DEMOCRATIC AND PEOPLE'S REPUBLIC OF ALGERIA

MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

UNIVERSITY Dr. TAHAR MOULAY SAIDA

FACULTY: TECHNOLOGY

DEPARTMENT: COMPUTER SCIENCE

MASTER THESIS

Option: Computer science Modeling of Knowledge and Reasoning

The use of surveillance cameras and artificial intelligence (AI) for the post COVID-19 vaccine

Presented by:

Bentadj Cheimaa

Framed by:

Dr. BOUARARA Hadj Ahmed

June 2020-2021

Abstract

The idea was to prove this claim: artificial intelligence is a viable solution to building a smart system to minimize the spread of COVID-19. For this we have proposed a solution using different deep learning tools to deal with different problems related to post COVID-19 spots. This solution is made up of several parts: 1) data will be received by surveillance cameras, sensors and drones. 2) the YOLOV3 with transfer learning and configuration change were used to detect foreground objects in videos and ignore static objects. 3) Detect whether a person is wearing a mask or not through the use of data augmentation and adaptation of MobileNetV2, VGG-16, and AlexNet. 4) calculate the distance between people circulating in public or private places using MobileNet-SSD and the Euclidean distance measure. After evaluating our solutions in different contexts and on different benchmark datasets, the results obtained represent an empirical validation of the benefit derived from the use of deep learning, the internet of things and computer vision to minimize the spread of COVID-19. For this we proposed a solution called I3S-Covid19 (Intelligence system for safer society in covid-19).

Keywords: YOLOV3, AlexNet, data augmentation, transfer learning, object detection, object tracking.

Résumé

L'idée était de prouvé cette allégation : l'intelligence artificiel représente une solution viable pour batir un système intelligent afin de minimiser la propagation du COVID-19. Pour cela nous avons proposé une solution utilisant différents outils du deep learning pour traiter différents problèmes liés aux taches du post COVID-19. Cette solution est composée de plusieurs parties : 1) les données seront reçus par les caméras de surveillance, les capteurs et les drones. 2) le YOLOV3 avec transfert learning et changement de configuration ont été utilisé pour la détection des objets du premier plan dans les videos et ignorer les objets statiques. 3) Détecter si une personne porte un masque ou non à travers l'utilisation du data augmentation et l'adaptation de MobileNetV2, VGG-16, et AlexNet. 4) calculer la distance entre les personnes circulant dans les lieux publiques ou privés utilisant Mobilenet-SSD et la mesure de distance euclidienne. Après l'évaluation de nos solutions dans différents contextes et sur différents ensembles de données benchmarks, les

résultats obtenus représentent une validation empirique de l'avantage tiré de l'utilisation du deep learning, l'internet des objets et la vision par ordinateur pour minimiser la propagation du COVID-19. Pour cela nous avons proposé une solution appelé I3S-covid19 (Intelligence system for safer society in COVID-19).

Mots clés: YOLOV3, AlexNet, augmentation de données, apprentissage par transfert, détection d'objets, suivi d'objets.

الملخص

كانت الفكرة هي إثبات هذا الادعاء: الذكاء الاصطناعي هو حل قابل للتطبيق لبناء نظام ذكي لتقليل انتشار كوفيد-١٩. لهذا ، اقترحنا حلاً باستخدام أدوات التعلم العميق المختلفة للتعامل مع المشكلات المختلفة المتعلقة بمواقع ما بعد كوفيد-١٩. يتكون هذا الحل من عدة أجزاء: ١). سيتم استلام بيانات بواسطة كاميرات المراقبة وأجهزة الاستشعار والطائرات بدون طيار. ٢). تم استخدام YOLOV3 مع نقل التعلم وتغيير التكوين لاكتشاف الكائنات الأمامية في مقاطع الفيديو وتجاهل الكائنات الثابتة. ٣).اكتشف ما إذا كان الشخص يرتدي قناعاًا أم لا من خلال استخدام زيادة البيانات وتكييف MobilenetV2 و VGG-16 و AlexNet. ٤). يحسب المسافة بين الأشخاص الذين يتجولون في الأماكن العامة أو الخاصة باستخدام MobileNet-SSD و YOLOV3 و مقياس المسافة الإقليدية. بعد تقييم حلولنا في سياقات مختلفة وعلى مجموعات بيانات معيارية مختلفة ، تمثل النتائج التي تم الحصول عليها إثباتاًا تجريبياًا للفائدة المستمدة من استخدام التعلم العميق وإنترنت الأشياء ورؤية الكمبيوتر لتقليل انتشار كوفيد-١٩. لهذا اقترحنا حلاً يسمى 19 I3S-Covid (نظام ذكاء لمجتمع أكثر أماناًا في كوفيد-١٩). YOLOV3 و YOLOV3، AlexNet، زيادة البيانات، نقل التعلم ، اكتشاف الكائنات ، تتبع الكائن.

الكلمات المفتاحية: YOLOV3، AlexNet، زيادة البيانات، نقل التعلم ، اكتشاف الكائنات ، تتبع الكائن.

ACKNOWLEDGEMENTS

First of all, I thank Allah Almighty for giving me the strength and patience to finish my search. This study is dedicated from my heart to my beloved parents, who have inspired me and gave me strength when I thought about giving up, they provide me with spiritual, emotional and financial support. To my brothers, sisters, relatives, professors, friends and colleagues who shared with me their words of advice and encouragement to finish this study.

I would like to express my thanks of gratitude to my Supervisor Dr. BOUARARA Hadj Ahmed, for all the instructions, assistance, and supports to my graduate project. Especially, I deeply appreciate his patience with my many questions.

Furthermore, I would like to express my sincere thanks to myself for my great patience and effort.

Dedicate

I dedicate my dissertation work to my family and many friends. A special feeling of gratitude to **my loving Parents**, whose words of encouragement and push for tenacity ring in my ears. My sisters **Slamet**, **Hanane** and **Sara** have never left my side and are very special, they are part of my soul.

I also dedicate this dissertation to my brothers **Zakaria** and **Yacine** who have supported me throughout the process. I will always appreciate all they have done,

A special thanks to my dear Aunt **Dr. Taibi Nour el Houda** and my dear Uncle's wife **Dr. Rachida Rouane**, who have supported me and stood by me when things looked bleak.

ACRONYMS

ACRONYMS

WHL : *World Health Organization*
DL : *Deep Learning.*
ML : *Machine Learning.*
AI : *Artificial Intelligence.*
CNN : *Convolutional Neural Network.*
CV : *Computer Vision.*
ANN : *Artificial Neural Network.*
ReLu : *Rectified Linear Unit.*
OF : *Overfitting.*
UF : *Underfitting.*
COVID-19 : *Coronavirus Disease 19.*
SARS-CoV-2 : *Severe Acute Respiratory Syndrome Coronavirus 2.*
YOLO :*You Only Look Once.*
IOU : *Intersection Over Union.*
SSD : *Single Shot Multi-Box Detector.*
NMS : *Non Maxima Suppression.*

CONTENTS

List of Tables

List of Figures

Figure **Page**

CHAPTER 1

GENERAL INTRODUCTION

1.1 Introduction

1.1.1 Context

The Corona pandemic suddenly came to the world, while the world was not prepared for it, even with the slightest precautions, which led to many human losses. When we talk about the world, we highlight the Arab world, especially Algeria, where there is a serious lack of artificial intelligence technology and computer vision. We focused on our project on computer vision, which involves providing surveillance cameras and drones in all public places and streets. As for the idea of this project, it aims to reduce the coronavirus by applying the necessary precautions codified by the World Health Organisation , and all of this will be monitored by surveillance cameras and drones to uncover the wrongs committed by citizens such as not wearing a mask or disrespect social distancing, also, it will play the role of a heat detector, it dispenses us from the use of a thermometer, moreover, it is interested in discovering the necessary doses that the patient will inject.

1.1.2 Motivation and problematic

COVID-19 has thrown a big challenge across the workplaces around the globe for cultivating safe organizational culture. New safety protocols are being introduced with daily routines for enhanced safety programs. Technology is the key player here. Video anomaly detectors driven by computer vision have been proven to be effective here. This can automatically monitor and analyze any anomaly like the absence of safety masks with other necessary regulations like maintaining social distancing to ensure employee safety. COVID-19 has enforced a reduced workforce leading to declining margins in 2020. Hence, 2021 will be a crucial year to compensate

for this gap. To do that, industrial leaders are looking for optimal Quality, accuracy, low cost, and flexibility by leveraging technologies like AI and computer vision.

Non-destructive testing computer vision is one such application solution that can detect defects and identify the area with a high probability of anomalies using NDT techniques with radiology images. This automated computer vision feature widens the visible spectrum and detects any defects on the metal surface, which is usually invisible to the human eye.

1.1.2.1 Sensor networks and IOT

Intuitive control interfaces are helping to improve the integration of sensor and vision data with new advancements. Here robust edge computing and closed-loop information exchange is also playing a crucial role. Automated surveillance cases have been unleashed through video analytics, which can automatically detect and alert security events, which undoubtedly contributes to physical security at national borders.

In 2021, we can see more advancement in this area where more advanced capabilities will integrate machine learning-based models and physics for exploring co-relate vision anomalies with sensor anomalies. This will help to generate actionable insights. As a result, the focus will be on developing related frameworks, the capability to integrate data, and asset monitoring with video insights. This will help in diagnosing track anomalies, asset downtime, and plant safety.

1.1.2.2 Data augmentation and annotation

The sophistication of computer vision models depends on their training data volume and its Quality. Data annotation, a manual task, and can be sourced or outsourced is a tedious and complicated task too. This needs a good amount of training as well as knowledge of annotation tools. Besides, it would help if you tracked annotation and its speed. As a consequence, it increases the chance of human error and cost. However, with the advancement of AI, new computer vision platforms are in place that helps to automate data labeling. This also ensures faster data throughput and minimal error. Computer vision trends 2021 will witness end-end automated solutions for image and video annotation. This automated solution will seamlessly fuel data pipelines helping faster activation of computer vision applications.

1.2 Object detection and Object tracking

Detecting and tracking objects are among the most prevalent and challenging tasks that a surveillance system has to accomplish in order to determine meaningful events and suspicious activities, and automatically annotate and retrieve video content. Under the business intelligence notion, in our thesis the object can be a face, a head and a human.

In this thesis, our focus is on mask detection and social distancing detection, we use transfer learning for each algorithm to improve the generalization ability of the mode, and helps to

improving results and to speed up the neural network. Moreover, to improve the accuracy we use the different optimization techniques and we change the hyper-parameters until we get the optimal weights.

1.2.1 Mask detection

To curb certain respiratory viral ailments, including COVID-19, wearing a clinical mask is very necessary. The public should be aware of whether to put on the mask for source control or aversion of COVID-19.Therefore, face mask detection has become a crucial task in present global society. Face mask detection involves in detecting the location of the face and then determining whether it has a mask on it or not, and in order to embody our project we used:

- **YOLOv3**, which is used for human detection as it improves predictive accuracy, particularly for small-scale objects.
- **MobileNetV2**, which is used for real-time detection, and for webcam feed to detect the purpose webcam which detects the object in a video stream.

1.2.2 Social distancing detection

Social distancing associates with the measures that overcome the virus' spread, by minimizing the physical contacts of humans, such as the masses at public places, and for reduce risk of serious illness from COVID-19, we use the MobileNet-SSD model and YOLOV3 to detect the poeple, after human detection, the Euclidean distance between each detected centroid pair is computed using the detected bounding box and its centroid information.

1.3 Classification

Deep Convolutional Neural Networks (CNNs) are widespread, efficient tools of visual recognition. We apply two pre-trained deep Convolutional Neural Networks (CNN) namely, VGG-16, AlexNet and use them to extract deep features from the obtained regions (mostly face and people) and use them for the two purposes of mask detection and social distancing detection.

1.4 Goals

- Detect a face mask detection and social distancing using different CNN models.
- Use data augmentation to improve the performance and outcomes of CNN models by forming new and different examples to train datasets to obtain better and more accurate performance.

- Use transfer learning to accelerate training and achieve more accurate results, all with less data and time.
- The main goal of the project is to reduce Covid-19 disease and to create a static CNN models that will be a solution or at least to control and facilitates any process to eliminate any problem are pandemic in the future.

1.5 Organization work

Work organization Our thesis is made up of four chapters:

Chapter 01: In the first chapter we will present the techniques used in this memoir to solve this problem which is the use of surveillance cameras and artificial intelligence (AI) for the post COVID-19 vaccine, their different architectures of models and their different fields of application.

Chapter 02: It covers what COVID-19 disease is and its effects on daily life, and how to reduce it to protect yourself and others, as well as explain what COVID-19 vaccines are and the required doses for them.

Chapter 03: We will talk about future work and give an in-depth explanation of the project's framework, mostly theoretically without getting into the coded software. We will introduce the techniques used for each modules (mask detection, social distancing detection).

Chapter 04: Experimentally validate the effectiveness of our work and discuss our conclusions and highlight future work of research to improve our models.

CHAPTER 2

DEEP LEARNING

"Artificial Intelligence is the new electricity." Andrew Ng

Artificial intelligence is broad branch of computer science, the goal of IA is to create systems that can function intelligently and independently, The purpose of artificial intelligence is to make the machine learn by itself without human guidance and to think like humans and to mimic human consciousness. The ideal feature of IA is its ability to make a decision that has the best chance of achieving a specific goal, techniques from the field of artificial intelligence, and more specifically deep learning methods, have been core components of most recent developments in the field of computer vision. Where it's already being exploited to solve the problems and diseases of the times like COVID-19. The COVID-19 pandemic poses a number of challenges to the Artificial Intelligence (AI) Community, among these challenges are use it for social control and detection of face mask violation and social distance, use it in the search for treatments and a vaccine, also to detect the violation of protective equipment, all of these using a Convolutional Neural Network (CNN)

2.1 COVID-19 pandemic defined

We now have a name for the disease caused by coronavirus and it's COVID-19", said Dr Tedros Adhanom Ghebreyesus, Director-General of WHO on Feb 11, 2020 [30], World Health Organization (WHO) recently updated the name novel coronavirus pneumonia, previously named by Chinese scientists [30],to coronavirus disease 2019 (COVID-19). More attention should be paid to comorbidities in the treatment of COVID-19. In the literature, COVID-19 is characterized by the symptoms of viral pneumonia such as fever, fatigue, dry cough, and lymphopenia. Many of the older patients who become severely ill have evidence of underlying illness such as cardiovascular

disease, liver disease, kidney disease, or malignant tumours [7] Humanity is currently facing a new type of Coronavirus, where the first patients with coronavirus disease 2019 (COVID-19) were recorded in December 2019 in China [24], since then, the Coronavirus has spread globally.

2.2 COVID-19 and its impact on society

Social distancing involves staying away from people to avoid the spreading and catching the virus. It is a new emerging terminology which means to avoid the crowd. This has forced people to work from home and avoid social gatherings and contacting even their near ones [31]. Eric Kleinberg, a New York University sociologist, stated that "we've also entered a new period of social pain. There's going to be a level of social suffering related to isolation and the cost of social distancing that very few people are discussing this yet." Statistics WHO indicate that at least 165,000 have succumbed to the disease, most patients with COVID-19 suffer from severe respiratory problems, and elderly people in particular or persons with comorbidities seem to be at greatest risk [25]. In the following figure, we note the extent of the Coronavirus spread around the world

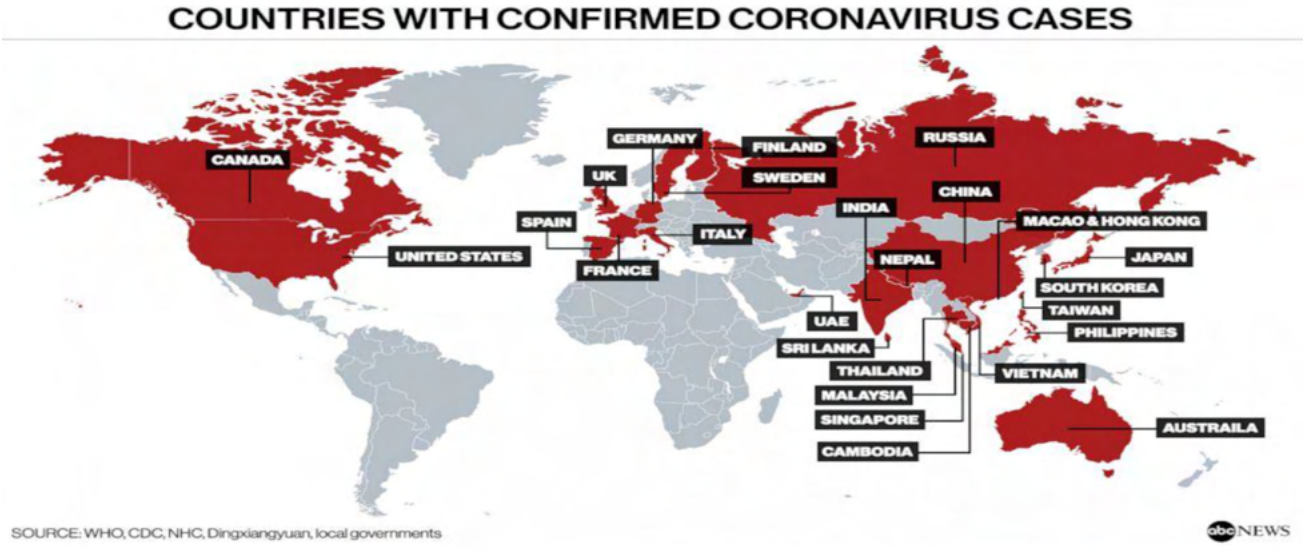


Figure 2.1: Countries with confirmed Corona Virus Cases. Figure reproduced from [25].

2.3 Coronavirus prevention

Prevention is better than cure, so it is best to follow the guidelines required by the World Health Organization to stay safe by taking some simple precautions, such as physical distancing, wearing a mask, keeping rooms well ventilated, avoiding crowds, cleaning your hands, and coughing into

a bent elbow or tissue. Check local advice where you live and work do it all! Maintain at least a 1-metre distance between yourself and others to reduce your risk of infection when they cough, sneeze or speak. Maintain an even greater distance between yourself and others when indoors. The further away, the better. Make wearing a mask a normal part of being around other people. The appropriate use, storage and cleaning or disposal are essential to make masks as effective as possible.And need for sterilization as well.

2.4 Machine Learning (ML)

Machine learning is a subset of artificial intelligence, this means that computer programs can learn automatically from new data and adapt to it, without consulting or being assisted by humans. It's building a mathematical model based on sample data, known as "training data," in order to make predictions or decisions without being explicitly programmed to perform the task [9]. Machine learning can be divided into several categories, can be supervised learning, unsupervised learning or reinforcement learning.

2.4.1 Overfitting (OF) and Underfitting (UF)

These two factors correspond to the two central challenges in machine learning: underfitting and overfitting. Underfitting occurs when the model is not able to obtain a sufficiently low error value on the training set. Overfitting occurs when the gap between the training error and test error is too large [15].

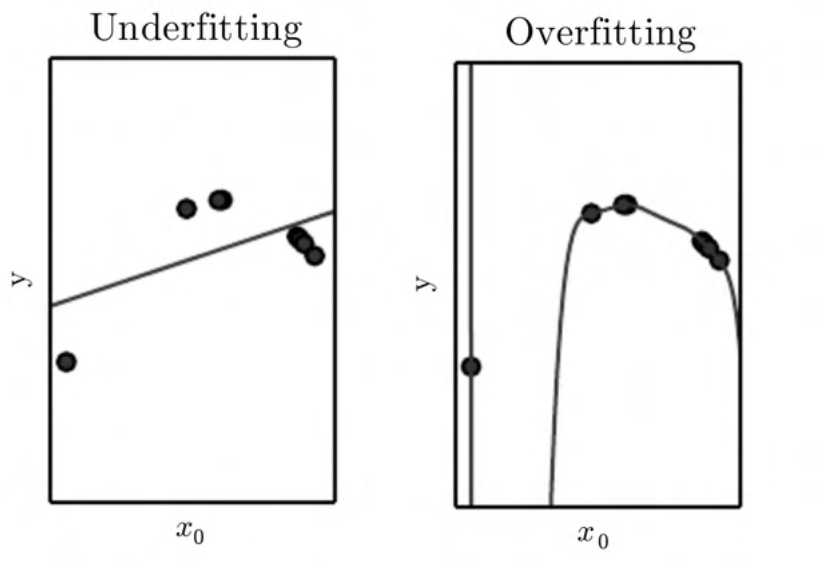


Figure 2.2: The Underfitting and Overfitting. Figure reproduced from [9].

The equation of underfitting is simple and not able to cover all points. It has a big problem in predicting the upcoming values and the loss function is large. The mathematical formula for overfitting used adjusts all points, which causes a problem in prediction due to the unhelpful excess of data.

2.4.2 Supervised learning

It's the process of training a model by entering data as an input, as well as correct output data, the term of "labeled data" is usually referred to the input/output pair. The learning algorithm is given labeled data and the computer builds relationships and patterns between the data to predict the outputs when we give it a new input. Supervised learning is often used to create machine learning models for two types of problems, regression and classification, the first, its outputs are real numbers and as for the second, its outputs are classes.

2.4.3 Unsupervised learning

In unsupervised learning, unlabeled data are used to predict unknown results and the algorithm has to determine how to discover the hidden patterns in the dataset [27] without the need for human intervention, so the main task is to find the solutions on its own. Unsupervised learning models are utilized for two main tasks, clustering and dimensionality reduction.

2.4.4 Reinforcement learning

It's a learning paradigm concerned with learning to control a system so as to maximize a numerical performance measure that expresses a long-term objective [8], Safe Reinforcement Learning can be defined as the process of learning policies that maximize the expectation of the return in problems in which it is important to ensure reasonable system performance and/or respect safety constraints during the learning and/or deployment processes [2].

2.5 Deep learning (DL)

Deep learning is a form of machine learning that enables computers to learn from experience and understand the world in terms of a hierarchy of concepts. Because the computer gathers knowledge from experience [15], it's a particular case of machine learning but at a deep level, if we do a simple comparison between deep learning and machine learning, we will find the following result: The more data, the Performance increased in deep learning, as for the machine learning, the more data, the stable in the performance. So in these chapters we will focus on the deep learning field especially the Convolutional Neural Network.

2.6 Computer vision (CV)

Computer vision (CV) is a study of how to use computer simulation of human visual science, its main task is through the collection of images (or video) analysis and understanding [14], we highlight computer vision combined with artificial intelligence algorithms that achieved important results in image recognition (such as face recognition, handwritten characters), image classification, it's one of the areas that's been advancing rapidly thanks to deep learning. Deep learning computer vision is now helping self-driving cars figure out where the other cars and pedestrians around so as to avoid them, at the same time, high-tech such as intelligent robot and many other disciplines

2.6.1 State of Computer Vision

Figure 2.3: Data Vs. Hand engineering .

2.7 Artificial Neuron (AN)

Artificial Neuron is a basic building block of every artificial neural network. Its design and functionalities are derived from observation of a biological neuron that is basic building block of biological neural networks (systems) which includes the brain, spinal cord and peripheral ganglia. Similarities in design and functionalities can be seen in 2.4 [12]. Where the left side of a figure represents a biological neuron with its some, dendrites and axon and where the right side of a figure represents an artificial neuron with its inputs, weights, transfer function, bias and outputs [12].

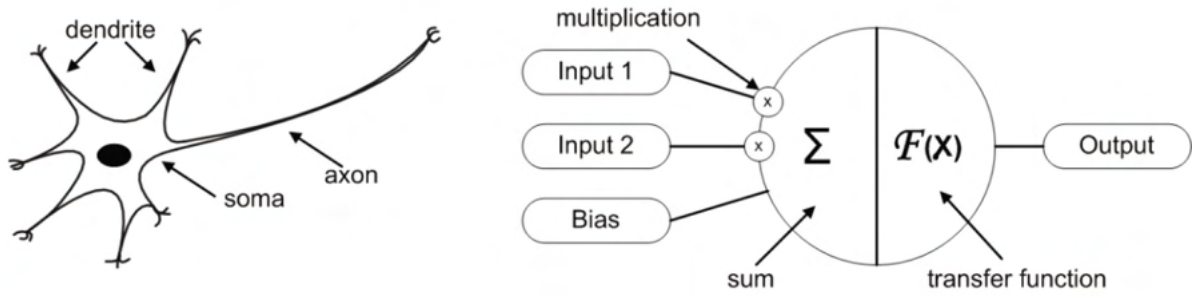


Figure 2.4: Biological and artificial neuron design. Figure reproduced from [12].

2.8 Artificial Neural Networks (ANNs)

An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as process information, human brain [18], that mimic the human operations and the way the brain analyzes and processes information. Using the idea that brain cells think and analyze data, to design a similar algorithm, the idea has nothing to do with human thinking, but rather with the method of analysis. Artificial neurons are designed to gathers neurons in the brain, which contain multiple layers, the first having x1, ..., xn inputs, and the following layers considers the outputs of the previous layers an inputs to it ,until the output layers that has a single neuron output or multi neuron output. The neuron output signal "a" is given by the following relationship:

$$\mathrm{a} = f(g) = f\left(\sum_{j=1}^{n} w_j x_j\right)$$

Where

w_j The weight vector

w_j The input vectors, and the function is $f(g)$ is referred to as an activation function

A typical artificial neuron and the modeling of a multilayered neural network are illustrated in Figure 2.5 [18], In Figure "a" we have one layer with four inputs and one output, which is the activation function, in Figure "b" we have a simple three layer, comprised of an input layer, a hidden layer and an output layer. This structure is the basis of a number of common ANN architectures,

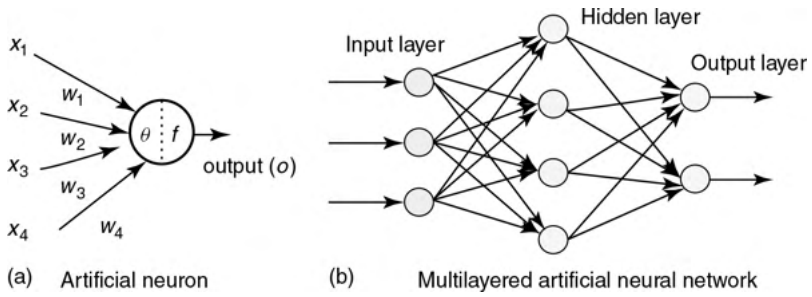


Figure 2.5: Architecture of an artificial neuron and a multilayered neural network. Figure reproduced from [18].

2.8.1 Activation Function

Different activation functions can exist to activate the neuron they are listed in Table 1 [23]. The most used functions are: "Sigmoid", "Tanh", "ReLu", and "Softmax".

The name of function	Input / output relationship
Sigmoid	$\frac{1}{1+e^{-x}}$
Tanh	$\text{Tanh}(x) = \frac{1-e^{(-2x)}}{1+e^{(-2x)}}$
ReLU	$\text{ReLu}(x) = \max(x,0)\ f(x) = x, x \geq 0\ f(x) = 0, x < 0$
SoftMax	$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{k} e^{zk}}$ For j = 1,...,K

Table 2.1: Different activation functions [23]

2.9 Convolutional Neural Network (CNN)

Convolutional neural networks are deep learning algorithms that take input images and convolves it with filters or kernels to extract features. A NxN image is convolved with a fxf filter and this convolution operation learns the same feature on the entire image [33] In recent years, the convolutional neural network (CNN) has emerged as the prevalent model for the machine learning and computer vision. Now, deep CNNs are widely used in a broad range of real-life applications, such as image classification, object detection and image segmentation [16]. In the following figure, we have an example of 2-D convolution with a kernel

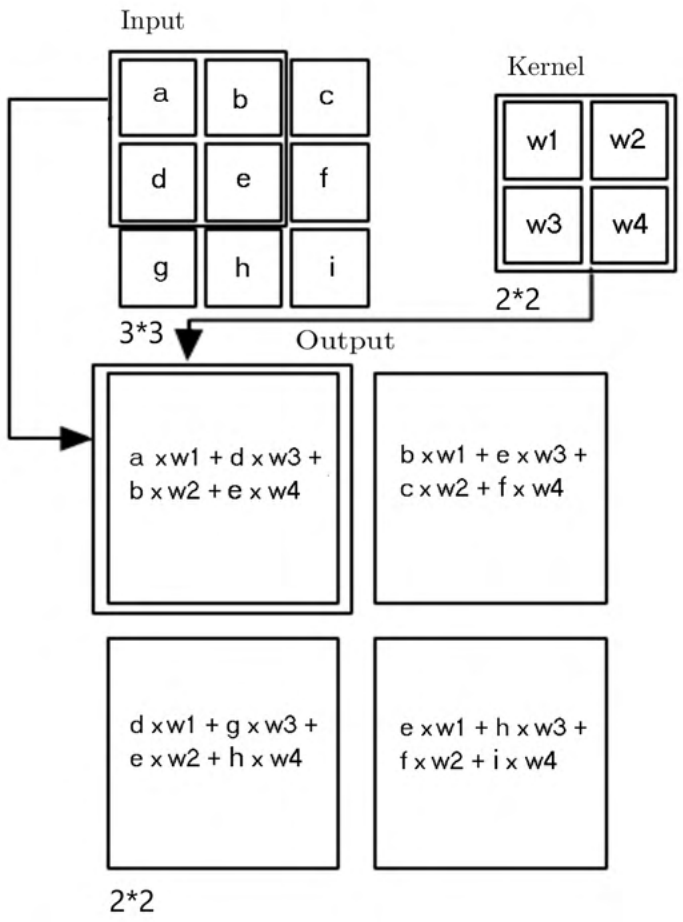


Figure 2.6: An example of 2-D convolution with a kernel.

2.9.1 Edge detection

Edge detection is a fundamental image processing technique which involves computing an image gradient to quantify the magnitude and direction of edges in an image, image gradients are used in various downstream tasks in computer vision such as line detection, feature detection, and image classification [19].

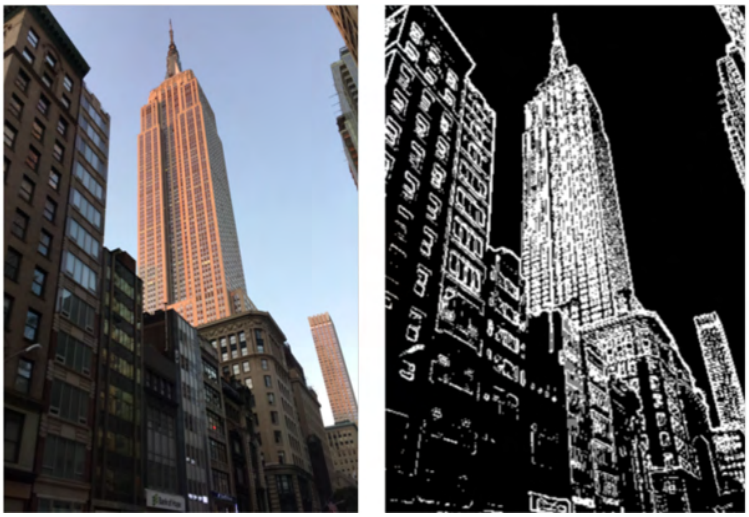


Figure 2.7: Edge detection using Sobel filters. Figure reproduced from [19].

2.9.1.1 More edge detection

In the following figure, we apply the edge detection using Sobel filters

1	1	1
0	0	0
-1	-1	-1

Horizontal edge detection

1	0	-1
2	0	-2
1	0	-1

Sobel filter (puts a little bit more weight to the central row, and this makes it maybe a little bit more robust) them

3	0	-3
10	0	-10
3	0	-3

Horizontal Scharr filter

the both of them have an horizontal version

2.9.2 Padding

One of the drawbacks of the convolution step is the loss of information that might exist on the border of the image [3], that the filter does not pass to all the pixels evenly, so a simple and effective way to solve the problem is to use a zero padding, and this makes it pass by all the borders at least three times, while it was only once, and also it keep the dimensions of the input matrix. For example, in 2.8, the input image was N = 6, after the padding N= 8, F = 3 and stride= 1, the result will be 6x6

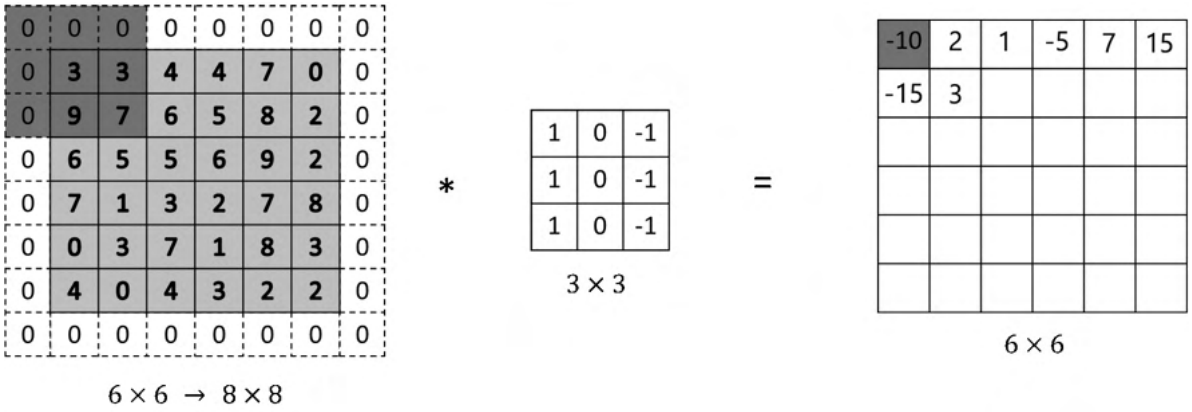


Figure 2.8: ConvNet application using a single padding.

The valid convolutions : "No padding", it's mean

$$\boldsymbol{N.N} * \boldsymbol{F.F} \quad \longrightarrow \quad (\boldsymbol{N} - \boldsymbol{F} + \mathbf{1}) * (\boldsymbol{N} - \boldsymbol{F} + \mathbf{1})$$

The Same convolutions : Dimension of input matrix = dimension of output matrix

$$(\boldsymbol{N.N}) * (\boldsymbol{F.F}) \longrightarrow ((\boldsymbol{N} + 2\boldsymbol{P} - \boldsymbol{F})/\boldsymbol{S}) + \mathbf{1} * ((\boldsymbol{N} + 2\boldsymbol{P} - \boldsymbol{F})/\boldsymbol{S}) + \mathbf{1}$$

Through this equation we find this rule to achieve the same convolutions:

$$\boldsymbol{P} = (\boldsymbol{F} - \mathbf{1})/\mathbf{2}$$

Strided convolutions It's helpful for save time but the quality decreases

$$(\boldsymbol{N.N}) * (\boldsymbol{F.F}) \Longrightarrow ((\boldsymbol{N} + \mathbf{2}\boldsymbol{P} - \boldsymbol{F})/\boldsymbol{S}) + \mathbf{1} * ((\boldsymbol{N} + \mathbf{2}\boldsymbol{P} - F)/\boldsymbol{S}) + \mathbf{1}$$

2.9.3 Pooling

We have three types

Max pooling : Get the max value

Min pooling : Get the min value

Average pooling : Get the average value

2.9.3.1 Classic model

LeNet-5 : LeNet was introduced by Yan LeCun for digit recognition Figure 2.9. It includes 5 convolutional layers [3], it's a simple and old network which contains only black and white images

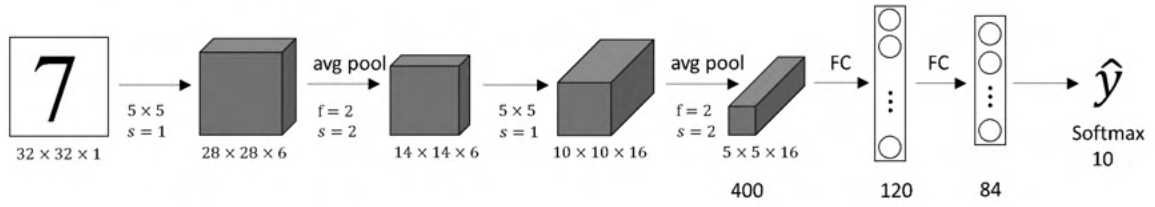


Figure 2.9: Lenet-5 architecture.

LeNet-5 CNN architecture is made up of 7 layers. The layer composition consists of 3 convolutional layers, two subsampling layers which is average pooling and 2 fully connected layers. In this example we have 60K parameters, the lower the length and width, the more layers The concept of the LeNet-5 network

Input (image) => CNN +Pool => CNN +Pool => FC =>FC => out put

After update:

Input (image) => CNN +Pool => CNN +Pool => FC =>FC => softmax

Often this type of convolutional uses Tanh or sigmoid because ReLu function did not exist

AlexNet : The architecture consists of eight layers: five convolutional layers and three fully-connected layers. Through the following example, this network look like LeNet-5 but are deeper, with LeNet-5 we have 60K parameters and with AlexNet we have 60M, that means more accuracy and more time. Often the ReLu function is used in place of the Tanh or Sigmoid function.

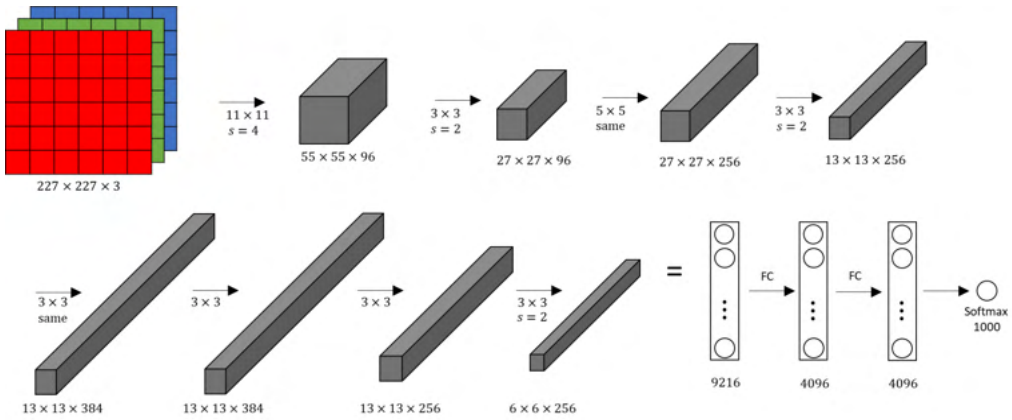


Figure 2.10: AlexNet architecture.

VGG-16 : VGG16 architecture consists of 16 layers. It is considered to be one of the excellent vision model architecture till date. Most unique thing about VGG16 is that instead of having a large number of hyper-parameter they focused on having convolution layers of 3x3 filter with a stride 1 and always used same padding and max pooling layer of 2x2 filter of stride 2.

Number 16 in the name VGG-16 refers to the fact that this has 16 layers that have some weights. This is a pretty large network, and has a total of about 138 million parameters.

VGG-19 neural network which is bigger then VGG-16, but because VGG-16 does almost as well as the VGG-19 a lot of people will use VGG-16. We notice that the length and the width decrease in half, while the deep increases to double.

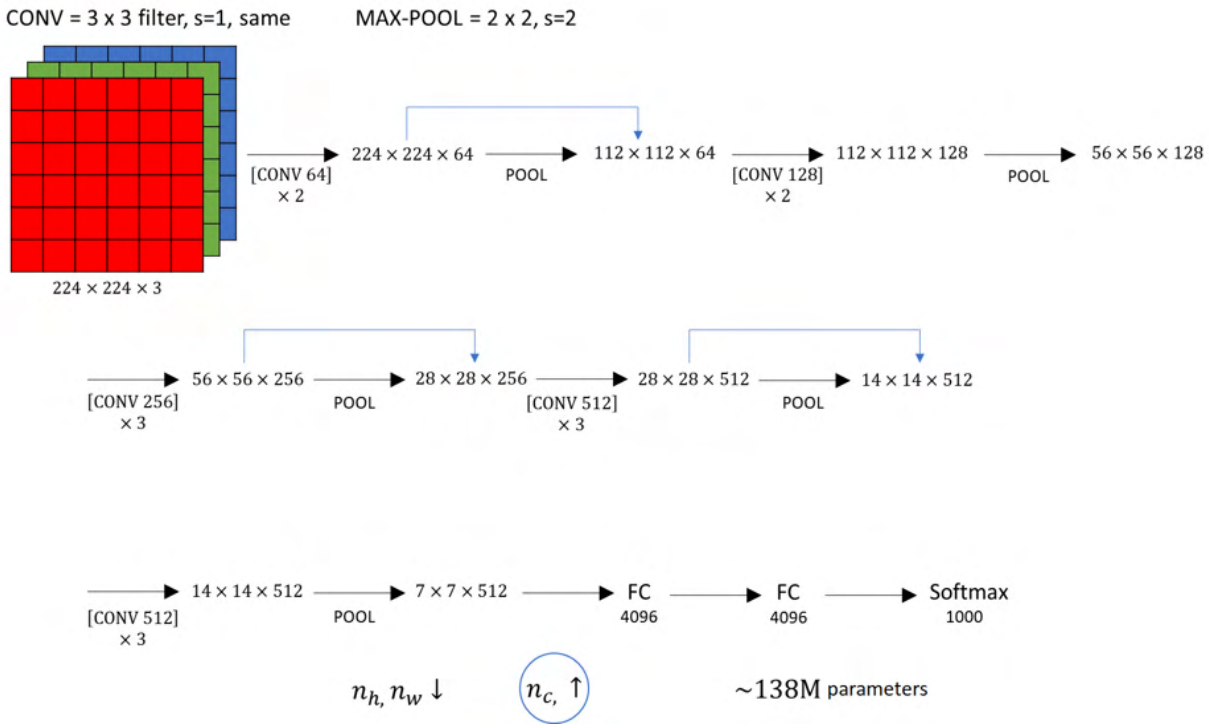


Figure 2.11: VGG-16 architecture.

Residual Networks (ResNets) :

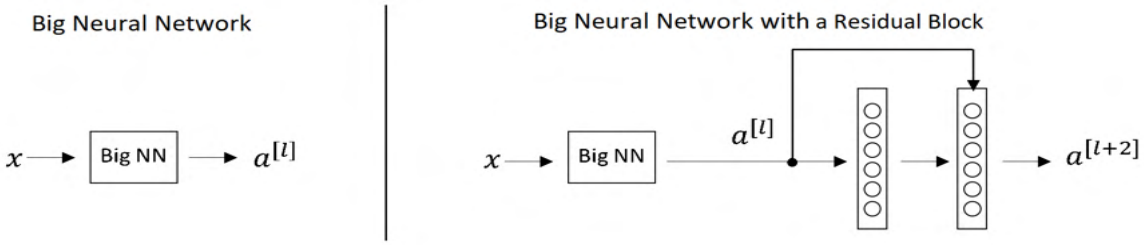


Figure 2.12: Big NN with ResNets architecture.

The picture above 2.12 is the most important for developers looking to quickly implement the residual block and test it out, the most important modification to understand is the 'Skip Connection', identity mapping. This identity mapping does not have any parameters and is just there to add the output from the previous layer to the layer ahead.

How does the ResNets work :

$$\text{ReLu} \quad \boldsymbol{a > 0}$$
$$\boldsymbol{a[l+2] = g(z[l+2] + a[l])}$$
$$\boldsymbol{= g(b[l+2] \cdot a[l+1] + b[l+1] + a[l])}$$

When we do the regularization for the values of $z[l+2]$ this leads to reduces the value $w[l+2]$ and that's mean increased the values of $a[l+1]$, Thus reducing the value $b[l+2]$ As the $w[l+1]$ approaches zero, the $a[l+1]$ increases dramatically, which makes the $b[l+2]$ also decrease to zero, which makes each of z devolving to zero, and here it becomes $a[l+2] = g(a[1])$ And because the $ReLu$ function is equal to the value of the input, if it were positive, then that means increasing two layers using the ResNets, doesn't almost change the output Trainer error

Ps: The $a[l]$ must be the same size as the $z[l+2]$ Exception: if the size of $a[l] \neq z[l+2]$, so we should to add a new matrix and we multiply it by Ws

Eg:

$a[l] = 128 * 1$

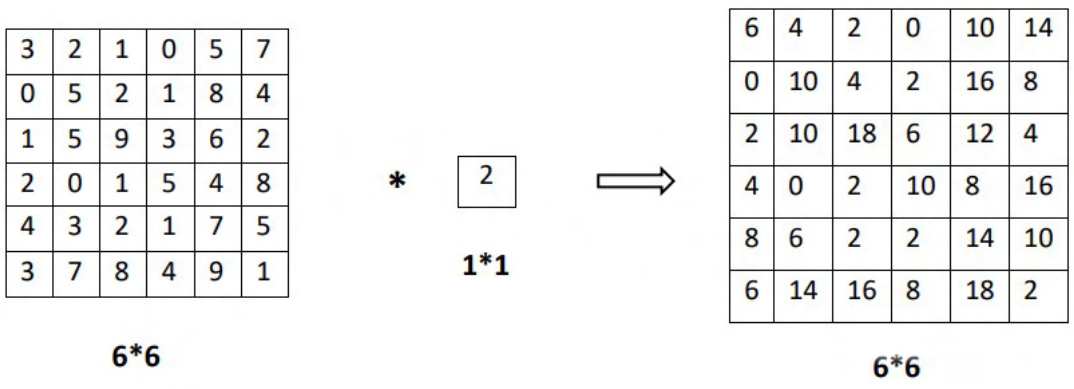
$z[l+2] = 256 * 1$

So $Ws = 256 * 128 \rightarrow Ws * a[l] = 256 * 128128 * 1$

$= 256 * 1$

2.9.3.2 Networks in Networks and 1x1 Convolutions

We have an example of 1 x 1 convolution



It's seems like the same matrix with the dimension but each number multiply by 2, so it's useless in this case

1x1 Convolutions be useful when we use it with a stereoscopic matrix

2.9.3.3 Using 1 x 1 convolution

It is useful to reduce the number of parameters, in the following example we see how it works A 1 x 1 Convolution is a convolution with some special properties in that it can be used for

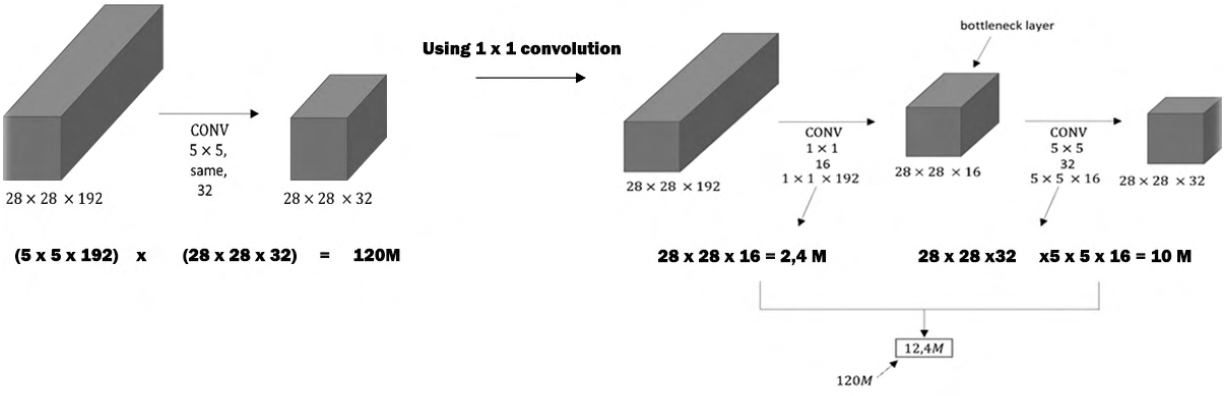


Figure 2.13: The role of 1 x 1 convolution in reducing parameters.

dimensionality reduction, efficient low dimensional embeddings, and applying non-linearity after convolutions. It maps an input pixel with all its channels to an output pixel which can be squeezed to a desired output depth.

In our figure 2.13, the number of parameters has been reduced from 120 million to 12.4 million.

2.9.3.4 Transfer Learning

Many computer vision researchers have trained their algorithms on these datasets. Sometimes this training takes several weeks and might take many GPUs. The fact that someone else has done this task and gone through the painful high-performance research process means that we can often download open source weights. Next, we can use them as a very good parameter's initialization for own neural network. That is, we can use transfer learning to transfer knowledge from some of these very large public datasets to our own problem with a freeze method.

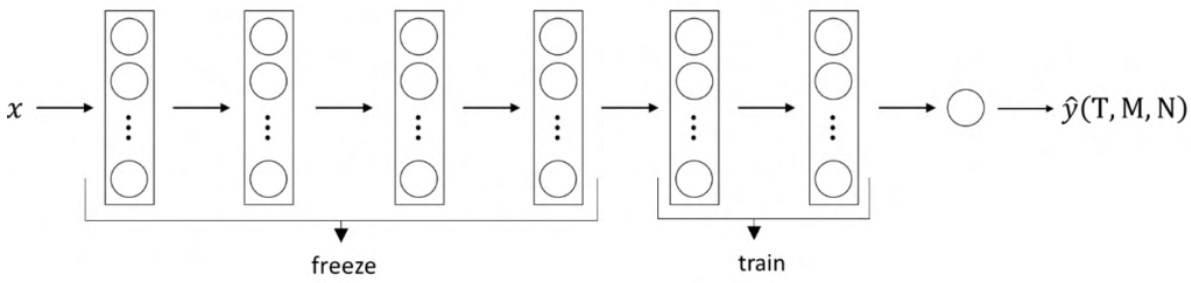


Figure 2.14: Transfer learning Technique

So in this Figure 2.14 we will freeze all the weights of the previous layers by calculating the activation function and saving them in our disk, sometimes we train the softmax layer only with our own issues with the new output and sometimes we can change the last layers using SoftMax.

Its help us to gain the time and accelerate the training because we don't need to compute the forward and backward propagation because we have an optimal weights

2.9.3.5 Data augmentation

Most computer vision tasks could use more data and data augmentation is one of the techniques that is often used to improve the performance of computer vision systems and augment the accuracy of any algorithm.

We have many data augmentation methods in computer vision, Some of them are identified in the following figure 2.15

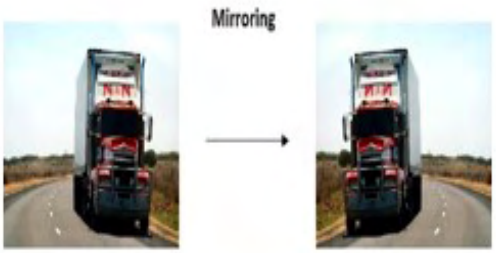
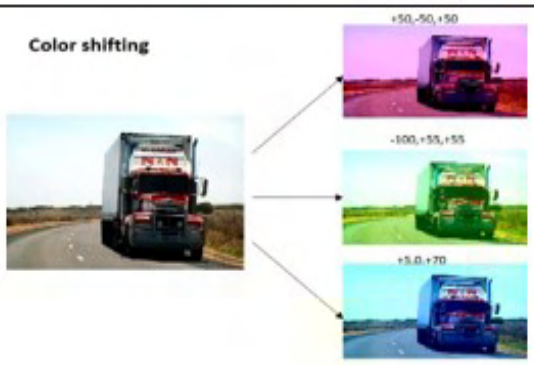
Type of data augmentation	Example
Mirroring	
Random cropping	
Color shifting	

Figure 2.15: Different augmentation methods.

We have also (Rotation, Shearing, local wraping)

2.9.4 Dataset

The best way to make a machine learning model generalize better is to train it on more data [9], it's better to create your own datasets than to use the data augmentation

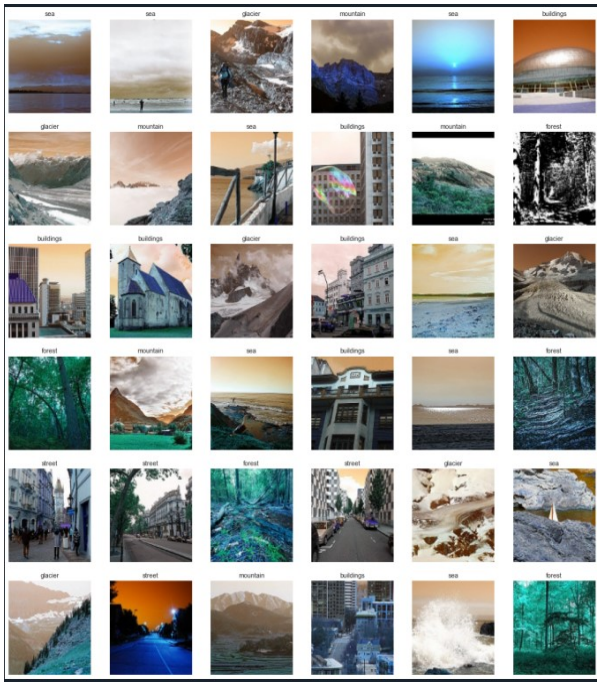


Figure 2.16: A simple Dataset image.

The MNIST Dataset : The MNIST dataset consists of scans of handwritten digits and associated labels describing which digit 0–9 is contained in each image [9], the dataset has 70,000 images to train and test the model. The training and test set distribution is 60,000 train images and 10,000 test images. The size of each image is 28x28 pixels (784 pixels) [5].

Fashion-MNIST Dataset : The Fashion-MNIST dataset which consists of 60,000 training set pictures and 10,000 test set pictures. Each symbol is a gray-scale image of 28×28, linked to 10-category labels as shown in Figure 2.17 [13].

Figure 2.17: Fashion-MNIST Dataset.

2.10 Conclusion

In this chapter we presented what is artificial intelligence, specifically deep learning, as we have highlight on the convolutional neural network and its characteristics.

CHAPTER 3

COVID-19

"We are in this together - and we will get through this, together."
UN Secretary-General Antonio Guterres

In December 2019, an outbreak of severe acute respiratory syndrome coronavirus 2 (SARSCoV-2) infection occurred in Wuhan, Hubei Province, China and spread across China and beyond. On February 12, 2020, WHO officially named the disease caused by the novel coronavirus as Coronavirus Disease 2019 (COVID-19) [35]. Since most COVID-19 infected patients were diagnosed with pneumonia and characteristic CT imaging patterns, radio logical examinations have become vital in early diagnosis and assessment of disease course. To date, CT findings have been recommended as major evidence for clinical diagnosis of COVID-19 in Hubei, China. This review focuses on the etiology, epidemiology, and clinical symptoms of COVID-19, while highlighting the role of chest CT in prevention and disease [35].

3.1 COVID-19 pandemic

COVID-19 is the disease caused by a new coronavirus called SARS-CoV-2. WHO first learned of this new virus on 31 December 2019, following a report of a cluster of cases of 'viral pneumonia' in Wuhan, People's Republic of China. In most cases, COVID-19 causes mild symptoms including dry cough, tiredness and fever, though fever may not be a symptom for some older people. Other mild symptoms include aches and pains, nasal congestion, runny nose, sore throat or diarrhoea. Some people become infected but don't develop any symptoms and don't feel unwell. Most people recover from the disease without needing special treatment. Around 1 out of every 6 people who get COVID-19 becomes seriously ill and has difficulty breathing [1].

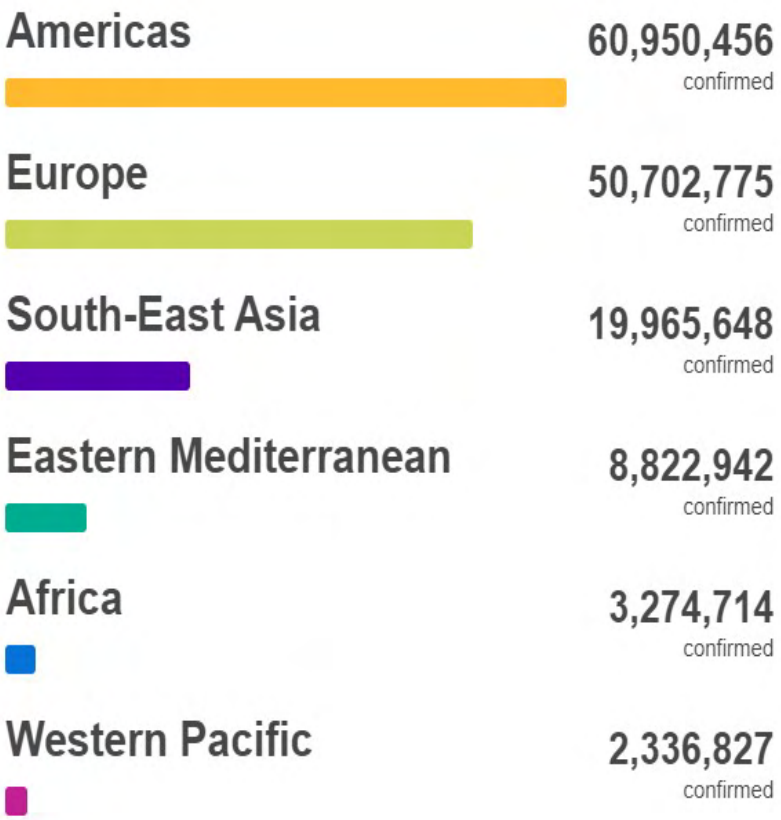


Figure 3.1: Situation by WHO Region to the number of confirmed cases of COVID-19 in the world [1].

3.2 Main symptoms of Coronavirus (Covid-19)

3.2.1 High temperature

World Health Organization (WHO) reported that there have been pneumonia cases in Wuhan City, Hubei Province, China, but the etiology was unknown. The case was developed very fast, until January 7, 2020, the Chinese government said that pneumonia was a new type of coronavirus or covid-19. Common signs and symptoms of covid-19 infection include symptoms of acute respiratory disorders such as fever, coughing and shortness of breath. The average incubation period is 5–6 days with the longest incubation period of 14 days. In severe cases, covid-19 can cause pneumonia, acute respiratory syndrome, kidney failure, and even death. The clinical signs and symptoms reported in the majority of cases are fever, with some cases having difficulty breathing, and X-rays show extensive pneumonia infiltrates in both lungs [29].

3.3 Effects of COVID-19 pandemic

COVID-19 (Coronavirus) has affected day to day life and is slowing down the global economy. This pandemic has affected thousands of peoples, who are either sick or are being killed due to the spread of this disease. The most common symptoms of this viral infection are fever, cold, cough, bone pain and breathing problems, and ultimately leading to pneumonia. This, being a new viral disease affecting humans for the first time, vaccines are not yet available. Thus, the emphasis is on taking extensive precautions [10], such as social distancing, and wearing of masks and this is among what we will cover in this thesis. Presently, the impacts of COVID-19 are widespread and have far-reaching consequences. It can be divided into different categories:

Healthcare

- Challenges in the diagnosis, quarantine and treatment of suspected or confirmed cases
- High burden of the functioning of the existing medical system
- Patients with other disease and health problems are getting neglected
- Overload on doctors and other healthcare professionals, who are at a very high risk
- Overloading of medical shops
- Requirement for high protection
- Disruption of medical supply chain

Economic

- Slowing of the manufacturing of essential goods
- Disrupt the supply chain of products
- Losses in national and international business
- Poor cash flow in the market
- Significant slowing down in the revenue growth

Social

- Service sector is not being able to provide their proper service
- Cancellation or postponement of large-scale sports and tournaments
- Avoiding the national and international travelling and cancellation of services
- Disruption of celebration of cultural, religious and festive events

- Undue stress among the population
- Social distancing with our peers and family members
- Closure of the hotels, restaurants and religious places
- Closure of places for entertainment such as movie and play theatres, sports clubs, gymnasiums, swimming pools, and so on.
- Postponement of examinations

This COVID-19 has affected the sources of supply and effects the global economy. There are restrictions of travelling from one country to another country. During travelling, numbers of cases are identified positive when tested, especially when they are taking international visits. All governments, health organizations and other authorities are continuously focussing on identifying the cases affected by the COVID-19. Healthcare professional face lot of difficulties in maintaining the quality of healthcare in these days [10].

3.4 How to protect yourself and others

3.4.1 Wearing a mask

The World Health Organization (WHO) advises the use of masks as part of a comprehensive package of prevention and control measures to limit the spread of SARS-CoV-2, the virus that causes COVID-19 [20], everyone 2 years and older should wear masks in public, masks should be worn in addition to staying at least 6 feet apart, especially around people who don't live with you, if someone in your household is infected, people in the household should take precautions including wearing masks to avoid spread to others. WHO continues to advise that anyone suspected or confirmed of having COVID-19 or awaiting viral laboratory test results should wear a medical mask when in the presence of others (this does not apply to those awaiting a test prior to travel). Depending on the type, masks can be used either for protection of healthy persons or to prevent onward transmission (source control), for any mask type, appropriate use, storage and cleaning or disposal are essential to ensure that they are as effective as possible and to avoid an increased transmission risk.

3.4.2 Social distancing

However, the use of a mask alone is insufficient to provide an adequate level of protection or source control, and other personal and community level measures should also be adopted to suppress transmission of respiratory viruses. Whether or not masks are used, compliance with social distancing and other infection prevention and control (IPC) measures are critical to prevent human-to human transmission of COVID-19 [20] Social distancing refers to measures aimed at reducing interactions within a community, which can include infected individuals as yet

unidentified, hence not in isolation. Since diseases transmitted through respiratory droplets require a certain physical proximity for contagion to occur, social distancing allows transmission to be reduced. Examples of social distancing measures that have been adopted include: the closure of schools and workplaces, closure of certain businesses, and cancellation of events to avoid mass gatherings. Social distancing is particularly useful in settings where there is community transmission of the virus, where the restriction measures imposed exclusively on known cases or on the most vulnerable segments of the population are considered insufficient to prevent new transmissions [32].

3.5 COVID-19 Vaccine

The world is in the midst of a COVID-19 pandemic. As WHO and partners work together on the response tracking the pandemic, advising on critical interventions, distributing vital medical supplies to those in need they are racing to develop and deploy safe and effective vaccines. Vaccines save millions of lives each year. Vaccines work by training and preparing the body's natural defenses the immune system to recognize and fight off the viruses and bacteria they target. After vaccination, if the body is later exposed to those disease-causing germs, the body is immediately ready to destroy them, preventing illness. Vaccines are a critical new tool in the battle against COVID-19 and it is hugely encouraging to see so many vaccines proving successful and going into development. Working as quickly as they can, scientists from across the world are collaborating and innovating to bring us tests, treatments and vaccines that will collectively save lives and end this pandemic.

There are three main approaches to designing a vaccine:

- The whole-microbe approach
- The subunit approach
- The genetic approach (nucleic acid vaccine)

As of 18 February 2021, at least seven different vaccines across three platforms have been rolled out in countries. Vulnerable populations in all countries are the highest priority for vaccination. At the same time, more than 200 additional vaccine candidates are in development, of which more than 60 are in clinical development. COVAX is part of the ACT Accelerator, which WHO launched with partners in 2020. COVAX, the vaccines pillar of ACT Accelerator, convened by CEPI, Gavi and WHO, aims to end the acute phase of the COVID-19 pandemic by [1]:

- Speeding up the development of safe and effective vaccines against COVID-19,
- Supporting the building of manufacturing capabilities,

- Working with governments and manufacturers to ensure fair and equitable allocation of the vaccines for all countries, the only global initiative to do so [1].

3.5.1 Vaccine doses

For some COVID-19 vaccines, two doses are required. It's important to get the second dose if the vaccine requires two doses. For vaccines that require two doses, the first dose presents antigens – proteins that stimulate the production of antibodies – to the immune system for the first time. Scientists call this priming the immune response. The second dose acts as a booster, ensuring the immune system develops a memory response to fight off the virus if it encounters it again. Because of the urgent need for a COVID-19 vaccine, initial clinical trials of vaccine candidates were performed with the shortest possible duration between doses. Therefore an interval of 21–28 days (3–4 weeks) between doses is recommended by WHO. Depending on the vaccine, the interval may be extended for up to 42 days – or even up to 12 weeks for some vaccines – on the basis of current evidence [1].

3.5.2 The COVID-19 candidate vaccine landscape and tracker

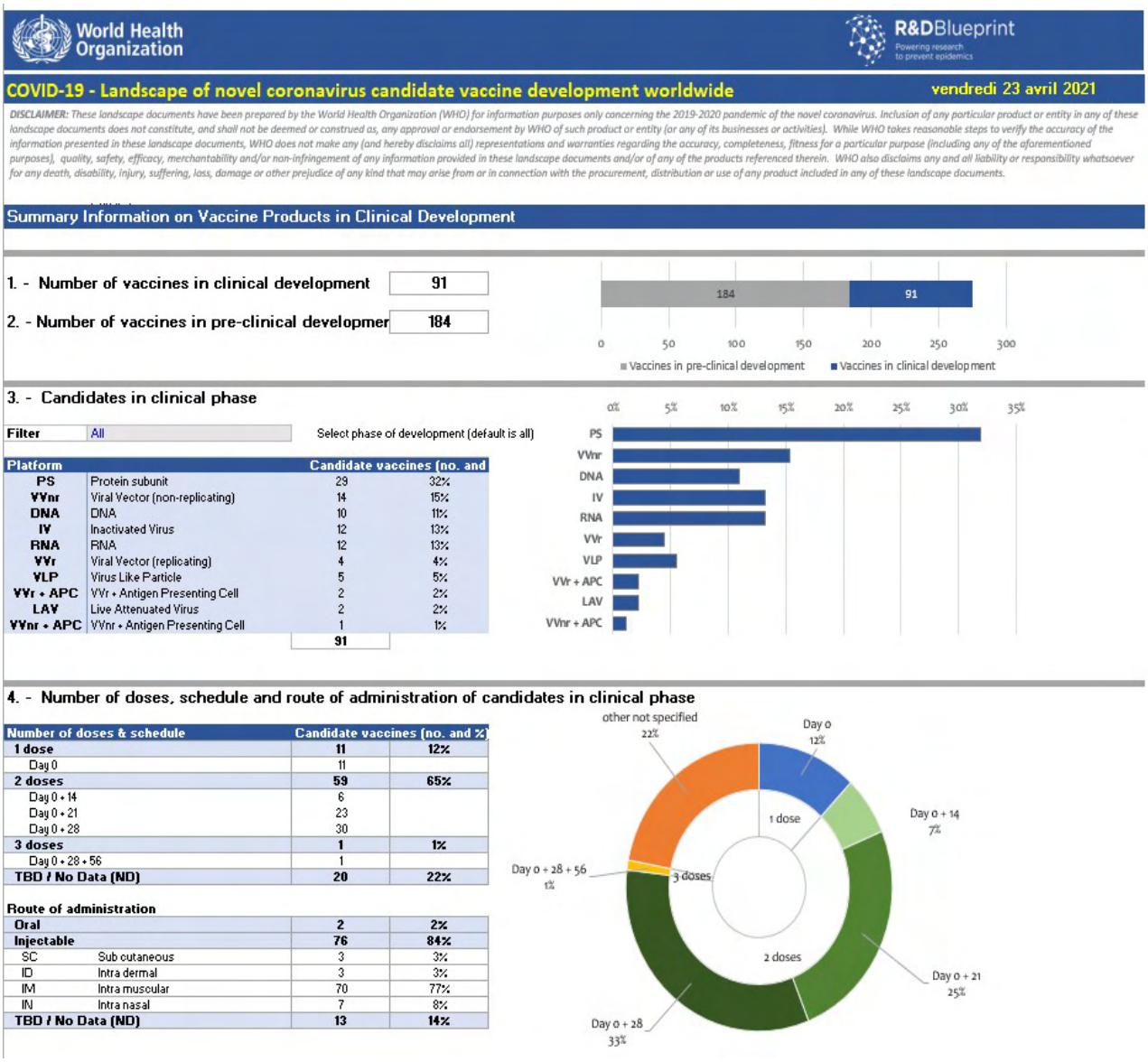


Figure 3.2: Tables of COVID-19 vaccine candidates in both clinical and pre-clinical development. Figure reproduced from [1].

provides summary tables of COVID-19 vaccine candidates in both clinical and pre-clinical development, provides analysis and visualization for several COVID-19 vaccine candidate categories, tracks the progress of each vaccine from pre-clinical, Phase 1, Phase 2 through to Phase 3 efficacy studies and including Phase 4 registered as interventional studies, provides links to published reports on safety, immunogenicity and efficacy data of the vaccine candidates, includes information on key attributes of each vaccine candidate and allows users to search for COVID-19 vaccines through various criteria such as vaccine platform, schedule of vaccination, route of

administration, developer, trial phase and clinical endpoints [1].

3.6 Conclusion

Coronavirus is an infectious disease that is transmitted from animals to humans and has caused many problems at all social, economic and health levels, and to reduce this crisis, the World Health Organization has suggested some measures to prevent it and scientists also from across the world are collaborating and innovating to bring us tests, treatments and vaccines that will collectively save lives and end this pandemic. Vaccines are a critical new tool in the battle against COVID-19 and it is hugely encouraging to see so many vaccines proving successful and going into development.

CHAPTER 4

CONTRIBUTION

'Deep learning will revolutionize supply chain automation." Dave Waters

4.1 Intelligence system for safer society in Covid-19 (I3S-Covid19)

The illustration shows a prototype of the final project ,we need in our project surveillance camera, sensors and drones in order to control people if they apply the necessary precautions against the coronavirus like wearing mask, respecting social distancing, detecting those with fever and disclosing the necessary doses for the vaccine. The project is implemented in public places such as schools, hospitals and shops because it has surveillance camera and sensors, as for the street, if there are drones to save on us the installation of surveillance cameras in every part of the streets.

The goals of the project is to reduce COVID-19 disease and to create a static CNN models that will be a solution or at least to control and facilitates any process to eliminate any problem are pandemic in the future

In this thesis, we are talking about two modules, which are mask detection and social distancing.

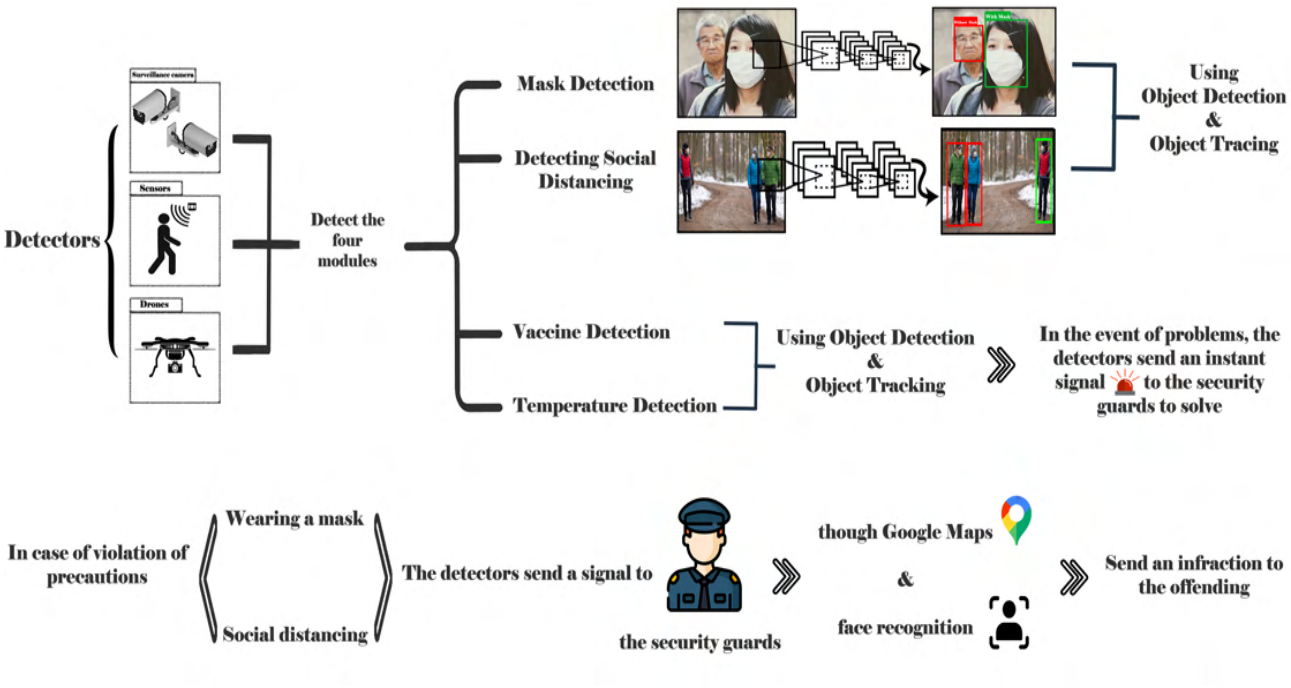


Figure 4.1: An illustration of I3S-Covid19 (Intelligence system for safer society in Covid-19).

4.2 Mask detection

4.2.1 Object detection

4.2.1.1 MobileNetV2

MobileNetV2 is a convolutional neural network architecture that seeks to perform well on mobile devices. It is based on an inverted residual structure where the residual connections are between the bottleneck layers. The intermediate expansion layer uses lightweight depthwise convolutions to filter features as a source of non-linearity. As a whole, the architecture of MobileNetV2 contains the initial fully convolution layer with 32 filters, followed by 19 residual bottleneck layers [22]. MobileNetV2: Each line describes a sequence of 1 or more identical (modulo stride) layers, repeated n times. All layers in the same sequence have the same number c of output channels. The first layer of each sequence has a stride s and all others use stride 1. All spatial convolutions use 3 × 3 kernels

1. **Model architecture :**

   We used transfer learning technology, we frozen 28 layers that means the weights can't be modified further. Layer Freezing is a technique to speed up neural network training

Input	Operator	t	c	n	s
$224^2 \times 3$	conv2d	-	32	1	2
$112^2 \times 32$	bottleneck	1	16	1	1
$112^2 \times 16$	bottleneck	6	24	2	2
$56^2 \times 24$	bottleneck	6	32	3	2
$28^2 \times 32$	bottleneck	6	64	4	2
$14^2 \times 64$	bottleneck	6	96	3	1
$14^2 \times 96$	bottleneck	6	160	3	2
$7^2 \times 160$	bottleneck	6	320	1	1
$7^2 \times 320$	conv2d 1x1	-	1280	1	1
$7^2 \times 1280$	avgpool 7x7	-	-	1	-
$1 \times 1 \times 1280$	conv2d 1x1	-	k	-	

Figure 4.2: The architecture of MobileNet model. Figure reproduced from [22]

by gradually freezing hidden layers. We built two new fully connected (FC) and we create our own softmax layer that predicts one of 2 classes (with mask, without mask). Transfer learning has several benefits, but the main advantages are saving training time, better performance of neural networks (in most cases), and not needing a lot of data. Usually, a lot of data is needed to train a neural network from scratch but access to that data isn't always available, this is where transfer learning comes in handy. With transfer learning a solid machine learning model can be built with comparatively little training data because the model is already pre-trained. Additionally, training time is reduced because it can sometimes take days or even weeks to train a deep neural network from scratch on a complex task.

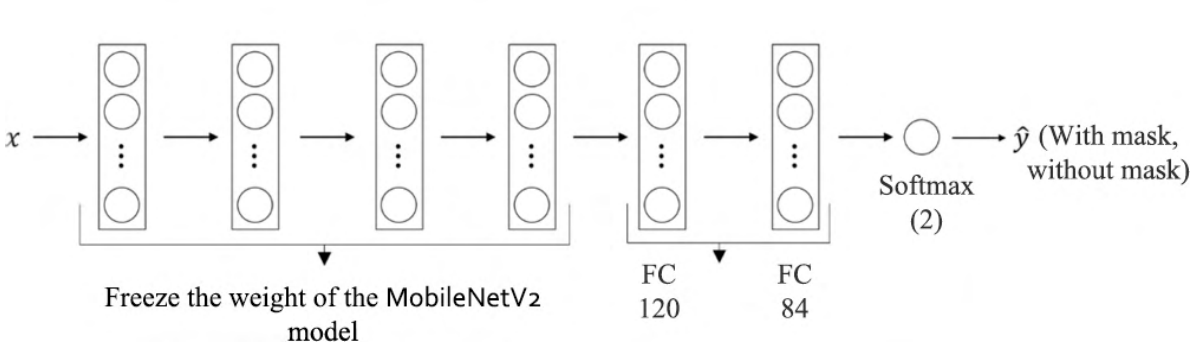


Figure 4.3: Transfer learning using MobileNetV2 model.

2. **Optimization techniques :**

- **Stochastic Gradient Descent (SGD) :**
  Stochastic gradient descent, or SGD for short, is the standard algorithm used to optimize the weights of convolutional neural network models [26]. The equation is defined as follows:

$$W = W - \alpha dW$$

- **Momentum :**
  The momentum algorithm accumulates an exponentially decaying moving average of past gradients and continues to move in their direction [9]. The effect of momentum is illustrated in figure 4.4

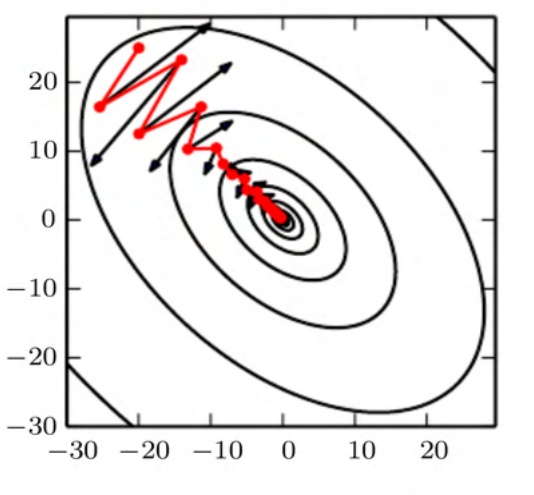


Figure 4.4: An illustration of the momentum effect. Figure reproduced from [9].

The equation is defined as follows:

$$W = W - \alpha V_{dW}$$

Such as:

$$V_{dW} = \beta V_{dW} + (1-\beta) dW$$

- **RMSProp :**
  RMSProp has been shown to be an effective and practical optimization algorithm for deep neural networks. It is currently one of the go-to optimization methods being employed routinely by deep learning practitioners [9]. RMSprop's adaptive learning rate usually prevents the learning rate decay from diminishing too slowly or too fast, it uses more memory for a given batch size than stochastic gradient descent and Momentum, but less than Adam.
  The equation is defined as follows:

$$W = W - \alpha \frac{dW}{\sqrt{S_{dW}} + \varepsilon}$$

Such as:

$$S_{dW} = \beta S_{dW} + (1-\beta) dW^2$$

- **Adam :**

  The name “Adam” derives from the phrase “adaptive moments.” In the context of the earlier algorithms, it is perhaps best seen as a variant on the combination of RMSProp and momentum with a few important distinctions [9].

  The equation is defined as follows:

$$W = W - \alpha \frac{V_{\text{corr } dW}}{\sqrt{S_{\text{corr } dW}} + \varepsilon}$$

  Such as:

$$V_{dW} = \beta_1 V_{dW} + (1 - \beta_1)\, dW$$
$$S_{dW} = \beta_2 S_{dW} + (1 - \beta_2)\, dW^2$$
$$V_{\text{corr } dW} = \frac{V_{dW}}{(1 - \beta_1)^t}$$
$$S_{\text{corr } dW} = \frac{S_{dW}}{(1 - \beta_2)^t}$$

- **Regularization for Deep Learning :**

  A central problem in machine learning is how to make an algorithm that will perform well not just on the training data, but also on new inputs. Many strategies used in machine learning are explicitly designed to reduce the test error, possibly at the expense of increased training error. These strategies are known collectively as regularization [9]. Regularization is any modification we make to a learning algorithm that is intended to reduce its generalization error but not its training error [22], it is defined as follows:

  $CostFunction = Lossfunction + Regularizationfactor$

$$J(w,b) = \frac{1}{m} \sum_{i=1}^{m} \mathscr{L}\left(\hat{y}^{(i)}, y^{(i)}\right)$$

  With this regularization term, the values of the weight matrices decrease because it assumes that a neural network with smaller weight matrices leads to simpler models. Therefore, it will also reduce overfitting. This regularization term differs in L1 and L2

  L2 Parameter Regularization

  Of the simplest and most common kinds of parameter norm penalty: the L2 parameter norm penalty commonly known as weight decay. This regularization strategy drives the weights closer to the origin, L2 regularization is also known as ridge regression or Tikhonov regularization [9].

  L2 regularization is defined as:

$$\text{Cost Function} = \text{Loss function} + \frac{\lambda}{2m} * \|w\|^2$$
$$J(w,b) = \frac{1}{m} \sum_{i=1}^{m} \mathscr{L}\left(\hat{y}^{(i)}, y^{(i)}\right) + \frac{\lambda}{2m} \|w\|_2^2$$

Such as:

$$\|w\|_2^2 = \sum_{j=1}^{n} w_j^2 = w^T w$$

The Frobenius norm is defined the L2 regularization as follows:

$$J(w,b) = \frac{1}{m}\sum_{i=1}^{m} \mathscr{L}\left(\hat{y}^{(i)}, y^{(i)}\right) + \frac{\lambda}{2m}\|w\|_F^2$$

Such as:

$$\left\|w^{(l)}\right\|_F^2 = \sum_{i=1}^{n(l-1)} \left(w_{ij}^{(l)}\right)^2$$

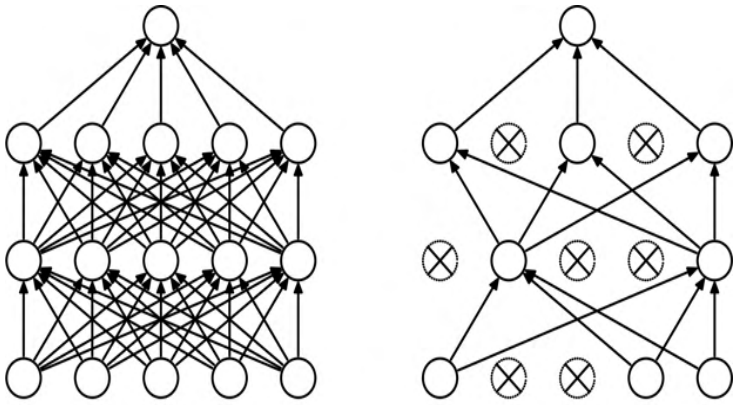
L1 Parameter Regularization

While L2 weight decay is the most common form of weight decay, there are other ways to penalize the size of the model parameters. Another option is to use L1 regularization [9].

$$\text{Cost Function} = \text{Loss function} + \frac{\lambda}{2m} * \|w\|$$

$$J(w,b) = \frac{1}{m}\sum_{i=0}^{m} \mathscr{L}\left(\hat{y}^{(i)}, y^{(i)}\right) + \frac{\lambda}{2m}\|w\|_1$$

- **Dropout :** Dropout is a technique that addresses both these issues. It prevents overfitting and provides a way of approximately combining exponentially many different neural network architectures efficiently. The term “dropout” refers to dropping out units (hidden and visible) in a neural network. By dropping a unit out, we mean temporarily removing it from the network, along with all its incoming and outgoing connections [26], as shown in Figure 4.5 The choice of



(a) Standard Neural Net (b) After applying dropout.

Figure 4.5: Dropout Neural Net Model. **Left:** A standard neural net with 2 hidden layers. **Right:** An example of a thinned net produced by applying dropout to the network on the left. Crossed units have been dropped. Figure reproduced from [26].

which units to drop is random. In the simplest case, each unit is retained with

a fixed probability p independent of other units, where p can be chosen using a validation set or can simply be set at 0.5, which seems to be close to optimal for a wide range of networks and tasks. For the input units, however, the optimal probability of retention is usually closer to 1 than to 0.5 [26].

- **Early Stopping :**
  Early stopping is a form of regularization used to avoid overfitting when training a learner with an iterative method. This mean as the number of epochs increases, the algorithm is trained more to give us better results. The cost function decreases in training, but its value increases in testing, causing overfitting, so we choose the value of Early Stopping when the test and training values are low, as shown in the following figure 4.6, the training is stopped as soon as the errors on the train set decreases as compared to the errors on the test set

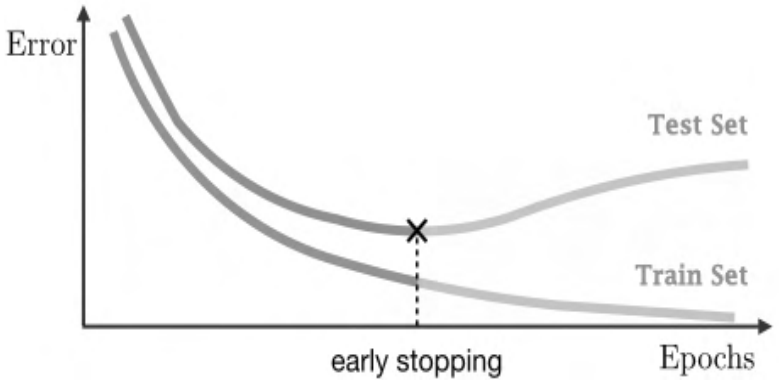


Figure 4.6: An illustration of how the early stop work .

3. **Dataset used :**
   Face Mask Detection Data set, here we have created a model that detects face mask trained on 7553 images with 3 color channels (RGB).

   The Data set consists of 7553 RGB images in 2 folders as with mask and without mask. Images are named as label with mask and without mask. Images of faces with mask are 3725 and images of faces without mask are 3828, and here is the link: [1]

4.2.1.2 YOLO model

YOLO stands for You Only Look Once. It's an object detector that uses features learned by a deep convolutional neural network to detect an object. Before we get out hands dirty with code, we must understand how YOLO works. In this thesis we will use the YOLOv3, YOLOv3 is proposed based on YOLOv2 [21], the detection speed of YOLOv2 is maintained and the detection

[1] https://www.kaggle.com/omkargurav/face-mask-dataset

accuracy is greatly improved. YOLOv3 uses the idea of the residual neural network [11]. The introduction of multiple residual network modules and the use of multiscale prediction improve the shortcomings of YOLOv2 network in small object recognition. This algorithm is one of the best algorithms in object detection because of the high accuracy and timeliness of its detection. This model uses several 3×3 and 1×1 convolution layers with good performance, and some residual network structures are also used in the subsequent multiscale prediction. This model has 53 convolution layers and can also be called Darknet-53 [17].

The YOLOv3 method divides the input image into S × S small grid cells. If the center of an object falls into a grid cell, the grid cell is responsible for detecting the object. Each grid cell predicts the position information of B bounding boxes and computes the abjectness scores corresponding to these bounding boxes. Each bounding box can be described using four descriptors: Center of the box (bx, by), Width (bw), Height (bh), Value c corresponding to the class of an object. Along with that we predict a real number pc, which is the probability that there is an object the bounding box.

And each objectness score can be obtained as follows:

$$C_i^j = P_{i,j}(\text{ Object }) * IOU_{pred}^{\text{truth}}$$

7 Whereby C_i^j is the objectness score of the jth bounding box in the ith grid cell. $P_{i,j}(\text{ Object })$ is merely a function of the object. The $IOU_{pred}^{\text{truth}}$ represents the intersection over union (IOU) between the predicted box and ground truth box [34].

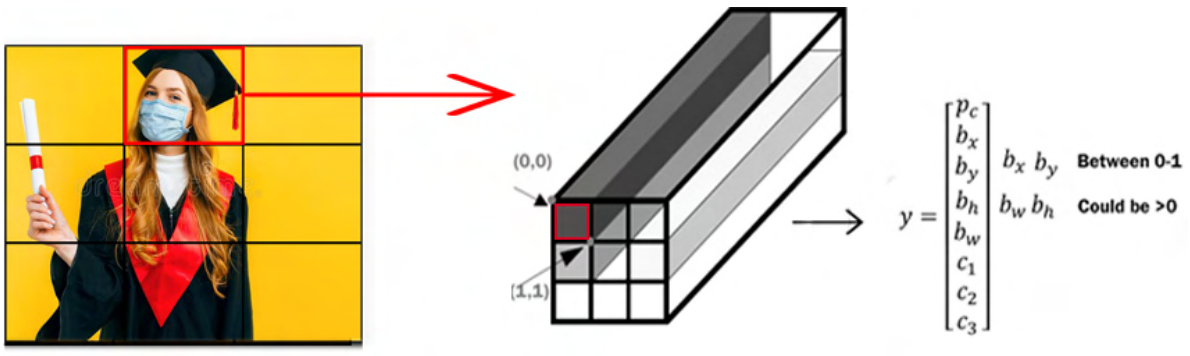


Figure 4.7: An illustration of how YOLO works.

1. **The dataset used :**

   The dataset contains 1510 images belonging to the 2 classes, as well as their bounding boxes in the YOLO format labeled text files, so the summary will be 3020 files. The classes are: with mask and without mask. The Dataset consists of hand-annotated images of with mask, and without mask with their annotations in txt format. Each

image is linked with its txt annotations. Images are annotated by free open source software LabelImg.

2. **Dataset Preparation :**

   The dataset preparation similar to How to train YOLOv3 to detect custom objects blog in medium and here is the link: [2]

Train YOLOv3 to detect custom objects :

The training was done using Google Colab so that we could get GPU for faster and efficient training of the network. After preprocessing dataset i.e. creating label file for each image, both images and their respective label files are to be kept together.

The total amount of time required to train the network with the above configurations was approximately 3-4 hours. The weights thus generated after 6000 iterations were used to carry out detections and analyzing the performance. YOLOv3 needs certain specific files to know how and what to train. We'll be creating these three files (.weight, .txt and .cfg).

The following code explains how to create the three folders (yolov3.cfg, yolov3.weight, classes.txt).

[2] `https://www.kaggle.com/techzizou/labeled-mask-dataset-yolo-darknet`.

YOLOv3 Configuration

```
!nvidia-smi
from google.colab import drive
drive.mount('/content/gdrive')
!ln -s /content/gdrive/My\ Drive/ /mydrive
!ls /mydrive
!git clone https://github.com/AlexeyAB/darknet
%cd darknet
!sed -i 's/OPENCV=0/OPENCV=1/' Makefile
!sed -i 's/GPU=0/GPU=1/' Makefile
!sed -i 's/CUDNN=0/CUDNN=1/' Makefile
!make
!cp cfg/yolov3.cfg cfg/yolov3_training.cfg
!sed -i 's/batch=1/batch=64/' cfg/yolov3_training.cfg
!sed -i 's/subdivisions=1/subdivisions=16/' cfg/yolov3_training.cfg
!sed -i 's/max_batches = 500200/max_batches = 4000/' cfg/yolov3_training.cfg
!sed -i '610 s@classes=80@classes=2@' cfg/yolov3_training.cfg
!sed -i '696 s@classes=80@classes=2@' cfg/yolov3_training.cfg
!sed -i '783 s@classes=80@classes=2@' cfg/yolov3_training.cfg
!sed -i '603 s@filters=255@filters=21@' cfg/yolov3_training.cfg
!sed -i '689 s@filters=255@filters=21@' cfg/yolov3_training.cfg
!sed -i '776 s@filters=255@filters=21@' cfg/yolov3_training.cfg
!echo -e 'with mask\nwithout mask' > data/obj.names
!echo -
e 'classes= 2\ntrain  = data/train.txt\nvalid  = data/test.txt\nnames = data/obj.names\nbackup = /mydr
ive/yolov3' > data/obj.data
!cp cfg/yolov3_training.cfg /mydrive/YOLOV3/yolov3_testing.cfg
!cp data/obj.names /mydrive/YOLOV3/classes.txt
!mkdir data/obj
!unzip /mydrive/YOLOV3/images.zip -d data/obj
import glob
images_list = glob.glob("data/obj/*.jpg")
with open("data/train.txt", "w") as f:
f.write("\n".join(images_list))
!wget https://pjreddie.com/media/files/darknet53.conv.74
!./darknet detector train data/obj.data cfg/yolov3_training.cfg darknet53.conv.74 -dont_show
```

The modifications will be according to the number of classes and will be as follow:

Line 13: set $batch = 64$, this means we will be using 64 images for every training step

Line 14: set $subdivisions = 16$, the batch will be divided by 16 to decrease GPU VRAM requirements.

Line 603: set $filters = (classes + 5) * 3$, in our case $filters = 21$

Line 610: set $classes = 2$, the number of categories we want to detect

Line 689: set $filters = (classes + 5) * 3$, in our case $filters = 21$

Line 696: set $classes = 2$, the number of categories we want to detect **Line 776:** set $filters = (classes + 5) * 3$, in our case $filters = 21$ **Line 783:** set $classes = 2$, the number of categories we want to detect

4.3 Social distancing detection

Social distancing associates with the measures that overcome the virus' spread, by minimizing the physical contacts of humans, such as the masses at public places (e.g., shopping malls, parks, schools, universities, airports, workplaces)

And for reduce risk of serious illness from COVID-19 we use the MobileNet-SSD model to detect the people, after human detection, the Euclidean distance between each detected centroid pair is computed using the detected bounding box and its centroid information. Simply, a centroid is the center of a bounding box. A predefined minimum social distance violation threshold is specified using pixel to distance assumptions

Figure 4.8: The concept of the social distancing detection service.

4.3.1 Object detection

4.3.1.1 MobileNet SSD model

MobileNet for Single Shot Multi-Box Detector (SSD) model. This algorithm is used for real-time detection, and for webcam feed to detect the purpose webcam which detects the object in a video stream, in our model, we use it to find out the distance between people.

Therefore, we use an object detection module that can detect what is in the video stream.

The name Single Shot means that, like in YOLO, the tasks of object localization and classification are done in a single forward pass of the network, simultaneously predicting the bounding box and the class as it processes the image, MobileNet-SSD based on the VGG-16 classifier.

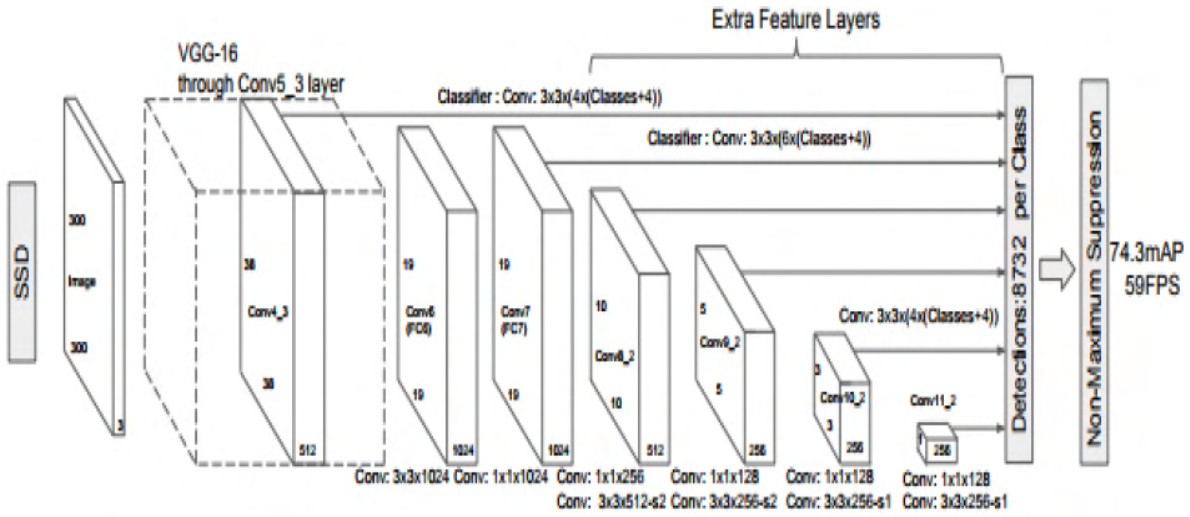


Figure 4.9: SSD proposed by Liu et al. Figure reproduced from [4].

In Figure 4.9, we utilize the Visual Geometry Group (VGG) – 16 layers up until $conv_6$ and then detach all other layers, including the fully-connected layers.
The reason that the $VGG-16$ was used as the base network is because of its strong performance in high quality image classification tasks and its popularity for problems where transfer learning helps in improving results. A set of new CONV layers are then added to the architecture, these are the layers that make the SSD framework possible [4].

- **Implementation of MobileNet SSD model :**

  SSD generated .caffemodel is the base model for MobileNet-SSD. The MobileNet architecture (stored as MobileNet.prototxt) used to cross train the SSD's .caffemodel by the MobileNet architecture. The cross training process is more time consuming than a fresh training of SSD network. The newly generated .caffemodel is now used for object detection.

4.3.1.2 YOLOv3 model

YOLOv3 is used for human detection as it improves predictive accuracy, particularly for small-scale objects. YOLO treats object detection as a regression problem, taking a given input image and simultaneously learning bounding box coordinates and corresponding class label probabilities, it is used to return the person prediction probability, bounding box coordinates for the detection, we loaded the configuration and weight from Darknet and for the classes we only use the person class, and the centroid return values are used to calculate the distance between people

NMS (Non-maxima suppression) is also used to reduce overlapping bounding boxes to only a single bounding box, thus representing the true detection of the object. Having overlapping boxes is not exactly practical and ideal, especially if we need to count the number of objects in an image.

4.4 classification

We applied classification methods, which are alexNet and VGG-16, for the mask detection we trained our classifier models with the same dataset used with the mobileNet model.

4.4.1 Dataset for social distancing

The dataset contains two folders:

- Two Meter Distance: This folder contains 2 video recordings with corresponding annotations for at-least 2 subjects in each video frame who are standing at fixed distance of exactly 2 meters

  Video 1: Top View (aerial perspective) Ground truth frames: This folder contains 1917 images (.png). Each image shows a rectangular box indicating the selected persons within the image who are standing 2-meter apart from each other. Video2: Perspective/Horizontal View Ground truth frames: This folder contains 379 images (.png). Each image shows a rectangular box indicating the selected persons within the image who are standing 2-meter apart from each other.

- One Meter Distance: This folder contains one video recordings of perspective view with corresponding annotations for at-least 2 subjects in each video frame who are standing at fixed distance of exactly 1 meter Ground truth frames: This folder contains 1689 images (.png). Each image shows a rectangular box indicating the selected persons within the image who are standing 1-meter apart from each other.

  and here is the link of the dataset: [3]

4.4.2 VGG-16 Model

VGG-16 Architecture

[3] https://data.mendeley.com/datasets/xh6m6gxhvj/1?fbclid=IwAR0_kp5yJeNMMCp7Ydqv6SkbGGvKfY6Avpkh1ql1pIl4t_UxWGv-vaK57KA

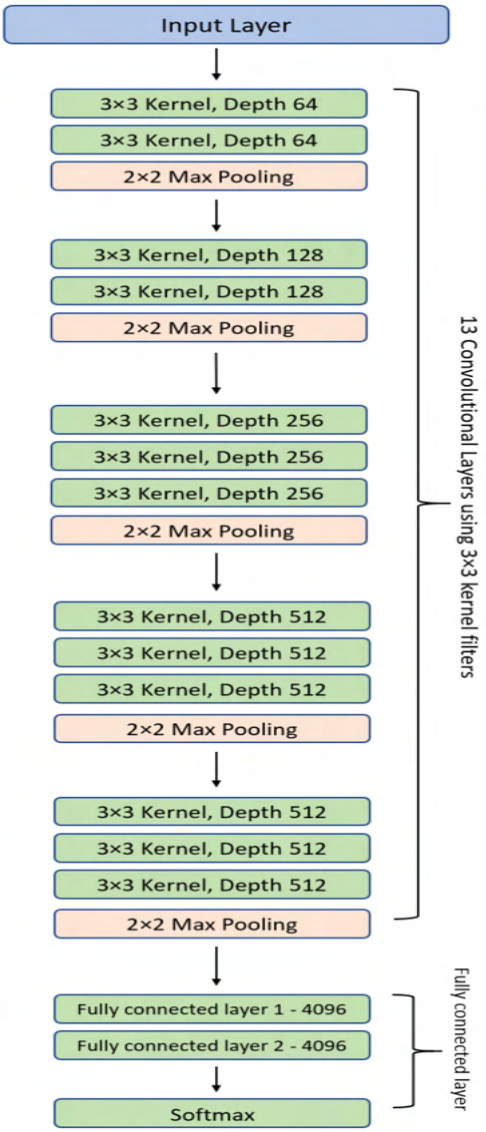


Figure 4.10: VGG-16 model architecture – 13 convolutional layers and 2 Fully connected layers and 1 SoftMax.. Figure reproduced from [28].

The precise structure of the VGG-16 network shown in Figure 4.10 is as follows:

- The first and second convolutional layers are comprised of 64 feature kernel filters and size of the filter is 3×3. As input image (RGB image with depth 3) passed into first and second convolutional layer, dimensions changes to 224x224x64. Then the resulting output is passed to max pooling layer with a stride of 2.
- The third and fourth convolutional layers are of 124 feature kernel filters and size of filter is 3×3. These two layers are followed by a max pooling layer with stride 2 and the resulting output will be reduced to 56x56x128.

- The fifth, sixth and seventh layers are convolutional layers with kernel size 3×3. All three use 256 feature maps. These layers are followed by a max pooling layer with stride 2.

- Eighth to thirteen are two sets of convolutional layers with kernel size 3×3. All these sets of convolutional layers have 512 kernel filters. These layers are followed by max pooling layer with stride of 1.

- Fourteen and fifteen layers are fully connected hidden layers of 4096 units followed by a softmax output layer (Sixteenth layer) of one unit [28].

4.4.3 AlexNet model

AlexNet Architecture The precise structure of the VGG-16 network shown in Figure 4.11 is as

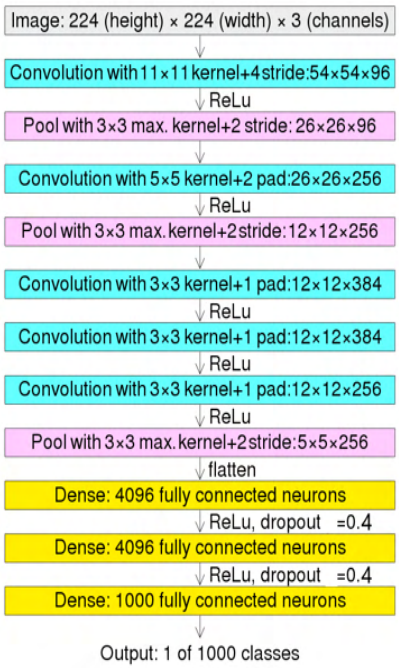


Figure 4.11: AlexNet model architecture.

follows:

- The first convolutional layers are comprised of 96 feature kernel filters and size of the filter is 11×11. As input image (RGB image with depth 3) passed into first convolutional layer, dimensions changes to 224x224x3. Then the resulting output is passed to max pooling layer with a stride of 2.

- The second convolutional layers are of 256 feature kernel filters and size of filter is 5×5. These two layers are followed by a max pooling layer with stride 2 and the resulting output will be reduced to 12x12x256.
- The Third fourth, layers are convolutional layers with kernel size 3×3. All three use 384 feature maps.
- The fifth convolutional layers are comprised of 256 feature kernel filters and size of the filter is 3×3, this layer is followed by a max pooling layer with padding 1
- Sixth, seventh and eighth layers are fully connected hidden layers of 4096 units followed by a softmax output layer of 1000 units.

4.5 Conclusion

Designing the model structure is important to improve classification and object detection results, but other factors such as datasets and hyperparameters play a very important role in optimizing our models.

In this chapter we have highlighted a transfer learning technique where transfer learning is used to improve the generalisation ability of the mode, and helps to improving results and to speed up the neural network.

CHAPTER 5

RESULT, DISCUSSION AND EXPERIMENTATION

'Deep learning takes years, nothing can happen in seconds."

Contrary to the previous chapter where the focus was on the theoretical description of our proposed system in positioning our contribution, this chapter presents the results achieved to assess the effectiveness of our solution to the issue of mask detection and social distancing by explaining at the same time the various tools used for the practical part of our work in order to reach our objectives. Experimental research has been carried out on real data freely available on the web. For the validation of the results, we used a set of supervised measures such as: recall, precision, f-measure, accuracy, loss function. To enhance the results obtained, a comparative study between object detection, object tracking and classification in terms of accuracy and loss also number of parameters, but before validating our models, we did a series of experiments to choose the best hyper-parameters and optimizers in order to improve the performance of the model.

5.1 Implementation tools

To implement and optimize the models offered, we now have easy-to-use open source deep learning framework that aim to simplify the implementation of complex and large-scale models.

5.1.1 TensorFlow

TensorFlow is an open-source end-to-end platform for creating Machine Learning applications. It is a symbolic math library that uses dataflow and differentiable programming to perform various tasks focused on training and inference of deep neural networks. It allows developers to create machine learning applications using various tools, libraries, and community resources, it was

built to run on multiple CPUs or GPUs and even mobile operating systems, and it has several wrappers in several languages like Python, C++ or Java. Let us see top uses of TensorFlow to understand TensorFlow applications

- Video Detection
- Image Recognition
- Voice Recognition
- Text-based applications

5.1.2 Kears

Keras is an API designed for human beings, not machines. Keras follows best practices for reducing cognitive load: it offers consistent and simple APIs, it minimizes the number of user actions required for common use cases, and it provides clear and actionable error messages. It also has extensive documentation and developer guides. Keras has several architectures, mentioned below, to solve a wide variety of problems (e.g. classification of images).

- Mobilenet
- VGG16
- VGG19...etc

5.1.3 Sklearn

Scikit-learn is a library in Python that provides many unsupervised and supervised learning algorithms. It's built upon some of the technology you might already be familiar with, like NumPy, pandas, and Matplotlib.

The functionality that scikit-learn provides include:

- Regression, including Linear and Logistic Regression
- Classification, including K-Nearest Neighbors
- Clustering, including K-Means and K-Means++
- Model selection
- Preprocessing, including Min-Max Normalization

5.2 Programming language used

5.2.1 Python

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

5.3 Evaluation measures

5.3.1 Confusion matrix

	actual positive	actual negative
predicted positive	TP	FP
predicted negative	FN	TN

Figure 5.1: Confusion matrix. Figure reproduced from [6]

To fully understand how a confusion matrix works, it is important to understand the four main terminologies: TP, TN, FP and FN. Here is the precise definition of each of these terms:

- Regression, including Linear and Logistic Regression
- TP (True Positives): cases where the prediction is positive, and where the real value is indeed positive.
- TN (True Negatives): cases where the prediction is negative, and where the real value is indeed negative.
- FP (False Positive): cases where the prediction is positive, but the real value is negative.
- FN (False Negative): cases where the prediction is negative, but the real value is positive.

5.3.2 Precision

Precision is the ratio of correctly predicted positive observations to the total predicted positive observations.

$$Precision = \frac{TP}{(TP + FP)}$$

5.3.3 Recall

Recall is the rate of true positives, that is to say the proportion of positives that we have correctly identified.

$$Recall = \frac{TP}{(TP + FN)}$$

5.3.4 F1 score

F1 Score is the weighted average of Precision and Recall.

$$F1 = 2x\frac{(PrecisionxRecall)}{(precision + Recall)}$$

5.4 Experiments and implementation

5.4.1 Mask detection (MobileNetV2)

We have done several experiments with different batch size, learning rate and optimizers, we also apply the data augmentation .

1. **Optimizers**

   Here we have the implementation with three types of optimizers Adam, RMSprop and SGD.

```
from tensorflow.keras.optimizers import Adam,SGD,RMSprop
optimizer = ['SGD', 'RMSprop', 'Adam']
for i in range(3):
    print("Model using",optimizer[i],"optimizer")
    model.compile(loss="binary_crossentropy", optimizer=optimizer[i],
    metrics=["accuracy"])
```

   In the following figure we have the experimentation with three types of optimizers Adam, RMSprop and SGD.

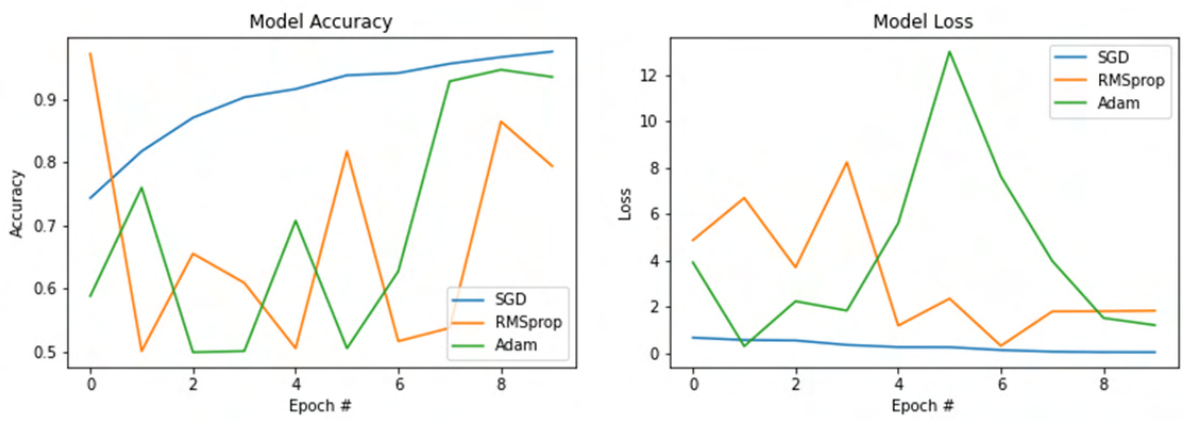


Figure 5.2: The accuracy and the loss function with different optimizers.

The SGD optimizer always gives good results but with a long time to calculate compare to Adam this is why Adam is the most used in the literature especially in computer vision problems.

2. **Learning Rate**

Here we have the implementation of the different values of Learning Rate

```
lr=0.0001
for i in range(5):
    print("Model using learning rate of",lr)
    opt = Adam(lr, decay=lr / EPOCHS)
    lr = lr + 0.0001
    model.compile(loss="binary_crossentropy", optimizer=opt,
    metrics=["accuracy"])
```

And here we have the experimentation of the different values of Learning Rate

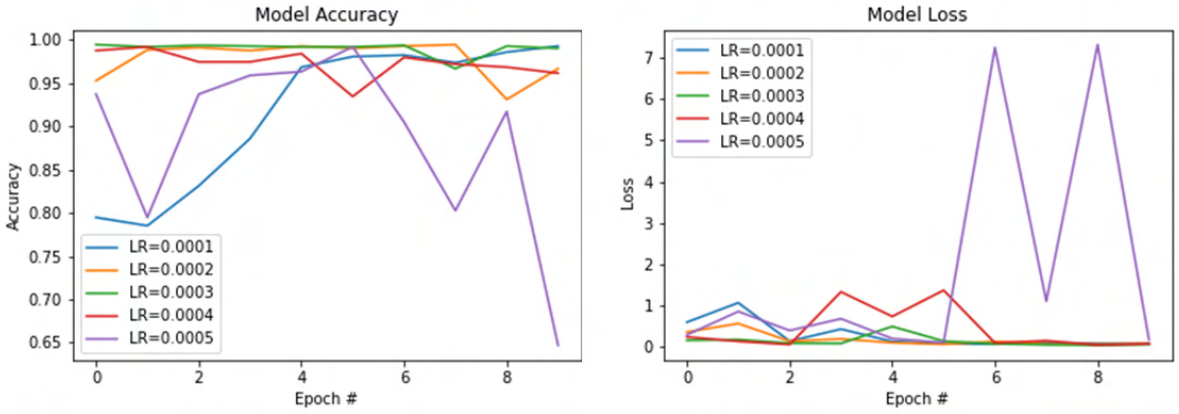


Figure 5.3: The accuracy and the loss function with different values of learning rate.

The learning rate is a very important parameter that is explained in the results of the curves. The learning rate controls how quickly the model is adapted to the problem. Smaller learning rates require more training epochs given the smaller changes made to the weights each update, whereas larger learning rates result in rapid changes and require fewer training epochs it is for this reason the choice of the learning rate is linked to the choice of the number of epoch and the batch size. For results with learning rate = 0.0004 and 0.0005 we notice an instability of the results validated by an instability of back propagated loss signals because they don't get a clear static picture of the loss.

3. **Batch Size**

Here we have the implementation of the different values of Batch Size

```
BS =[32,64,128]
lr=0.0003
for i in range(3):
    print("Model using",BS[i],"BS")
    opt = Adam(lr, decay=lr / EPOCHS)
    model.compile(loss="binary_crossentropy", optimizer=opt,
    metrics=["accuracy"])
```

And here we have the experimentation of batch size with a different values Batch Size

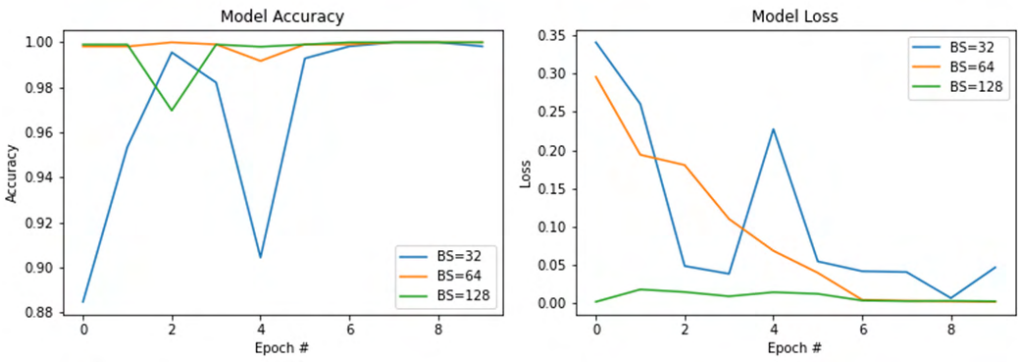


Figure 5.4: The accuracy and the loss function with different batch size.

After the comparison, the batch-sise with the value 128 gives the best result

4. **Data augmentation**

Here we have the implementation of data augmentation technology that we apply many techniques which are rotation, Flip Augmentation and here we have two type(Horizontal Flip Augmentation, Vertical Flip Augmentation) and the last one is Shift Augmentation and here We have two types which are horizontal shift augmentation and vertical shift augmentation

```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
aug = ImageDataGenerator(
  rotation_range=20,
  zoom_range=0.15,
  width_shift_range=0.2,
  height_shift_range=0.2,
  shear_range=0.15,
  horizontal_flip=True,
  fill_mode="nearest")
```

We have done data augmentation to improve model prediction accuracy as well as add more training data into the models, moreover, to prevent data scarcity for better models, and To emphasize the importance and necessity of using data augmentation, we tested our model with and without data augmentation

With data augmentation

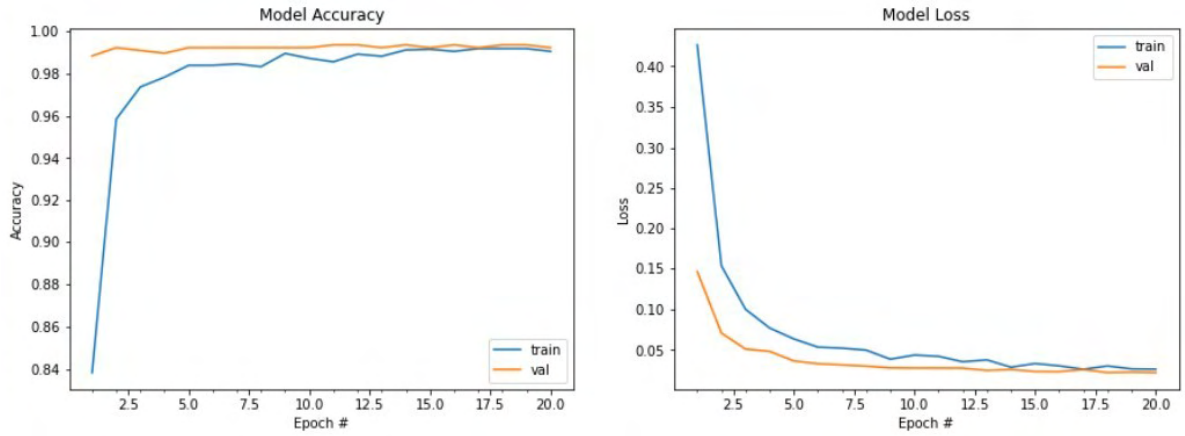


Figure 5.5: The accuracy and the loss function with data augmentation.

Overfitting is caused by having too little data. To solve this problem we have used the data augmentation Since in our dataset we have too few different use cases which makes it difficult to effectively train our deep learning model. Consequently, it cannot develop decision rules that can be generalized. The idea behind Data Augmentation is to reproduce pre-existing data by applying a random transformation to it. For example, applying a mirror effect to an image. The model is therefore exposed to more data. This allows him to generalize better.

Without data augmentation

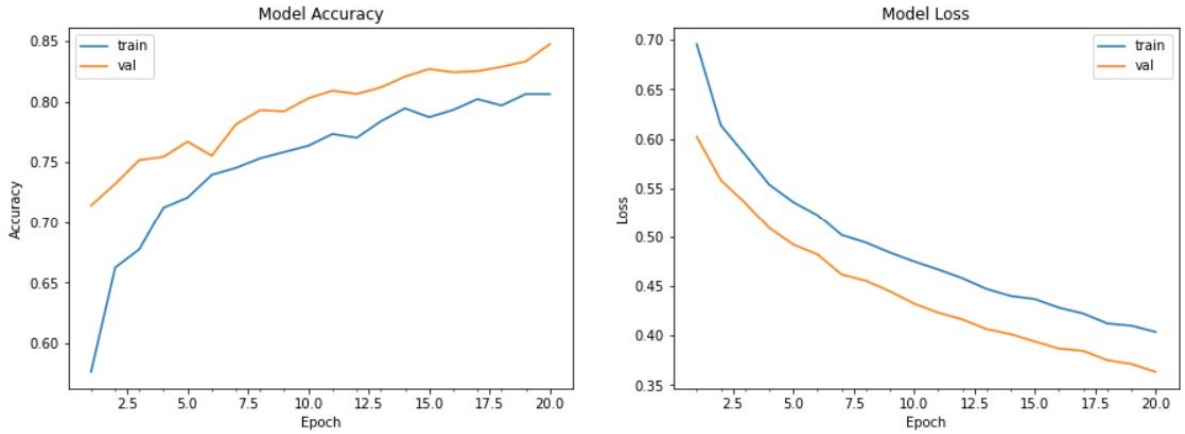


Figure 5.6: The accuracy and the loss function without data augmentation.

As you knew with Data Augmentation we cannot produce new information, we can only rework, remix existing information. This may not be enough to get rid of overfitting

completely, which is why we used Dropout. after several tests and an efficient setting of parameters in terms of learning rate, number of batch size epoche we have obtained with considerable results up to 0.99 success rate this is due to the lack of variation of the images especially after the use of data augmentation. We note that with the use of data increased the results were improved, this is explained by the fact that deep learning requires a lot of learning data. The curve of the learning phase and the test phase are close with the use of data augmentation explained by the absence of overfiting compare to results without the use of data augmentation.

5.4.1.1 Final result

Curve 5.7 represents the precision and the loss function for our model after the experimentation

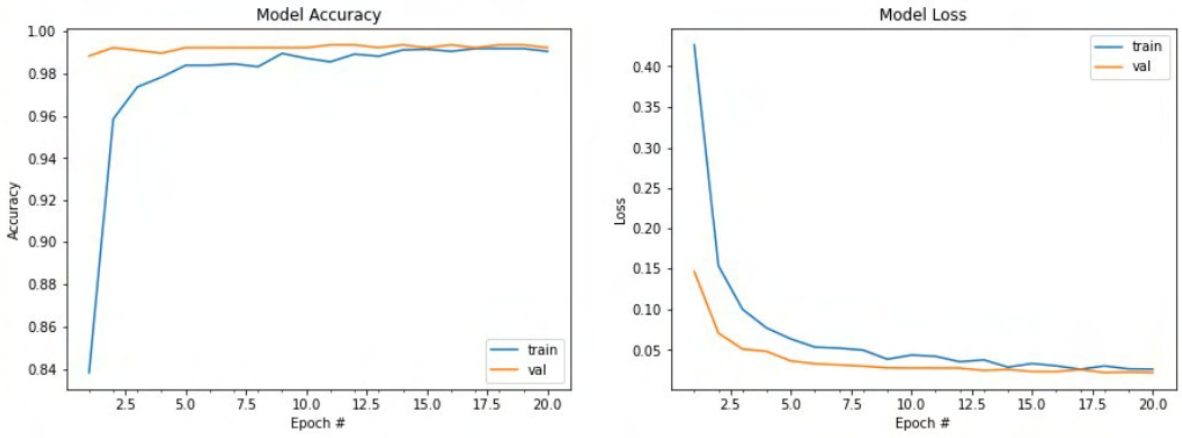


Figure 5.7: Validation results through The accuracy and the loss function.

we obtain the result in table 5.1

Accuracy	Loss function
0.99	0.01

Table 5.1: The accuracy and loss function after the experimentation.

5.4.1.2 Final configuration

After experiments, we come to the parameters configuration in Table 5.2

Batch Size	Optimizers	Learning Rate7	number of Epochs	model
128	Adam	0.0003	20	MobileNetV2

Table 5.2: The parameters selected for our model.

5.4.1.3 output Result

1. **Video output**

   Regarding the video, we got the following result

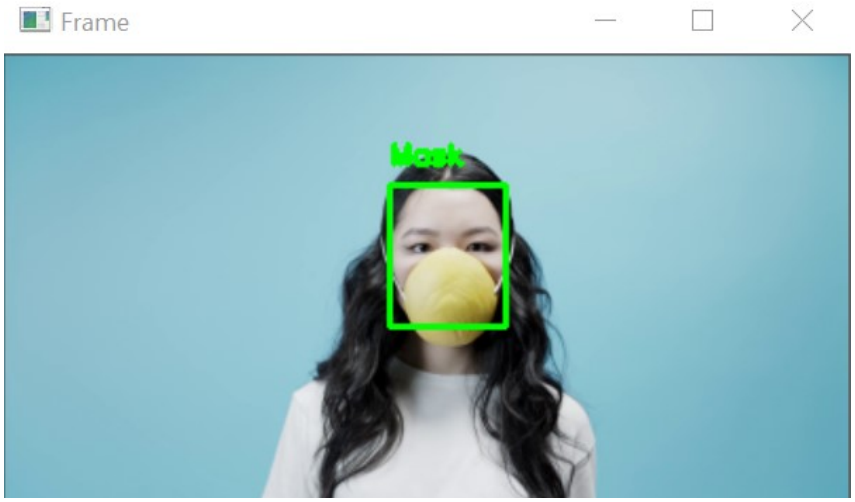


Figure 5.8: Video output for the detection mask.

Deep separable convolution is introduced, which greatly reduces the complexity cost and the size of the network model, which is suitable for mobile devices or any device with low computing power. We used the residual structure which helped to avoid overfiting. we have adapted the MobileNetV2 to our problem because it is validated that it gives good results vis-a-vis the problem of object detection and it does not require a lot of time to calculate so that we can add it in raspberry pi or microprocessors at low cost. using the bottleneck activation function has improved results compared to MobileNetV2.

2. **Real-time output**

   We tested our model in real time and got a very satisfactory result

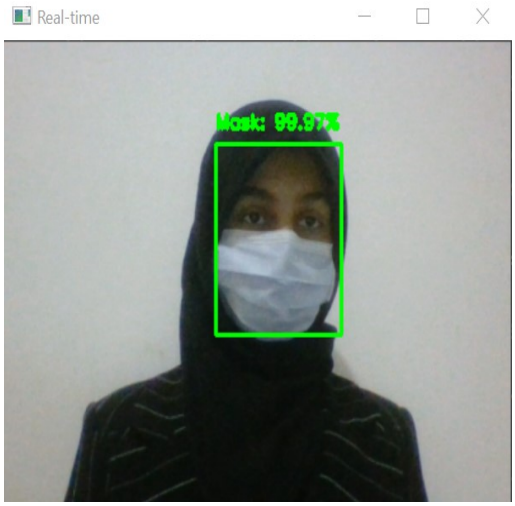


Figure 5.9: Real-time output for the mask detection.

5.4.2 Mask detection (YOLOV3)

In the previous chapter we mentioned about YOLOV3 and how we modified the configuration by number of classes and how we changed the weight with the dataset that annotated in text format by the free open source software LabelImg. And now we are just testing our model for an images and a videos.

1. **Image output**

Figure 5.10: Image output for the mask detection.

2. **Video output**

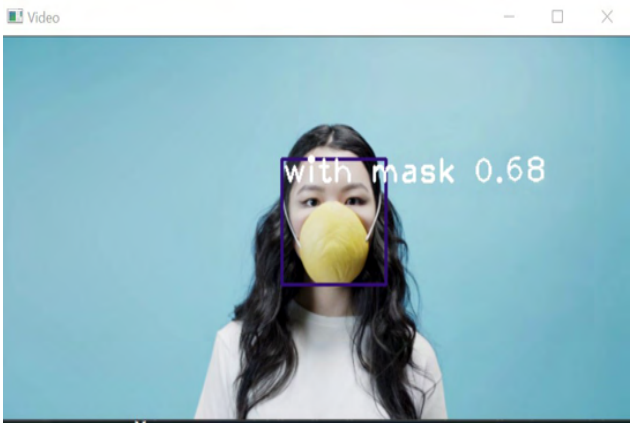


Figure 5.11: Video output for mask detection.

5.4.2.1 Comparison between YOLOV3 and MobileNetV2

The YOLO model predicts the confidence score for an object and predicts the probability of each class given that there is an object existing next, whereas the MobileNet network does not predict the confidence score for each object, but it gives the probability that a class is present in a given bounding box. The results demonstrated that YOLOv3 is the accurate but slowest object detection system while MobileNetV2 is the fastest one with the highest accuracy. YOLOv3 is the best one since it detects the objects with the highest accuracy, but for video-streams we notice that the accuracy decreases, Since there is a trade-off between accuracy and speed in all these systems, the most appropriate system for us is the one that combines both, accuracy and speed. So we conclude that MobileNetV2 is the best way with highest accuracy and fastest speed that can be achieved in detecting objects in images and video also in real time.

5.4.3 Mask detection (Classification)

We have in classification two models, VGG16 and AlexNet, where we depended on the comparison between them in accuracy and loss function, and all this is shown in Figure 5.12, 5.13

1. **VGG-16**

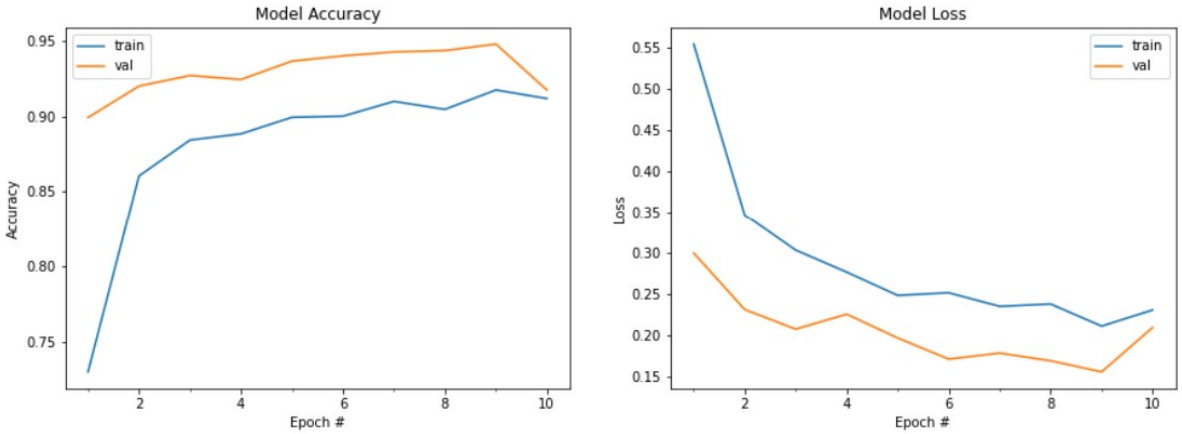


Figure 5.12: Accuracy and loss function for VGG-16 Model.

2. **AlexNet**

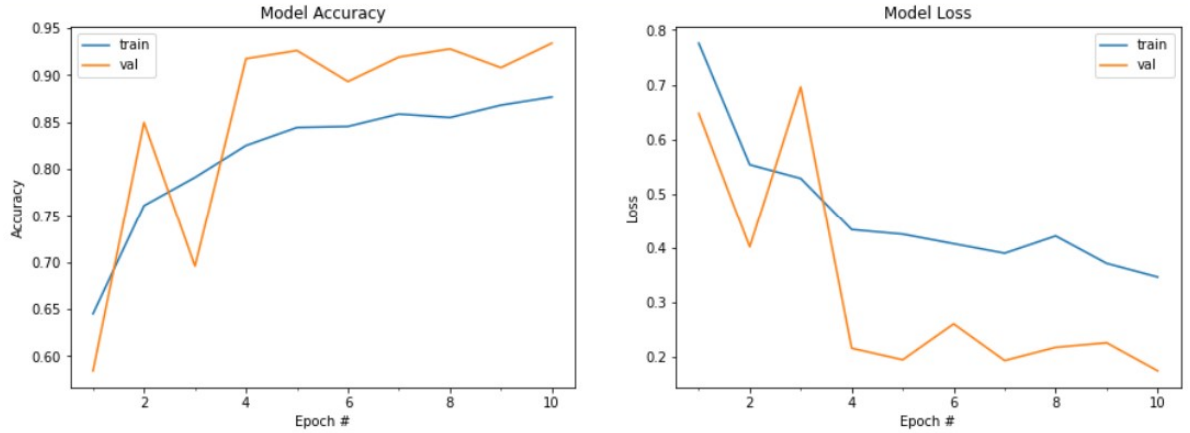


Figure 5.13: Accuracy and loss function for AlexNet Model.

5.4.3.1 output result for the two models VGG-16 and AlexNet

Figure 5.14: Output result for VGG-16 and AlexNet.

5.4.3.2 Comparison between VGG-16 and AlexNet

In this section, we attempt to compare the prediction processes of VGG-16 with AlexNet through the Figures. Within the scope of this figures, the performance of two highly acknowledged pretrained networks, namely,VGG-16 and AlexNet and the both of them converge quickly by scoring over 0,90 accuracy However, we notice a slight difference, which is that VGG-16 is superior to AlexNet in accuracy, but at the same time it takes a very long time to train than AlexNet, and this due to the numbers of layers and parameters. The following table 5.3 shows the performance of each model. The figures 5.12, 5.13 shows the accuracy and loss function for VGG16 and AlexNet

	Precision	Recall	F1 score	number of parameters
VGG-16	0.94	0.94	0.94	134,268,738
AlexNet	0.89	0.89	0.89	28,081,754

Table 5.3: Performance measures of VGG-16 and AlexNet of mask detection.

5.4.4 Social distancing detection(YOLOV3)

5.4.4.1 Real time output

In the screenshot below, we display the output real time of YOLOv3

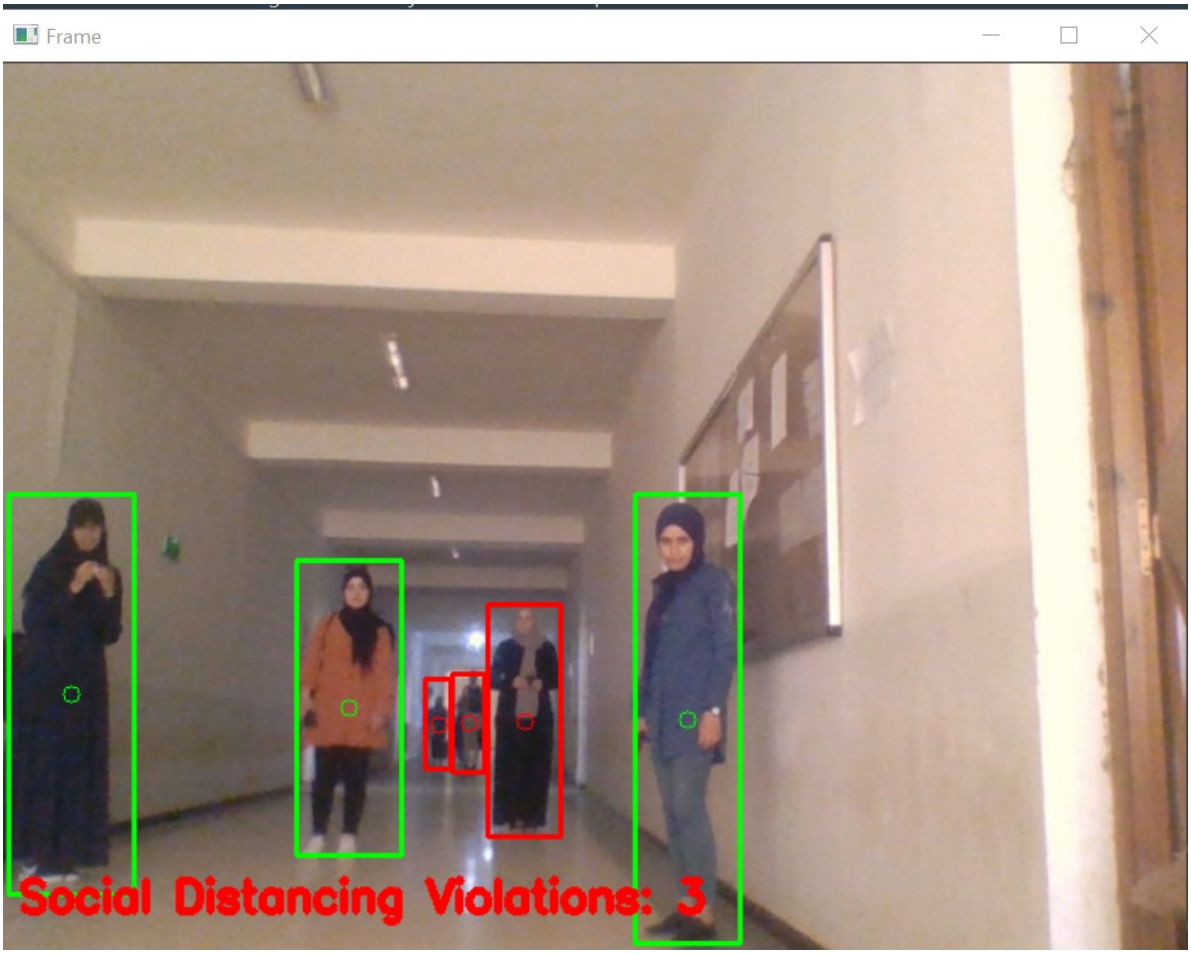


Figure 5.15: Real time output of YOLOV3 for social distancing detection.

5.4.4.2 Video output

In the screenshot below, we display the output video of YOLOv3

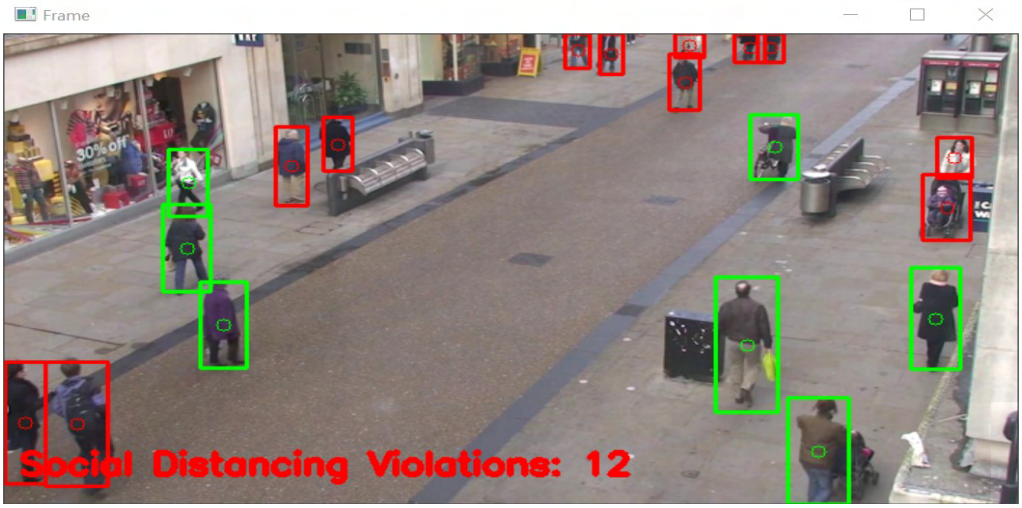


Figure 5.16: Video output of social distancing detection for YOLOV3.

5.4.5 Social distancing detection(MobilNet-SSD)

5.4.5.1 Video output

In the screenshot below, we display the output video of MobilNet-SSD

Figure 5.17: Video output of social distancing detection for MobilNet-SSD.

5.4.5.2 Comparison between YOLOV3 and MobileNet-SSD

According to our implementation results of one image and video, and real time object detection and as we talked about YOLOV3 in mask detection, the result of social distancing discovery gives

us the same result. Although YOLOV3 is the best model in object detection, it is only associated with images, but for real-time and video, MobileNet-SSD provides the best model because it is the fastest. YOLO has better performance in accuracy while MobileNet-SSD has a better performance in speed, in our project we foucs in accuracy and speed, since MobilNet-SSD provides good result, it is recommended to apply it.

5.4.6 Social distancing detection(Classification)

We have in classification two models, VGG16 and AlexNet, where we depended on the comparison between them in accuracy and loss function, and all this is shown in Figure 5.18, 5.19

1. **VGG-16**

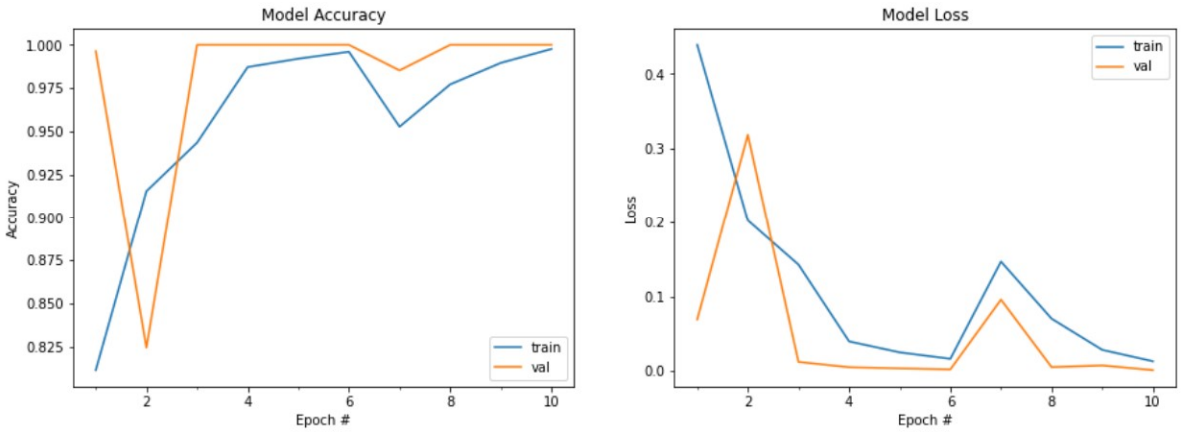


Figure 5.18: Accuracy and loss function for VGG-16 Model.

2. **AlexNet**

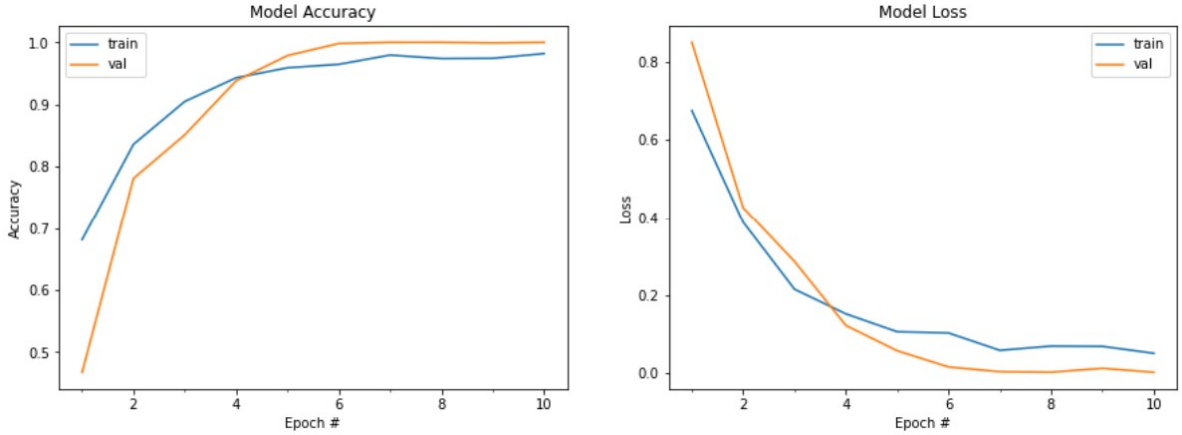


Figure 5.19: Accuracy and loss function for AlexNet Model.

5.4.6.1 Comparaison between VGG-16 and AlexNet

According to the figures 5.19, 5.18 and the evaluation metrics and as we talked in section of mask detection that the both of classifiers are highly acknowledged pretrained networks and give us a good result and for confirm this we trainned social distancing dataset with classic models(VGG-16, AlexNet) and we get the following table 5.4 which confirm the correctness of what we have found.

	Precision	Recall	F1 score	number of parameters
VGG-16	0.96	0.97	0.96	134,268,738
AlexNet	0.91	0.9	0.9	28,081,754

Table 5.4: Performance measures of VGG-16 and AlexNet of social distancing detection.

5.4.6.2 Output result for the two models VGG-16 and AlexNet

This result is for both models which are VGG-16 and AlexNet, and both give a correct prediction

Figure 5.20: Output result for VGG-16 and AlexNet.

5.5 Understanding Convolutional Neural Networks (CNNs) using Visualization

In this technique, given an input image, we will simply plot what each filter has extracted (output features) after a convolution operation in each layer. To understand Convolutional Neural Networks (CNNs) using visualization, we will take an example of mask detection using the VGG-16 model, the dimensions of the input layer are 224x224x3 and the results will be as follows:

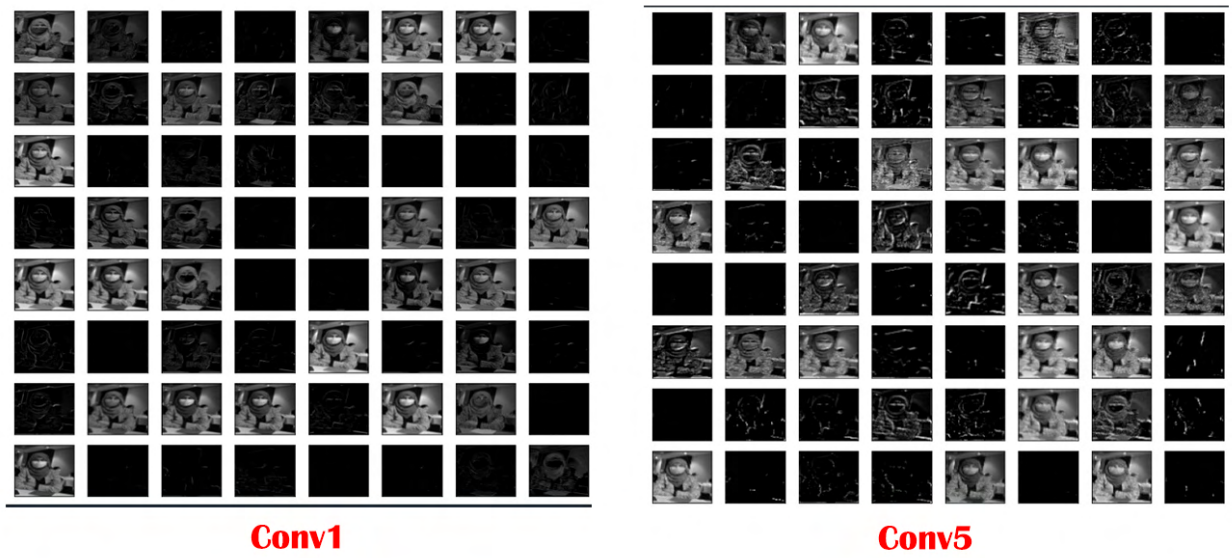


Figure 5.21: Visualizing the output of convolution operation after layers 1, 5 of VGG-16 network.

The initial layers (1, 2, 3, 4, 5) retain most of the input image features. It looks like the convolution filters are activated at every part of the input image. This gives us an intuition that these initial filters might be primitive edge detectors (since we can consider a complex figure to be made up of small edges, with different orientations, put together.).

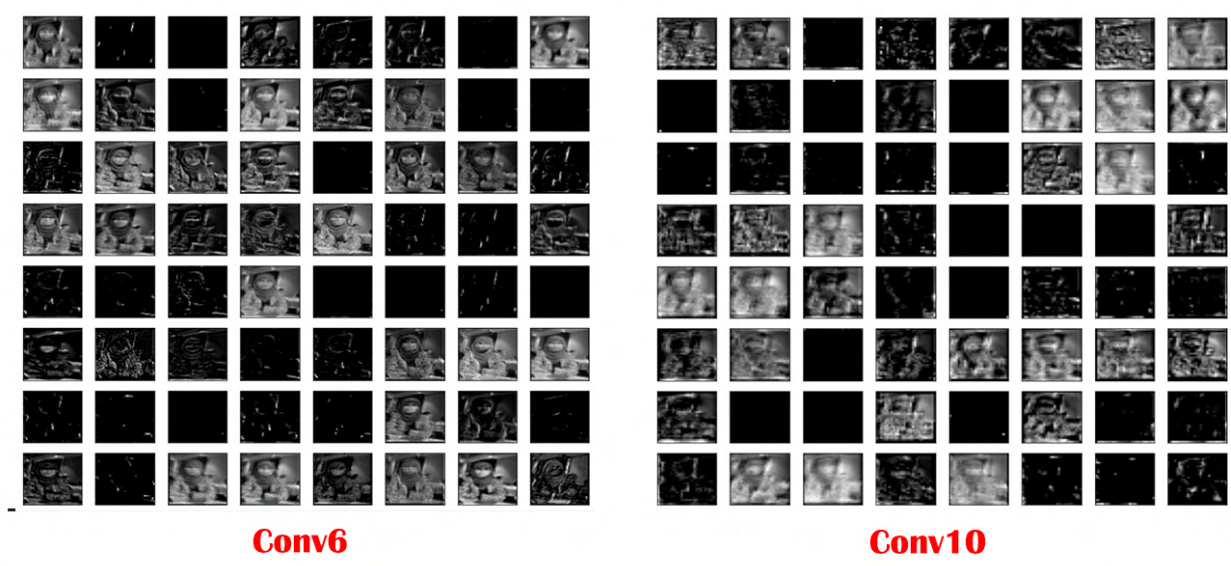


Figure 5.22: Visualizing the output of convolution operation after layers 6, 10 of VGG-16 network.

As we go deeper in layers (6, 7, 8, 9, 10) the features extracted by the filters become visually less interpretable. An intuition for this can be that the convnet is now abstracting away visual information of the input image and trying to convert it to the required output classification domain.

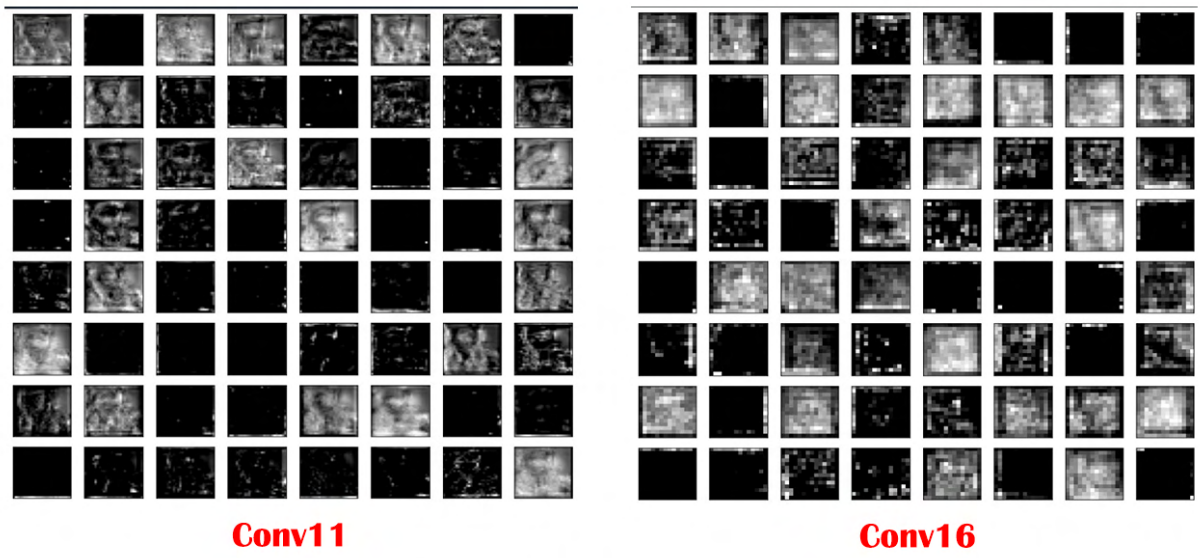


Figure 5.23: Visualizing the output of convolution operation after layers 11, 16 of VGG-16 network.

In layers (11, 12, 13, 14, 15, 16) we see a lot of blank convolution outputs. This means that the pattern encoded by the filters were not found in the input image. Most probably, these patterns must be complex shapes that are not present in this input image.

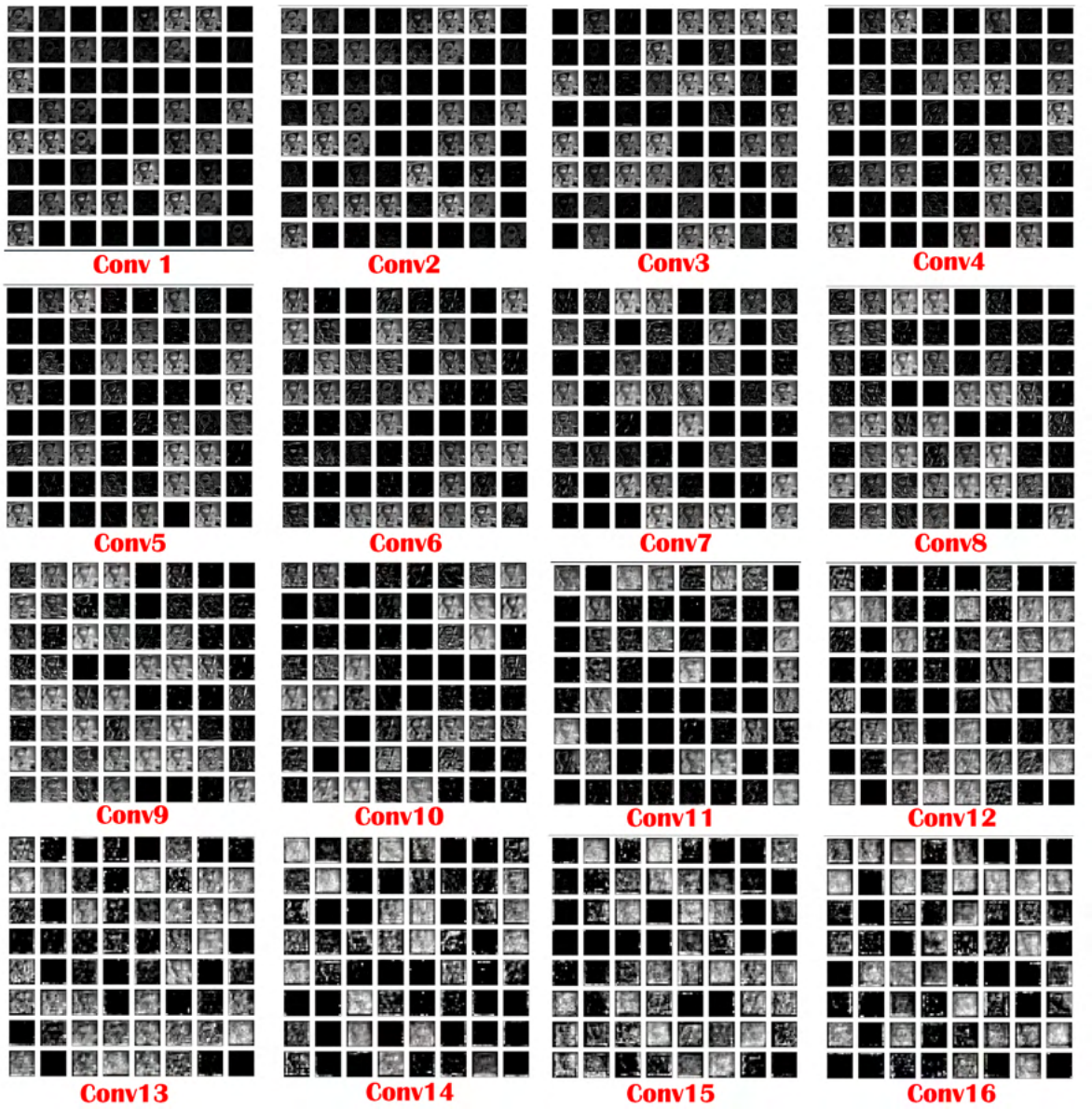


Figure 5.24: Visualizing the output of convolution operation after each layer of VGG16 network.

Running the example results in 16 plots showing the features output from the input image of the VGG16 model.

We can see that the features output of the model capture a lot of fine detail in the image and that as we progress deeper into the model, the features output show less and less detail. This pattern was to be expected, as the model abstracts the features from the image into more general concepts that can be used to make a classification. Although it is not clear from the final image that the model saw a person with mask, we generally lose the ability to interpret these features output.

5.6 Comparison of models in terms of the number of parameters

In the table below 5.5, we have a comparison of the following models in terms of the number of parameters

	Trainable parameters	Non-trainable parameters	Total parameters
VGG-16	134,268,738	0	134,268,738
AlexNet	28,060,618	21,136	28,081,754
YOLOV3	61,949,149	52,608	62,001,757
MobileNetV2	2,388,098	34,112	2,422,210
MobileNet-SSD	2,236,480	31,616	2,268,096

Table 5.5: Comparison of models in terms of the number of parameters.

5.7 Conclusion

In this chapter we have detailed the evaluation measures used in our work, Discuss and interpret the results obtained, Detail the stages of implementation of our work, Carry out a comparative study between the models of object detection, moreover, we did a comparative between the classical models of classification.

General conclusion

This thesis was intended to explore a field of research that fascinates many researchers, and above all to be ahead of a trend that is COVID-19 and Artificial Intelligence. To carry out this work, we used convolutional Neural Network as a method, and this choice of method is justified by the simplicity and efficiency of its methods. We had to first define the concept of deep learning and find out how to apply it to our research topic. We covered two topics, Mask Detection and Social Distancing detection, each using different techniques that are object detection and tracing as well as classification and for each we used a different models. We focus on transfer learning technology to accelerate neural network formation and, in addition, we have used data augmentation to prevent data scarcity for better models. The results we obtained confirm the effectiveness of our approach. This work remains open to further improvements that we can cite as perspectives in this area for the future.

Future work:

Following our findings during this thesis, some new research doors were opened which are detailed below:

When we talk about mask detection, we mean that the camera will detect people who wear masks as well as others. Thus, when the camera detects a breach, it will send an instant signal to address the matter. We will also use face recognition technology to identify people who wear masks when they enter public spaces such as stores, hospitals or schools.

Social distancing detection, here we use surveillance cameras and drones to detect violations committed by people, in case of overtaking within one meter, the camera sends an instant signal to stop the overtaking To detect the temperature, instead of using heat detectors to detect each person, we use the surveillance camera and sensors to accelerate the process

Vaccine detection technology is used only in hospitals to detect people, where the patient's gender, structure and age are determined, or if he has chronic diseases by simply standing in front of the camera, it will determine the correct dose that the patient will be injected.

BIBLIOGRAPHY

[1] *Coronavirus disease 2019 (covid-19): a perspective from china*, (2021).

[2] A. A. M. AL-SAFFAR, H. TAO, AND M. A. TALAB, *Review of deep convolution neural network in image classification*, in 2017 International Conference on Radar, Antenna, Microwave, Electronics, and Telecommunications (ICRAMET), IEEE, 2017, pp. 26–31.

[3] S. ALBAWI, T. A. MOHAMMED, AND S. AL-ZAWI, *Understanding of a convolutional neural network*, in 2017 International Conference on Engineering and Technology (ICET), Ieee, 2017, pp. 1–6.

[4] D. BISWAS, H. SU, C. WANG, A. STEVANOVIC, AND W. WANG, *An automatic traffic density estimation using single shot detection (ssd) and mobilenet-ssd*, Physics and Chemistry of the Earth, Parts A/B/C, 110 (2019), pp. 176–184.

[5] R. CHAUHAN, K. K. GHANSHALA, AND R. JOSHI, *Convolutional neural network (cnn) for image detection and recognition*, in 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC), IEEE, 2018, pp. 278–282.

[6] J. DAVIS AND M. GOADRICH, *The relationship between precision-recall and roc curves*, in Proceedings of the 23rd international conference on Machine learning, 2006, pp. 233–240.

[7] N. G. DE GROOT AND R. E. BONTROP, *Covid-19 pandemic: is a gender-defined dosage effect responsible for the high mortality rate among males?*, 2020.

[8] J. GARCIA AND F. FERNÁNDEZ, *A comprehensive survey on safe reinforcement learning*, Journal of Machine Learning Research, 16 (2015), pp. 1437–1480.

[9] I. GOODFELLOW, Y. BENGIO, A. COURVILLE, AND Y. BENGIO, *Deep learning*, vol. 1, MIT press Cambridge, 2016.

[10] A. HALEEM, M. JAVAID, AND R. VAISHYA, *Effects of covid 19 pandemic in daily life*, Current medicine research and practice, (2020).

[11] K. HE, X. ZHANG, S. REN, AND J. SUN, *Deep residual learning for image recognition*, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[12] N. KALCHBRENNER, E. GREFENSTETTE, AND P. BLUNSOM, *A convolutional neural network for modelling sentences*, arXiv preprint arXiv:1404.2188, (2014).

[13] M. KAYED, A. ANTER, AND H. MOHAMED, *Classification of garments from fashion mnist dataset using cnn lenet-5 architecture*, in 2020 International Conference on Innovative Trends in Communication and Computer Engineering (ITCE), IEEE, 2020, pp. 238–243.

[14] A. KRENKER, J. BEŠTER, AND A. KOS, *Introduction to the artificial neural networks*, Artificial Neural Networks: Methodological Advances and Biomedical Applications. InTech, (2011), pp. 1–18.

[15] C. KRITTANAWONG, H. ZHANG, Z. WANG, M. AYDAR, AND T. KITAI, *Artificial intelligence in precision cardiovascular medicine*, Journal of the American College of Cardiology, 69 (2017), pp. 2657–2664.

[16] Y. LI AND Y. DU, *A novel software-defined convolutional neural networks accelerator*, IEEE Access, 7 (2019), pp. 177922–177931.

[17] J. LIU AND X. WANG, *Early recognition of tomato gray leaf spot disease based on mobilenetv2-yolov3 model*, Plant Methods, 16 (2020), pp. 1–16.

[18] S. B. MAIND, P. WANKAR, ET AL., *Research paper on basic of artificial neural network*, International Journal on Recent and Innovation Trends in Computing and Communication, 2 (2014), pp. 96–100.

[19] S. MANI, *Artistic enhancement and style transfer of image edges using directional pseudo-coloring*, arXiv preprint arXiv:1906.07981, (2019).

[20] W. H. ORGANIZATION ET AL., *Advice on the use of masks in the context of covid-19: interim guidance, 5 june 2020*, tech. rep., World Health Organization, 2020.

[21] J. REDMON AND A. FARHADI, *Yolo9000: better, faster, stronger*, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 7263–7271.

[22] M. SANDLER, A. HOWARD, M. ZHU, A. ZHMOGINOV, AND L.-C. CHEN, *Mobilenetv2: Inverted residuals and linear bottlenecks*, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 4510–4520.

[23] S. SHARMA AND S. SHARMA, *Activation functions in neural networks*, Towards Data Science, 6 (2017), pp. 310–316.

[24] J. SINGH AND J. SINGH, *Covid-19 and its impact on society*, Electronic Research Journal of Social Sciences and Humanities, 2 (2020).

[25] ——, *Covid-19 and its impact on society*, Electronic Research Journal of Social Sciences and Humanities, 2 (2020).

[26] N. SRIVASTAVA, G. HINTON, A. KRIZHEVSKY, I. SUTSKEVER, AND R. SALAKHUTDINOV, *Dropout: a simple way to prevent neural networks from overfitting*, The journal of machine learning research, 15 (2014), pp. 1929–1958.

[27] C. SZEPESVÁRI, *Algorithms for reinforcement learning*, Synthesis lectures on artificial intelligence and machine learning, 4 (2010), pp. 1–103.

[28] S. TAMMINA, *Transfer learning using vgg-16 with deep convolutional neural network for classifying images*, International Journal of Scientific and Research Publications, 9 (2019), pp. 143–150.

[29] R. TOSEPU, J. GUNAWAN, D. S. EFFENDY, H. LESTARI, H. BAHAR, P. ASFIAN, ET AL., *Correlation between weather and covid-19 pandemic in jakarta, indonesia*, Science of The Total Environment, 725 (2020), p. 138436.

[30] T. WANG, Z. DU, F. ZHU, Z. CAO, Y. AN, Y. GAO, AND B. JIANG, *Comorbidities and multi-organ injuries in the treatment of covid-19*, The Lancet, 395 (2020), p. e52.

[31] ——, *Comorbidities and multi-organ injuries in the treatment of covid-19*, The Lancet, 395 (2020), p. e52.

[32] A. WILDER-SMITH AND D. O. FREEDMAN, *Isolation, quarantine, social distancing and community containment: pivotal role for old-style public health measures in the novel coronavirus (2019-ncov) outbreak.*, Journal of travel medicine, (2020).

[33] M. D. ZEILER AND R. FERGUS, *Visualizing and understanding convolutional networks*, in European conference on computer vision, Springer, 2014, pp. 818–833.

[34] L. ZHAO AND S. LI, *Object detection algorithm based on improved yolov3*, Electronics, 9 (2020), p. 537.

[35] Z. Y. ZU, M. D. JIANG, P. P. XU, W. CHEN, Q. Q. NI, G. M. LU, AND L. J. ZHANG, *Coronavirus disease 2019 (covid-19): a perspective from china*, Radiology, 296 (2020), pp. E15–E25.