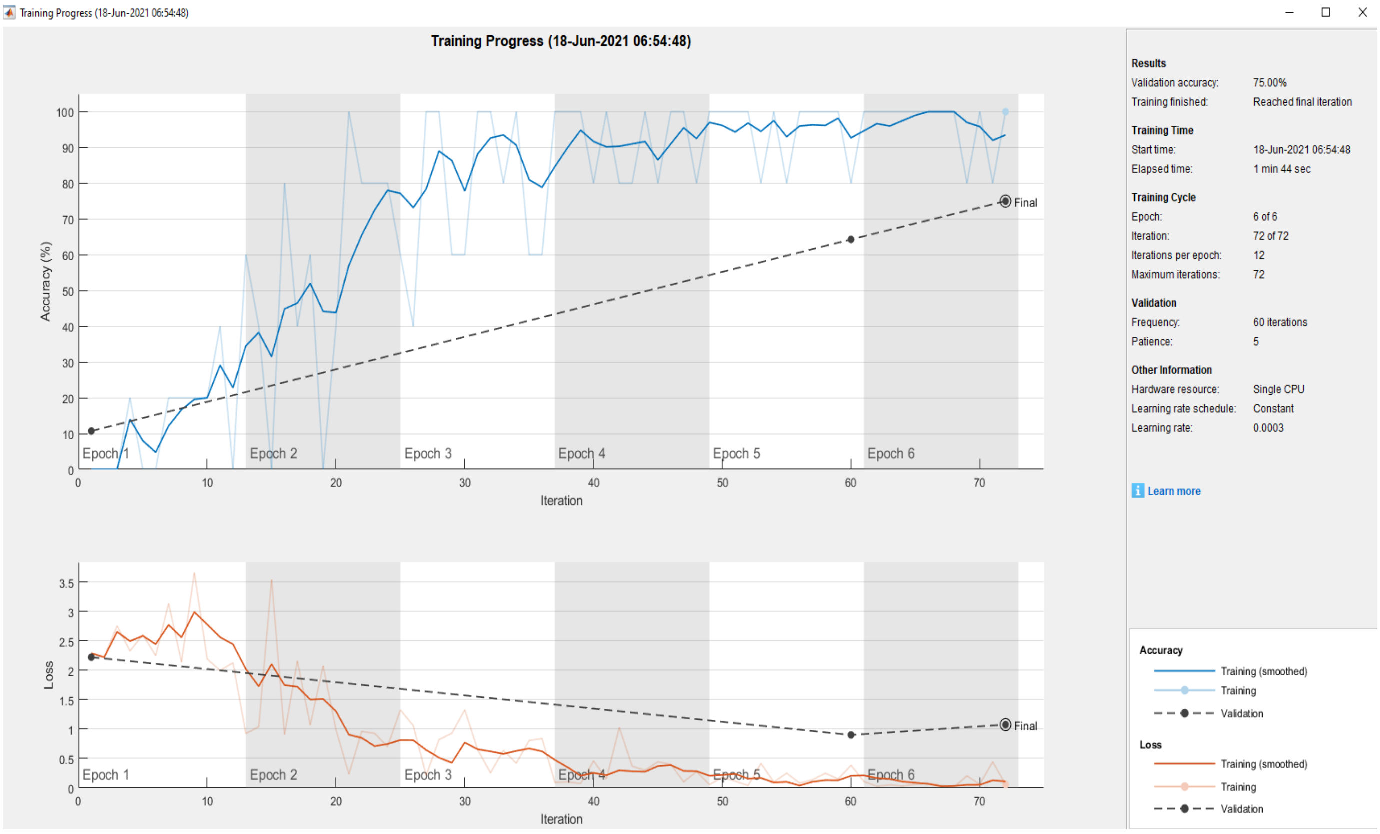
Here is the MATLAB code:

```
% You must download deep learningToolbox &Deep Learning Network Analyzer for Neural Network Toolbox befortuning This program
clc;
close all;
clearall;
Dataset = imageDatastore('My_Dataset','IncludeSubfolders',true,'LabelSource','foldernames');
[Training_Dataset,Validation_Dataset] = splitEachLabel(Dataset,0.7);

net = googlenet;
analyzeNetwork(net);

Input_Layer_Size = net.Layers(1).InputSize;

Layer_Graph = layerGraph(net);

Feature_Learner = net.Layers(142);
Output_Classifier = net.Layers(144);

Number_of_Classes = numel(categories(Training_Dataset.Labels));

New_Feature_Learner = fullyConnectedLayer(Number_of_Classes,...
```

```
'Name','FacialFeatureLearner', ...
'WeightLearnRateFactor',10, ...
'BiasLearnRateFactor',10);

New_Classifier_Layer = classificationLayer('Name', 'Face Classifier');

Layer_Graph = replaceLayer(Layer_Graph,Feature_Learner.Name,New_Feature_Learner);
Layer_Graph = replaceLayer(Layer_Graph, Output_Classifier.Name, New_Classifier_Layer);
analyzeNetwork(Layer_Graph);

Pixel_Range = [-30 30];
Scale_Range = [0.9 1.1];

Image_Augmenter = imageDataAugmenter(...
'RandXReflection', true,...
'RandXTranslation', Pixel_Range,...
'RandYTranslation', Pixel_Range,...
'RandXScale', Scale_Range,...
'RandYScale', Scale_Range);
Augmented_Training_Image = augmentedImageDatastore(Input_Layer_Size(1:2),Training_Dataset, ...
'DataAugmentation',Image_Augmenter);
% disp(Augmented_Training_Image);
Augmented_Validation_Image = augmentedImageDatastore(Input_Layer_Size(1:2),Validation_Dataset);

Size_of_Minibatch =5;

Training_Options = trainingOptions('sgdm',...
'MiniBatchSize', Size_of_Minibatch,...
'MaxEpochs', 6,...
'InitialLearnRate', 3e-4,...
'Shuffle', 'every-epoch',...
'ValidationData', Augmented_Validation_Image,...
'ValidationFrequency',50, ...
'Verbose', false,...
'Plots', 'training-progress');

net = trainNetwork(Augmented_Training_Image,Layer_Graph,Training_Options);
%% testing program
```

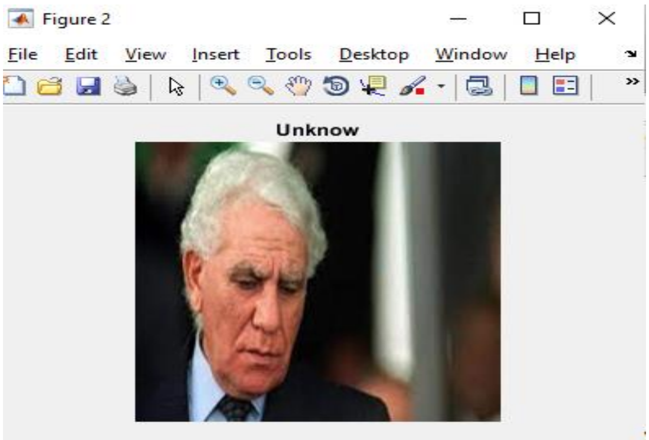
Now we create a function to test network code of function named test_networkbelow:

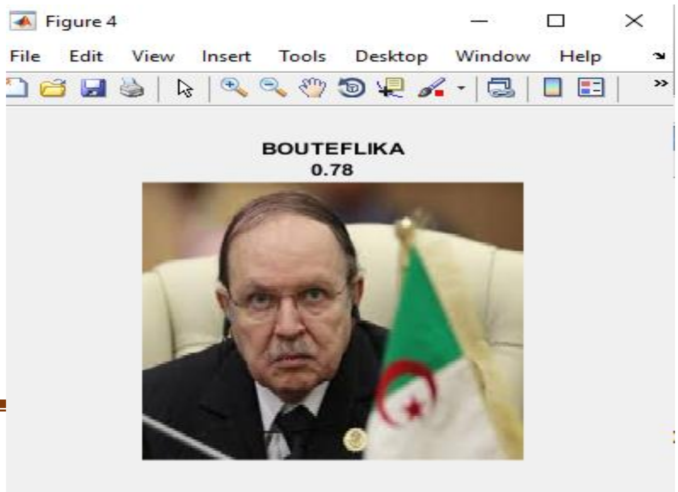
```
        functiontest_network(net, image)

I = imread(image);
G = imresize(I,[224 224]);
[Label,Prob] = classify(net,G);

figure;
imshow(G);
ifmax(Prob)<0.6
title(sprintf('Unknow'));
else
title({char(Label) num2str(max(Prob),2)});
end
```



The result of training progress accuracy is 75%

Testing this code by test_network function the results are





IV.4.2 Second model

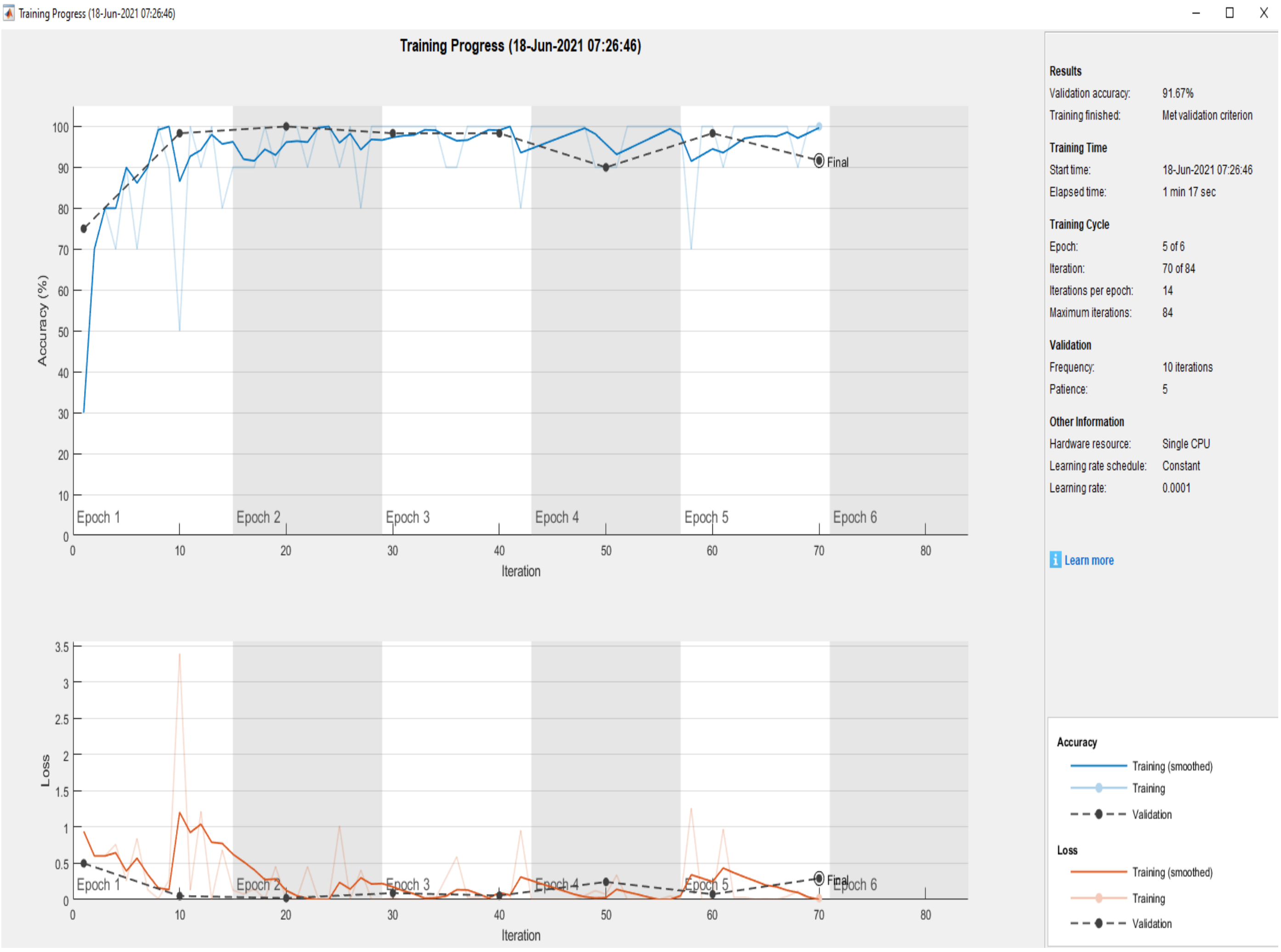
AlexNet is a classic convolutional neural network architecture. It consists of convolutions, max pooling and dense layers as the basic building blocks. Grouped convolutions are used in order to fit the model across two GPUs.we use animal dataset (Cat & Dog) to training

The code of this model is:

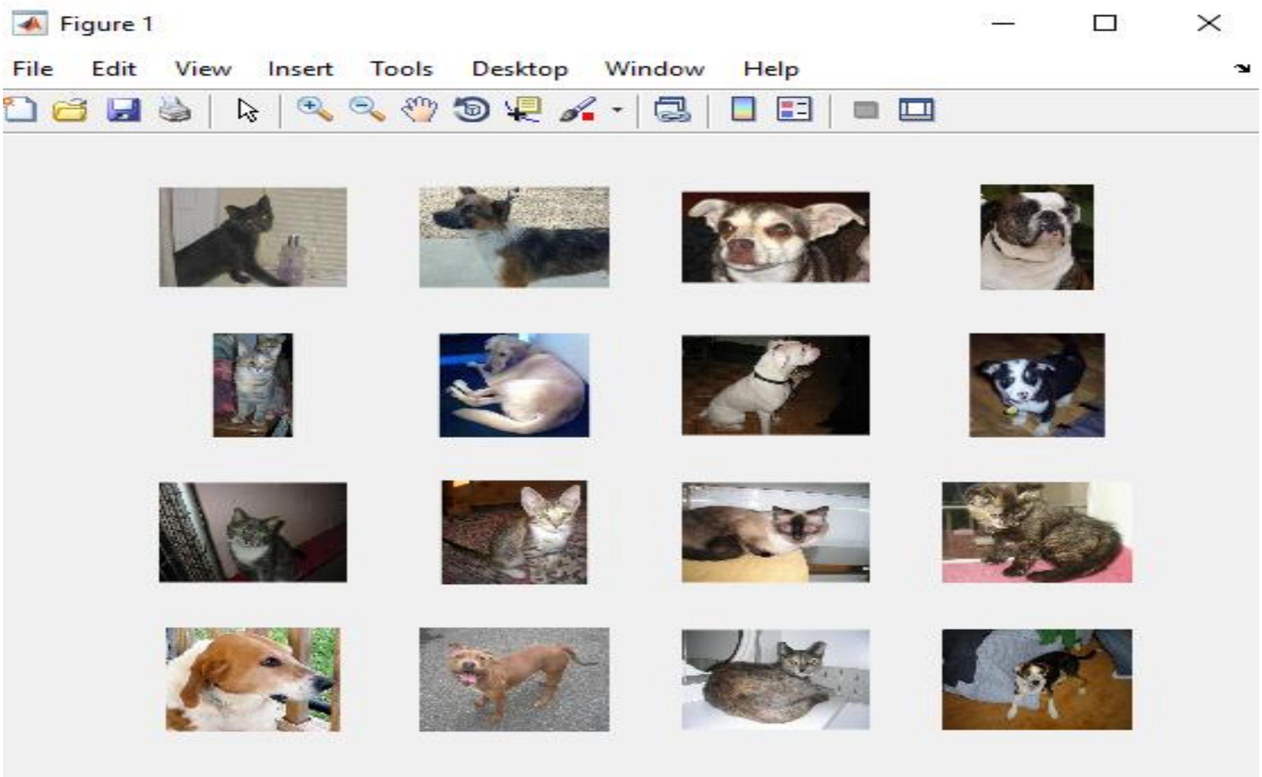
```
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear; close; clc;
imds = imageDatastore('DataStore_Image', ...
'IncludeSubfolders',true, ...
'LabelSource','foldernames');
[imdsTrain,imdsValidation] = splitEachLabel(imds,0.7,'randomized');
visualize = 1;
ifvisualize==1
numTrainImages = numel(imdsTrain.Labels);
idx = randperm(numTrainImages,16);
    fig1 = figure;
fori = 1:16
subplot(4,4,i)
        I = readimage(imdsTrain,idx(i));
imshow(I)
end
    print(fig1,strcat('DataStore_Image','input-data-selected'),'-djpeg')
end


%%
net = alexnet;
inputSize = net.Layers(1).InputSize ;
%%
inspect_network=0;
ifinspect_network==1
analyzeNetwork(net)
end
%%
layersTransfer = net.Layers(1:end-3);

numClasses = numel(categories(imdsTrain.Labels)) ;%the number of classes in the new data

layers = [
layersTransfer
    fullyConnectedLayer(numClasses,'WeightLearnRateFactor',20,'BiasLearnRateFactor',20)
softmaxLayer
classificationLayer];
%%
pixelRange = [-30 30];
imageAugmenter = imageDataAugmenter( ...
'RandXReflection',true, ... %randomly flip the training images along the vertical axis
'RandXTranslation',pixelRange, ... %randomly translate them up to 30 pixels horizontally and vertically
'RandYTranslation',pixelRange);

augimdsTrain = augmentedImageDatastore(inputSize(1:2),imdsTrain, ...
'DataAugmentation',imageAugmenter);
```

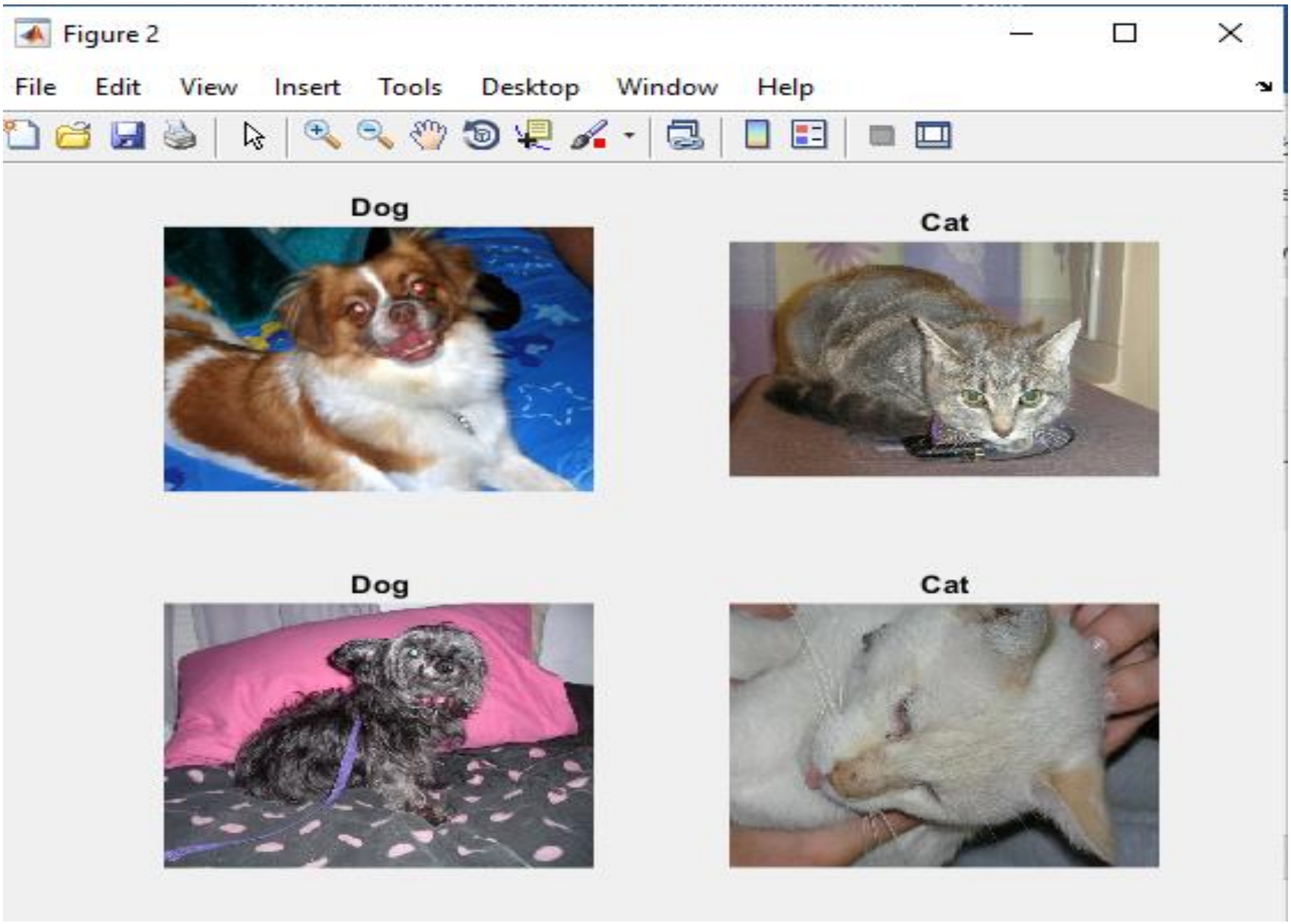
```
% automaticallyresize the validation images withoutperformingfurther data augmentation
augimdsValidation = augmentedImageDatastore(inputSize(1:2),imdsValidation);

%%
options = trainingOptions('sgdm', ...
'MiniBatchSize',10, ... %whenperformingtransferlearning, you do not need to train for as manyepochs
'MaxEpochs',6, ...
'InitialLearnRate',1e-4, ... %slow down learning in the transferredlayers( fastlearningonly in the new layers and slowerlearning in the otherlayers)
'Shuffle','every-epoch', ...
'ValidationData',augimdsValidation, ...
'ValidationFrequency',10, ...
'Verbose',false, ...
'Plots','training-progress', ...
'ExecutionEnvironment','auto'); %Hardware resource for training network - auto, cpu, gpu, multi-gpu, parallel

%%
netTransfer = trainNetwork(augimdsTrain,layers,options); %By default, trainNetwork uses a GPU if one isavailable

%%

[YPred,scores] = classify(netTransfer,augimdsValidation); %classifyusing the fine-tuned network
classify_visualize = 1;
ifclassify_visualize==1
idx = randperm(numel(imdsValidation.Files),4);
fig = figure;
fori = 1:4
subplot(2,2,i)
        I = readimage(imdsValidation,idx(i));
imshow(I)
        label = YPred(idx(i));
title(string(label));
end
%print(fig,strcat(filename,'classification-result'),'-djpeg')
end

%% end Program
```

The training progress is:



The result of test like below





General Conclusion

The quality of picture identification and object detection has improved considerably in recent years, owing mostly to improvements in deep learning, especially convolution networks. One piece of good news is that much of this growth is the product of new ideas, algorithms, and smarter network designs, rather than more powerful technology, more data sets, or larger models. This inspires and pushes us to continue on this path of deep study in order to overcome obstacles. The goal of this dissertation is to create a facial recognition program that uses a convolutional neural network to recognize faces. Most face recognition systems include several phases (feature extraction, classification, etc.) and may need a preprocessing step, making the recognition system more complex and lengthening the learning time. For these reasons, we used convolutional neuron networks, which integrate the extraction and classification processes.

The thesis was split into two halves (State of the art and Experimentations). The first section covered biometry, face recognition systems, network artificial neurons fundamentals, and the CNN principle. In the third section, we covered the procedures and tools required to complete our project, as well as the constraints of the model under consideration, as well as a walkthrough of the construction of our face recognition system and the results received from our CNN implementation. Although our CNN performed well in terms of prediction accuracy, it has a significant execution time, which is a drawback for CNNs

Bibliography

[1] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2016, http: //www.deeplearningbook.org.

[2] H. Bodlaender, J. Gilbert, H. Hafsteinsson, and T. Kloks, “Approximating treewidth, pathwidth, frontsize, and shortest elimination tree”, Journal of Algorithms, vol. 18, no. 2, pp. 238–255, 1995, ISSN: 0196-6774. DOI: https://doi.org/10.1006/ jagm.1995.1009. [Online]. Available: http://www.sciencedirect.com/science/ article/pii/S0196677485710097.

[3] S. Skansi, Introduction to Deep Learning. Springer, Cham, 2018.

[4] D. P. Kingma and M. Welling, “Auto-Encoding Variational Bayes”, ArXiv e-prints, Dec. 2013. arXiv: 1312.6114 [stat.ML].

28. Shah, T. 2017. About Train, Validation and Test Sets in Machine Learning. Date of retrieval 6.1.2020. Available:

https://towardsdatascience.com/train-validation-and-test-sets-72cb40cba9e7

29. Cai, E. 2014. Machine Learning Lesson of the Day – Overfitting and Underfitting. Date of retrieval 6.1.2020. Available:

https://chemicalstatistician.wordpress.com/2014/03/19/machine-learning-lesson-of-the-day-overfitting-and-underfitting/

30. Sharma, S. 2017. Epoch vs Batch Size vs Iterations. Date of retrieval 6.1.2020. Available:

https://towardsdatascience.com/epoch-vs-iterations-vs-batch-size-4dfb9c7ce9c9

31. Srivastava, N. et al. 2014. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. Cited 6.1.2020. Available:

http://jmlr.org/papers/v15/srivastava14a.html

32. D., F. 2017. Batch normalization in Neural Networks. Cited 6.1.2020. Available:

https://towardsdatascience.com/batch-normalization-in-neural-networks-1ac91516821c

33. Convolutional Neural Network (CNN). Date of retrieval 6.1.2020. Available:

https://www.tensorflow.org/tutorials/images/cnn

34. Keras team, 2019. Train a simple deep CNN on the CIFAR10 small images dataset. Date of retrieval 6.1.2020. Available:

https://github.com/keras-team/keras/blob/master/examples/cifar10_cnn.py

[1] *ImageNet*. http://www.image-net.org

[2] Zhou, Bolei, AdityaKhosla, Agata Lapedriza, Antonio Torralba, and Aude Oliva. "Places: An image database for deepsceneunderstanding." *arXivpreprint arXiv:1610.02055* (2016).

[3] *Places*. http://places2.csail.mit.edu/

[4] Szegedy, Christian, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, DragomirAnguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. "Goingdeeperwith convolutions." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1-9. 2015.

[5] *BVLC GoogLeNet Model*. https://github.com/BVLC/caffe/tree/master/models/bvlc_googlenet

…..