الجمهورية الجزائرية الديمقراطية الشعبية
The People's Democratic Republic of Algeria
وزارة التعليم العالي والبحث العلمي
Ministry of Higher Education and Scientific Research
جامعة محمد بوضياف بالمسيلة
University Mohamed Boudiaf of M'sila
جامعة محمد بوضياف - المسيلة
Université Mohamed Boudiaf - M'sila
قسم الإعلام الآلي
كلية الرياضيات والإعلام الآلي

Domain: Mathematics and Computer Science

Thesis Presented to Fulfill the Partial Requirement
For Master's Degree in Computer Science

Specialty: Information System and Software Engineering

Prepared By: Amina Chaa, Noussaiba Dilmi

Supervised By:

Hichem Debbi

ENTITLED

On the detection of pneumonia using deep learning and explainability

Jury Members

Adel Moussaoui	President
Hichem Debbi	Supervisor
Ali Dabba	Examiner

Academic Year 2023/2024

DEDICATION

First and foremost, I want to thank God, for His guidance and blessings throughout my journey.

To my dad Brahim, my hero, for his wisdom, patience, and unwavering belief in me. Your lessons have shaped who I am today.

To my loving mother Yasmina Hebara, for her endless love, support, and sacrifices. Your strength and encouragement have been my greatest inspiration.

To my family, for their constant support and understanding. Your love has been a pillar of strength.

To my friends, for their companionship and encouragement. Your presence made this journey enjoyable and memorable.

To the special people in my life, with all my love and gratitude for their love to me and their unique contributions and unwavering support. Your belief in me has been invaluable.

To Palestine, for its resilience and unwavering spirit. Your struggle for freedom and doctrine inspires me every day. The events of October 7, for reminding us of the enduring struggle and the sacrifices made for freedom.

To all the martyrs in Palestine, for their ultimate sacrifice. Your bravery and dedication inspire us all to continue the fight for justice.

Sincerely and with love,

[Amina CHAA]

من قال أنا لها نالها، وأنا لها رغما عنها أتيت بها

قال الله تعالى: ﴿وَقُلِ ٱعْمَلُواْ فَسَيَرَى اللهُ عَمَلَكُمْ وَرَسُولُهُۥ وَٱلْمُؤْمِنُونَ﴾

الحمد والشكر لله أولا وأبدا أن بعونه بدأت وأتممت هذا العمل

إليك دائمًا قدوتي وقوتي، من وضعني على الطريق وأرشدني، ساندني ودعمني حتى وصلت،

الحبيب أبي ديلمي صديق –رحمه الله– لاقاني الله بك في الجنة يا ضلعي الثابت الذي لا يميل

إلى من كان دعائها سر نجاحي، إلى من سهرت الليالي برفقتي، من أنشأتني على شغف الاطلاع والمعرفة

الحبيبة أمي حيمر رقية بحنانها ودعمها وإيمانها بي

إلى داعمتي في كل طريق أختاره أمي الثانية إحسانًا ووفاءًا لها خالتي عائشة الطيبة التي لن توفي الكلمات حقها

﴿وَاجْعَلْ لِي وَزِيرًا مِنْ أَهْلِي هَارُونَ أَخِي اشْدُدْ بِهِ أَزْرِي﴾

إلى أخواتي سلمى وصفية ومنال وميسون، أخي منير وفؤاد، أحمد زوج أختي، إلى أحباء خالتهم آدم وعهد

إلى الصديقة والأخت ورفيقة الدرب عصماء

﴿ وَمِنْ آيَاتِهِ أَنْ خَلَقَ لَكُم مِّنْ أَنفُسِكُمْ أَزْوَاجًا لِّتَسْكُنُوا إِلَيْهَا ﴾

إلى إختيار وهدية الله لي، روح الروح صلاح، رفيق الأيام الصعبة لإتمام هذا العمل بدعمه وإيمانه بي ودفعه لي نحو الأمام دائمًا

إلى السابع من أكتوبر، إلى أرواح الشهداء، إلى غزة الصامدين المجاهدين، إلى أبطال العبور نكتب ونقتدي ونلبي ليجدنا العدو أينما يحذر ... يا من أتعبتم من بعدكم نفسي فداكم

[نسيبة ديلمي]

ACKNOWLEDGMENT

First and foremost, we express our thanks with great pleasure and great respect to God almighty and merciful, and our sincere gratitude to supervisor **Dr. DEBBI Hichem** for his invaluable guidance, encouragement, and patience throughout this journey. His experience in the field of artificial intelligence, especially explaining artificial intelligence, has greatly benefited us in shaping this thesis. We thank him for accompanying us in this academic year to complete the graduation project. We also thank **Dr. Adel Moussaoui** and **Dr. Dabba Ali** for their time and valuable feedback. We thank everyone who contributed with the support and encouragement that was always present with us.

TABLE OF CONTENTS

TABLE OF CONTENTS 6
LIST OF FIGURES 9
LIST OF TABLES 11
GENERAL INTRODUCTION 12
CHAPTER I: PNEUMONIA AND MEDICAL IMAGES 14
1.1 INTRODUCTION 15
1.2 PNEUMONIA 15
1.2.1 Pneumonia types 15
1.2.1.1 Bronchopneumonia 15
1.2.1.2 Lobar pneumonia 16
1.2.2 Pneumonia classification 17
1.2.3 Symptoms 17
1.2.4 Causes 18
1.2.5 Risk factors 19
1.2.5.1 Health risk factors 19
1.2.5.2 Lifestyle risk factors 19
1.2.6 The relationship between pneumonia and COVID-19 19
1.2.7 Diagnosis 20
1.3 MEDICAL IMAGES 20
1.3.1 Computed Tomography 21
1.3.2 Magnetic Resonance Imaging 21
1.3.3 Chest radiography 22
1.3.3.1 X-ray Image 22
1.3.3.2 Pneumonia in chest x-ray images 23
1.3.3.3 Pneumonia can be undetectable 23
1.4 CONCLUSION 24
CHAPTER II: DEEP LEARNING 25
2.1 INTRODUCTION 26
2.2 DEEP LEARNING 26
2.3 COMPUTER VISION 26
2.3.1 Computer vision applications 27
2.3.2 Convolutional neural network 28

2.3.2.1 Convolutional Neural Networks layers 29
2.4 COMPUTER VISION TASKS 29
2.4.1 Image classification 30
2.4.1.1 ImageNet 31
2.4.2 Object Detection 31
2.4.3 Image Segmentation 31
2.4.4 Feature Extraction and Recognition 32
2.4.5 Real-Time Processing 33
2.5 TRANSFER LEARNING 34
2.6 VISUAL GEOMETRY GROUP 16 34
2.6.1 VGG16 Architecture 35
2.7 CONCLUTION 36
CHAPTER III: EXPLAINABLE ARTIFICIAL INTELLIGENCE 37
3.1 INTRODUCTION 38
3.2 EXPLAINABILITY 38
3.3 EXPLAINABILITY AI METHODS 39
3.3.1 Local Interpretable Model-agnostic Explanations 39
3.3.2 SHapley Additive exPlanations 39
3.3.3 Feature maps 40
3.3.3.1 Saliency Maps 41
3.3.3.2 Class Activation Mapping 41
3.3.3.3 Gradient-weighted Class Activation Mapping 41
3.4 EXPLAINABILITY TYPES 41
3.4.1 Perceptive Explainability 42
3.4.1.1 Saliency 42
3.4.1.2 Signal Method 42
3.4.1.3 Verbal Explainability 43
3.4.2 Explainability via Mathematical Structure 43
3.4.2.1 Pre-defined Model 44
3.4.2.2 Feature Extraction 44
3.4.2.3 Sensitivity 45
3.5 XAI IN MEDICAL FIELD RELATED WORKS 46
3.5.1 Perceptive explainability 46
3.5.1.1 Saliency 46

3.5.2 Explainability via Mathematical Structure 47
3.5.2.1 Pre-defined Model 48
3.5.2.2 Feature Extraction 48
3.6 EXPLAINABILITY CRITERIA 49
3.7 LOCALIZATION FOR EXPLAINABILITY 50
3.7.1 Bounding BOX 50
3.7.2 Intersection over Union 51
3.8 CONCLUSION 52
CHAPTER IV: EXPLAINING PNEUMONIA DISEASE THROUGH CAUSAL LEARNING 53
4.1 INTRODUCTION 54
4.2 DATASET OVERVIEW 54
4.3 MODEL ARCHITECTURE 56
4.4 EXPLANATION FRAMEWORK 57
4.4.1 Dataset augmentation 57
4.4.2 Training and results 58
4.4.3 Visualizing Heatmap and Bounding Box before CexCNN Method 59
4.4.4 Causal explanation of CNN 60
4.4.4.1 Causality code overview 60
4.4.5 Visualizing Heatmap and Bounding Box after CexCNN Method 62
4.4.6 Comparing Feature Maps 63
4.5 IMPLEMENTATION TOOLS 63
4.5.1 Programming language (Python) 63
4.5.2 Kaggle 64
4.5.3 Google Colaboratory 64
4.5.4 TensorFlow 65
4.5.5 Why TensorFlow 65
4.5.6 Numpy, sklearn, matplotlib, seaborn, skimage 66
4.6 CONCLUSION 66
GENERAL CONCLUSION 67
BIBLIOGRAPHIE 68

LIST OF FIGURES

Figure I- 1: Normal air sacs and inflamed air sacs filled with fluid (pneumonia) 15
Figure I- 2 : Bronchopneumonia (Type of pneumonia) 16
Figure I- 3 : Lobar pneumonia (Type of pneumonia) 16
Figure I- 4 : The classical stages of inflammatory lobar pneumonia 17
Figure I- 5 : CXR for Normal and pneumonic and COVID-19 image dataset 20
Figure I- 6 : Computed tomography (CT) 21
Figure I- 7 : Magnetic Resonance Imaging 22
Figure I- 8 : Chest x-ray image 22
Figure I- 9 : CXR for Normal and pneumonic image 23

Figure II- 1 : Human vision system & Computer vision system 27
Figure II- 2 : Convolutional neural network 28
Figure II- 3 : Object detection & Image segmentation & Image classification 30
Figure II- 4 : VGG16 model Architecture 36

Figure III- 1 : Explainable Artificial Intelligence Concept 38
Figure III- 2 : Local Interpretable Model-Agnostic Explanations method 39
Figure III- 3 : SHapley Additive exPlanations method 40
Figure III- 4 : (A) chest X-ray image of a patient with pneumonia, (B) CAM, (C) Grad-CAM, (D) saliency map 40
Figure III- 5 : Feature maps for brain tumor classification 47
Figure III- 6 : The visual representation of IoU 51
Figure III- 7 : IoU score representation 52

Figure IV- 1 : Function to convert the format of the images 54
Figure IV- 2 : Examples from the datasets 55
Figure IV- 3 : Framework explanation 57
Figure IV- 4 : Examples of class prediction for the “Chest X-Ray Images (Pneumonia)” dataset 58
Figure IV- 5 : Examples of class prediction for the “RSNA Pneumonia Detection Challenge” dataset 59
Figure IV- 6 : (A) Pneumonia chest X-ray, (B) CAM, (C) Bounding box on CAM of “Chest X-Ray Images (Pneumonia)” dataset before the CexCNN method 59
Figure IV- 7 : (A) Pneumonia chest X-ray, (B) CAM, (C) Bounding box on CAM of “RSNA Pneumonia Detection Challenge” dataset before the CexCNN method 60
Figure IV- 8 : Prediction Class Probabilities Code 61
Figure IV- 9 : Iterative Filter Pruning Impact Code 61
Figure IV- 10 : Sorted Weights Code 62

Figure IV- 11 : (A) Pneumonia chest X-ray, (B) CAM, (C) Bounding box on CAM of "Chest X-Ray Images (Pneumonia)" dataset after the CexCNN method 62
Figure IV- 12 : Pneumonia chest X-ray, (B) CAM, (C) Bounding box on CAM of "RSNA Pneumonia Detection Challenge" dataset after the CexCNN method 63
Figure IV- 13 : PYTHON LOGO 64
Figure IV- 14 : Kaggle LOGO 64
Figure IV- 15 : Google Colaboratory LOGO 65
Figure IV- 16 : TensorFlow LOGO 65

LIST OF TABLES

Table III- 1 : Related works 49

Table IV- 1 : “RSNA Pneumonia Detection Challenge” Dataset Overview 55
Table IV- 2 : “Chest X-Ray Images (Pneumonia)” Dataset Overview 55
Table IV- 3 : Architecture of the Proposed Model for VGG16 56

GENERAL INTRODUCTION

Pneumonia disease is an infection that inflames the air sacs in one or both lungs, caused by various pathogens such as bacteria, viruses, and fungi. According to the World Health Organization, pneumonia causes millions of deaths. Around 2.5 million people die from pneumonia each year [1], especially among children under five and the elderly. Early detection of pneumonia is important for reducing mortality and improving clinical outcomes. However, diagnosis can be complex due to the variability of symptoms and the quality of radiographic images. Traditional diagnostic techniques often rely on human expertise, which can lead to diagnostic errors, especially in resource-limited regions. Deep learning technologies, particularly convolutional neural networks (CNNs), have recently shown promising performance in medical image analysis. The Visual Geometry Group16 (VGG16) model, known for its depth and ability to extract relevant features from images, is one of the most widely used CNN models for various image classification tasks. Moreover, the explainability of deep learning models has become a major concern, especially in the medical field, where decisions must be transparent and understandable to clinicians. Explanation techniques, such as those based on causal explanation convolutional neural networks (CexCNN), help identify important features in images that influence model decisions, thereby enhancing the trust and acceptance of models by healthcare professionals.

In this thesis, we explore the use of the VGG16 model for detecting pneumonia from chest X-ray images, integrating explanation techniques to ensure model transparency. We aim to develop an efficient and interpretable detection system that can be used in various clinical contexts to improve the diagnosis and treatment of pneumonia, and the structure of this thesis is as follows:

The first chapter introduces pneumonia diseases, their causes, symptoms, and traditional diagnostic methods, and we dive into the medical image. The second chapter provides a state-of-the-art review of deep learning methods applied to disease detection from medical images, with a focus on the VGG16 model and explanation techniques.

The third chapter delves into the concept of explainability in Artificial Intelligence (AI) models. It explores different techniques for making these models more interpretable and examines relevant research in the medical field.

The fourth chapter explores explaining pneumonia diagnosis using causal learning. It delves into the inner workings of models like VGG16 and CexCNN for this task, alongside analyzing how bounding boxes contribute to the explanation process.

CHAPTER I: PNEUMONIA AND MEDICAL IMAGES

1.1 INTRODUCTION

Pneumonia is a major global public health concern and a leading cause of mortality and morbidity. It is the primary cause of death in children, particularly during the COVID-19 pandemic. According to the World Health Organization (WHO), one in three newborn infants dies from pneumonia, often due to late detection [2]. The United Nations International Children's Emergency Fund (UNICEF) states that pneumonia is the leading infectious killer of children, causing over 725,000 deaths annually, with a child dying every 43 seconds. Most of these deaths are preventable [2]. Medical image diagnosis can play a crucial role in early detection, yet medical services frequently face delays in identifying pneumonia cases.

1.2 PNEUMONIA

Pneumonia is inflammation of the air sacs in the lungs in response to an injury, like an infection. When the airways are also involved, it may be called bronchopneumonia. Pneumonia can be in one area of a lung or be in several areas (“double” or “multilobar” pneumonia). Pneumonia can range in severity from mild to life-threatening. It is most dangerous for infants, young children, people over 65 years old, and those with health issues or weak immune systems [3].

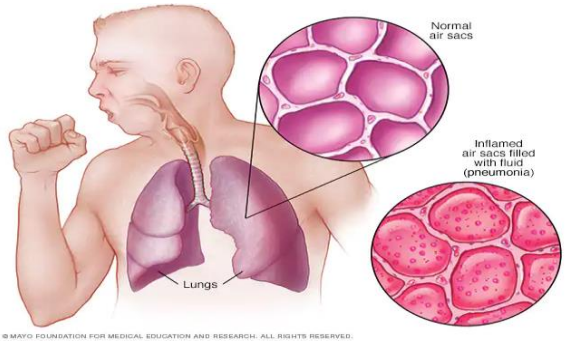


Figure I- 1: Normal air sacs and inflamed air sacs filled with fluid (pneumonia)

1.2.1 Pneumonia types

Pneumonia is typically categorized into two main types according to the extent of lung involvement. Depending on which lobes or areas of the respiratory system are affected [4].

1.2.1.1 Bronchopneumonia

Bronchopneumonia is a form of pneumonia that predominantly impacts the bronchus and bronchioles, the airway passages responsible for air entry into the lungs. If not effectively treated, this type of pneumonia often extends to other regions of the lungs [5].

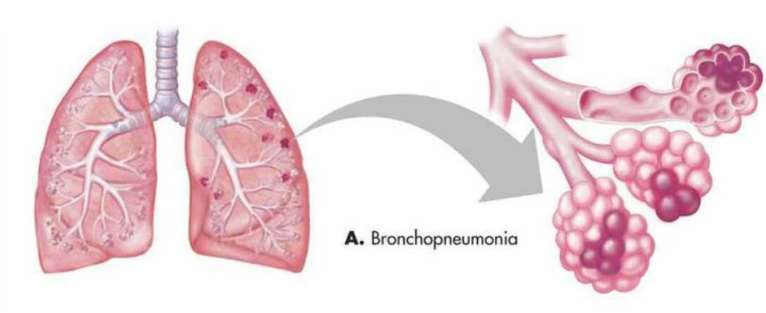


Figure I- 2 : Bronchopneumonia (Type of pneumonia)

1.2.1.2 Lobar pneumonia

This refers to any type of pneumonia that causes congestion or inflammation (a condition known as consolidation), which affects the complete or near-complete consolidation of a lobe of a lung. Most of these cases are caused by Streptococcus pneumoniae [6]

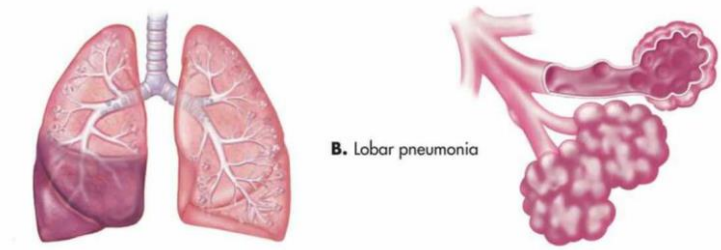


Figure I- 3 : Lobar pneumonia (Type of pneumonia)

Lobar pneumonia has distinct classical stages of inflammatory response, namely [6]:

- **Congestion:** Or consolidation, is typically the first stage of infection that occurs after a pneumonia infection has dominated one of the lobes in the lung. This stage usually lasts about 24 hours, and the lungs become inflamed, red, and weighed down by infection. When looking at lung tissue under a microscope during this stage, we'll see swollen air sacs (alveoli) and enlarged blood vessels where oxygen and carbon dioxide are exchanged;
- **Red hepatization**: During the red hepatization or early consolidation stage, which starts 2 to 3 days after initial consolidation and lasts for 2 to 4 days, the affected lung looks red-pink, dry, and firm like the liver. Inside, there's a buildup of cellular material, mainly neutrophils, along with some bacteria. Also, there's leakage of red blood cells, shedding of lung cells, and fibrin within the air sacs;
- **Gray hepatization:** During the gray hepatization or late consolidation stage, which happens 2 to 3 days after red hepatization and lasts for 4 to 8 days, the lung looks gray

and has a liver-like texture. This is because of the presence of a mixture of fibrin and pus, along with the breaking down of red blood cells and the accumulation of hemosiderin. Additionally, macrophages start to appear during this stage;

- **Resolution:** By the eighth day, the lungs begin to recover and rebuild their structure. Enzymes start breaking down the solid material in the center of the affected area, gradually restoring normal airflow. Macrophages, mainly containing swallowed neutrophils and debris, play a key role in this process.

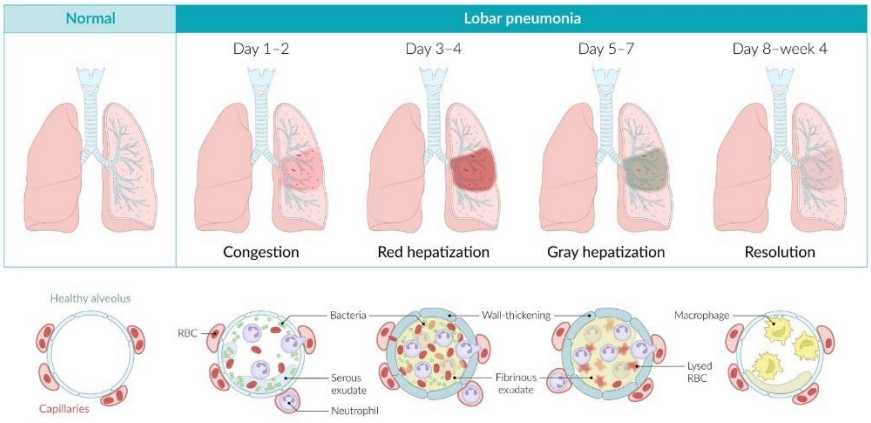


Figure I- 4 : The classical stages of inflammatory lobar pneumonia

1.2.2 Pneumonia classification

Pneumonia is classified into three subgroups on the basis of the clinical context in which the patient develops symptoms of infection [7]:

- **Community-acquired pneumonia (CAP):** This term used to describe an acute infection of the lungs that develops outside the hospital setting in a patient who has not been recently hospitalized [8];
- **Hospital-acquired (or nosocomial) pneumonia (HAP):** Is the second most common hospital acquired infection. HAP is a pneumonia that occurs 48 hours or more after admission and was not incubating at the time of admission [9];
- **Ventilator-associated pneumonia (VAP):** Is defined by infection of the pulmonary parenchyma in patients exposed to invasive mechanical ventilation for at least 48 h and is part of ICU-acquired pneumonia [10].

1.2.3 Symptoms

The signs and symptoms of pneumonia vary from mild to severe, depending on factors such as the type of germ causing the infection, and your age and overall health. Mild signs and

symptoms often are similar to those of a cold or flu, but they last longer. Signs and symptoms of pneumonia may include [11]:

- Chest pain when you breathe or cough;
- Confusion or changes in mental awareness (in adults age 65 and older);
- Cough, which may produce phlegm;
- Fatigue;
- Fever, sweating and shaking chills;
- Lower than normal body temperature (in adults older than age 65 and people with weak immune systems);
- Nausea, vomiting or diarrhea;
- Shortness of breath.

1.2.4 Causes

Pneumonia is typically caused by a virus or bacteria you have been exposed to in the environment or is passed to you from another person. Infection can be passed between people from direct contact (usually the hands) or inhaling droplets in the air from coughing or sneezing [3]. Here's a structured breakdown of the causes:

- **Bacterial pneumonia:** Most prevalent, often affecting a specific area of the lung. Culprits include Streptococcus pneumoniae, Haemophilus influenzae, Staphylococcus aureus, Mycobacterium tuberculosis, Gram-negative bacilli, and anaerobes [12];
- **Atypical bacteria:** These are bacteria that cannot be grown using normal methods, such as Mycoplasma pneumoniae, Chlamydia pneumoniae, and Legionella pneumophila can also cause pneumonia [13];
- **Viral pneumonia:** Predominantly afflicts children under 5, caused by viruses like influenza, coronaviruses, parainfluenza, adenovirus, respiratory syncytial virus (RSV), and human metapneumovirus (HMPV) [14];
- **Fungal pneumonia:** Can occur in individuals with compromised immune systems or chronic health problems. Fungal infections picked up from the soil in certain areas can lead to fungal pneumonia. Examples of fungi causing pneumonia include Cryptococcus, Histoplasma, Pneumocystis jiroveci, and Coccidioides (causing "valley fever") [15];

- **Aspiration pneumonia:** Occurs when foreign objects or substances, such as food, liquid, or vomit, are inhaled into the lungs, leading to inflammation and infection [16].

1.2.5 Risk factors

Pneumonia can be influenced by various factors, including health conditions and lifestyle choices. Understanding these risk factors can help in preventive measures and early detection.

1.2.5.1 Health risk factors

Pneumonia has the potential to impact individuals of all ages; however, it is essential to highlight two specific age groups that face a higher risk of contracting the disease and experiencing more severe cases. These groups include children below the age of 2 and adults who are 65 years and older. Other risk factors include [17]:

- Being in the hospital;
- Having a chronic disease;
- Having a suppressed immune system;
- Difficulty swallowing;
- Reduced consciousness;
- Difficulty coughing;
- older people with weakened immune systems;
- Pregnant women.

1.2.5.2 Lifestyle risk factors

Risk factors associated with lifestyle choices can elevate the probability of developing pneumonia and include [17]:

- Smoking;
- Drug or excessive alcohol use;
- Malnutrition;
- Poor dental health;
- Exposure to animals, chemicals, or environmental toxins.

1.2.6 The relationship between pneumonia and COVID-19

Coronavirus Disease 2019 (COVID-19) has become a global pandemic with an exponential growth rate and an incompletely understood transmission process. The virus is harbored most

commonly with little or no symptoms, but can also lead to a rapidly progressive and often fatal pneumonia in 2–8% of those infected [18]. COVID-19 symptoms and signs are almost identical to those of pneumonia because of a specific type called COVID pneumonia. When COVID-19 virus leads to pneumonia, it usually affects both lungs. While most people recover from other forms of pneumonia without any lasting lung issues, COVID-19-related pneumonia often results in persistent scarring and stiffness. This can cause a condition called interstitial lung disease, making it challenging to breathe properly and perform daily activities like eating [19].

Figure I- 5 : CXR for Normal and pneumonic and COVID-19 image dataset

1.2.7 Diagnosis

Diagnosis is based on symptoms and signs of an acute lower respiratory tract infection and confirmed by a chest X-ray showing new shadowing that is not due to any other cause (such as pulmonary oedema or infarction) [20]. The x-ray can reveal both the amount and location of the inflammation in the lung tissue. There may be a need for a more complex x-ray, called a CT scan. Also, ultrasonography can identify changes in the outermost parts of the lungs. Another method used is called pulse oximetry, it involves placing a clip with a light sensor on one of your fingers to measure the oxygen content optically [21]. And many other tests are used to confirm the diagnosis and find out exactly which germs caused the inflammation, such as blood, urine, and coughed-up mucus (phlegm) tests [22].

1.3 MEDICAL IMAGES

Medical imaging is the process of producing visible images of inner structures of the body for scientific and medicinal study and treatment as well as a visible view of the function of interior tissues. This process pursues the disorder identification and management. This process creates data bank of regular structure and function of the organs to make it easy to recognize the anomalies. This process includes both organic and radiological imaging which used electromagnetic energies (X-rays and gamma), sonography, magnetic, scopes, and thermal and isotope imaging [23]. Medical images are acquired using different modalities.

1.3.1 Computed Tomography

A Computed Tomography (CT) scan, or computed tomography, is a diagnostic imaging procedure combining X-rays and computer technology to produce detailed images of the body's internal structures, including bones, muscles, fat, organs, and blood vessels. Unlike standard X-rays, which provide limited detail, CT scans capture multiple views by rotating an X-ray beam around the body. This data is processed by a computer to create detailed two-dimensional images, with newer technology enabling three-dimensional images. CT scans are used to diagnose tumors, investigate internal bleeding, and check for other injuries or damage, and can also guide biopsies [24].

Figure I- 6 : Computed tomography (CT)

1.3.2 Magnetic Resonance Imaging

Magnetic Resonance Imaging (MRI) is a medical procedure that creates images of internal body structures using strong magnetic fields and radio waves. The imaging primarily captures signals from protons in fat and water molecules. During an MRI, an electric current generates a temporary magnetic field in the body, and radio waves are used to produce digital images.

The process typically takes between 20 to 90 minutes, depending on the area being scanned [25].

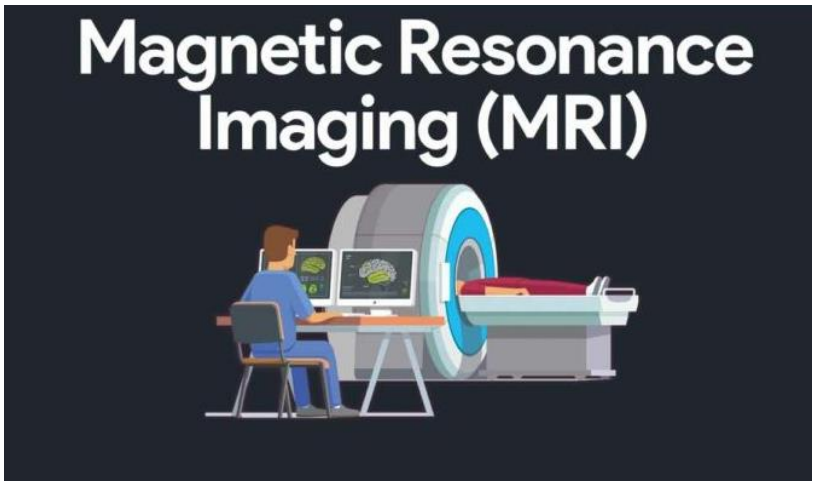


Figure I- 7 : Magnetic Resonance Imaging

1.3.3 Chest radiography

Chest radiography, or X-ray is among the most frequent and cost-effective diagnostic procedures done in medical facilities. X-rays are electromagnetic radiations that penetrate human body producing an image of the internal structures. It distinguishes objects based on its density; the four basic densities found on a radiograph are bone (appears white), soft tissues (appears white to gray), fat (appears gray) and gas (appears black) [26].

1.3.3.1 X-ray Image

X-ray images appear as if the negative images of black-and-white photographs. Human body have many tissues whose thickness varies depending on the sort of tissue [27]. In short, black is air, white is bone, and gray is tissue or fluid.

Figure I- 8 : Chest x-ray image

1.3.3.2 Pneumonia in chest x-ray images

In chest x-ray images, a pneumonic lung will have white spots or cloud-like structures called infiltrates, which is a sign of infection [27].

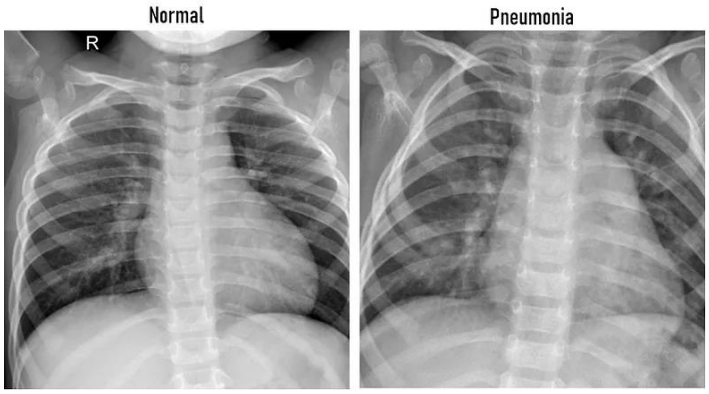


Figure I- 9 : CXR for Normal and pneumonic image

1.3.3.3 Pneumonia can be undetectable

Pneumonia can sometimes go undetected in a chest X-ray due to several reasons:

Certainly, here's a revised version with your addition [28]:

- **Hidden Pneumonias:** Some pneumonias are located in areas difficult for radiologists to see, such as below the diaphragms, along the spine, and near the heart and mediastinum.
- **Ground Glass Pneumonias:** Pneumonias with a ground glass appearance, like viral infections such as COVID-19 and pneumocystis pneumonia in immunocompromised patients, do not show well on chest X-rays.
- **Few Imaging Findings:** Some lung infections, like bronchitis or certain viral infections, have subtle findings on chest X-rays. Small pneumonias may blend in with normal structures or hide behind ribs or the heart.
- **Technical Factors and Obesity:** Obesity and technical limitations, especially with portable X-ray techniques, can make it difficult to detect pneumonias due to the soft tissues and fat of large patients.
- **Chronic Lung Disease:** Pneumonias can be harder to diagnose in patients with chronic lung disease as they can be mistaken for existing abnormalities, especially without prior X-rays for comparison.
- **Misdiagnosis:** Conditions like lung collapse (atelectasis) or cancer can be misdiagnosed as pneumonia. Follow-up imaging is often recommended to ensure the pneumonia clears after antibiotics.

- **Further Imaging Needed:** Pneumonia does not always show on chest X-rays. If suspected despite a normal X-ray, a CT scan or additional tests may be required. Follow-up imaging can help distinguish pneumonia from other abnormalities.

1.4 CONCLUSION

In conclusion, this chapter provided an overview of pneumonia and how to diagnose it, which requires a chest radiograph (x-ray), as it is the "gold standard" for confirming the presence of the disease, which plays a crucial role in the diagnosis and management of pneumonia. They offer a non-invasive and relatively quick method for detecting lung abnormalities associated with pneumonia, such as infiltrates or consolidations. Additionally, advancements in technology, particularly the integration of artificial intelligence, enhance the accuracy and efficiency of pneumonia detection from chest X-rays. Early detection through chest X-rays allows for prompt initiation of appropriate treatment, which can significantly improving patient outcomes and reducing the risk of complications. Therefore, chest X-rays remain an indispensable tool in the armamentarium of healthcare professionals for diagnosing and monitoring pneumonia.

CHAPTER II: DEEP LEARNING

2.1 INTRODUCTION

The term Deep Learning or Deep Neural Network refers to Artificial Neural Networks (ANN) with multiple layers. Over the last few decades, it has been considered one of the most powerful tools due to its ability to handle a huge amount of data. The interest in having deeper hidden layers has recently begun to surpass classical methods performance in different fields, especially in pattern recognition. One of the most popular deep neural networks is the Convolutional Neural Network (CNN). CNN has excellent performance in machine learning problems. Especially in the applications that deal with image data, such as the largest image classification data set (Image Net), VGG16, and computer vision, the results achieved were amazing. In this chapter, we will explain and define all the elements and important issues about deep learning related to CNN, VGG16, and computer vision.

2.2 DEEP LEARNING

Deep learning is an artificial intelligence (AI) function that mimics the functioning of the human brain in data processing and creating models for use in decision-making. Deep learning is a subset of machine learning in artificial intelligence that has networks that can learn unattended from unstructured or untagged data. Also known as deep neural learning or deep neural network, which is used for recognizing or diagnosing the real-time condition of patients. Deep learning has the ability to regenerate the real-time condition of patients on the basis of raw data or previous data [29].

2.3 COMPUTER VISION

Computer vision is a field of artificial intelligence (AI) that uses machine learning and neural networks to teach computers and systems to derive meaningful information from digital images, videos, and other visual inputs and to make recommendations or take actions when they see defects or issues. If AI enables computers to think, computer vision enables them to see, observe, and understand [30]. It works much the same as human vision, except humans have a head start. Human sight has the advantage of lifetimes of context to train how to tell objects apart, how far away they are, whether they are moving, or whether something is wrong with an image. It trains machines to perform these functions, but it must do it in much less time with cameras, data, and algorithms than with retinas, optic nerves, and a visual cortex. Because a system trained to inspect products or watch a production asset can analyze thousands of

products or processes a minute, noticing imperceptible defects or issues, it can quickly surpass human capabilities.

Instead of looking at an entire image like humans do, a computer divides it into pixels and uses the Red Green Blue (RGB) values of each pixel to understand if the image contains important features. Computer vision algorithms focus on one pixel blob at a time and use a kernel or filter that contains pixel multiplication values for edge detection of objects. The computer recognizes and distinguishes the image by observing all aspects of it, including colors, shadows, and line drawings. Computers learn pattern recognition by feeding them numerous labelled images so that they can look for patterns in all the elements.

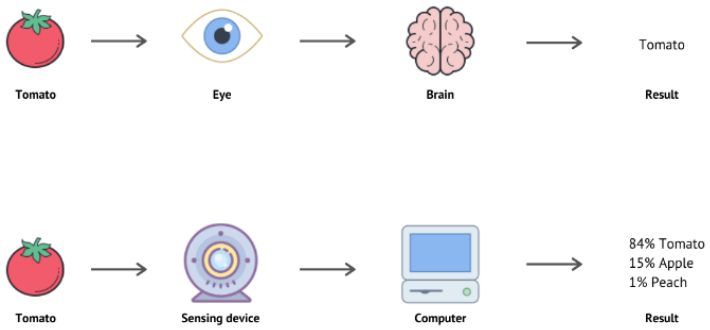


Figure II- 1 : Human vision system & Computer vision system

2.3.1 Computer vision applications

Now that our daily lives are hard without computer vision, it plays a major role in all fields of industry, health, transport, study, etc. There are many other examples [31]. The most prominent examples are:

- **Healthcare**: Computer Vision (CV) can help doctors and medical professionals quickly identify diseases, provide an accurate diagnosis, offer personalized treatments for patients, monitor medication use, and even predict health outcomes. Examples include cancer detection, COVID-19 diagnosis, cell classification, cell task detection, tumor detection, and disease progression score;
- **Retail**: as an example of this, visual search has been incorporated into some e-commerce platforms, allowing brands to recommend items that would complement an existing wardrobe. Other applications include customer tracking, people counting, and theft detection;

- **Transportation:** such as vehicle classification, occupancy detection, Automated License Plate Recognition (ALPR), pedestrian detection, road condition monitoring, and driver attentiveness detection;
- **Agriculture:** Animal Monitoring, Farm Automation Plantation Monitoring, Insect Detection, and Plant Disease Detection.

2.3.2 Convolutional neural network

A convolutional neural network (CNN), which takes its name from a mathematical linear operation between matrices called convolution, is a multilayer neural network that was biologically inspired by the animal visual cortex. The architecture is particularly useful in image-processing applications. The first CNN was created by Yann LeCun; at the time, the architecture focused on handwritten character recognition, such as postal code interpretation. As a deep network, early layers recognize features such as edges, and later layers recombine these features into higher-level attributes of the input [32].

The CNN architecture is made up of several layers that implement feature extraction and then classification. The image is divided into receptive fields that feed into a convolutional layer, which then extracts features from the input image. The next step is pooling, which reduces the dimensionality of the extracted features through down-sampling while retaining the most important information, typically through max pooling. Another convolution and pooling step is then performed, which feeds into a fully connected multilayer perceptron. The final output layer of this network is a set of nodes that identify features of the image; in this case, a node per identified number. that is training the network by using back-propagation [33].

The use of deep layers of processing, convolutions, pooling, and a fully connected classification layer opened the door to various new applications of deep learning neural networks. In addition to image processing, CNN has been successfully applied to video recognition and various tasks within natural language processing. Example applications: image recognition, video analysis, and natural language processing [34].

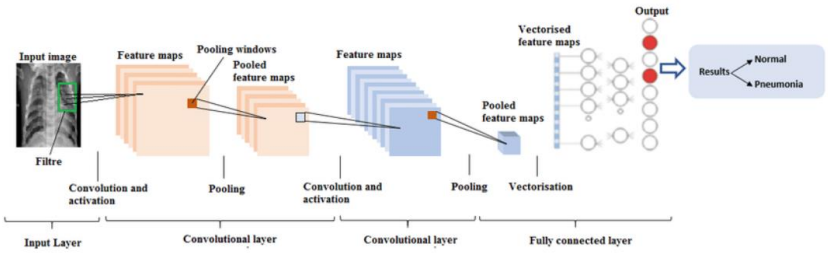


Figure II- 2 : Convolutional neural network

2.3.2.1 Convolutional Neural Networks layers

A CNN helps a machine learning or deep learning model “look” by breaking images down into pixels that are given tags or labels. It uses the labels to perform convolutions, a mathematical operation on two functions to produce a third function, and makes predictions about what it is “seeing.” The neural network runs convolutions and checks the accuracy of its predictions in a series of iterations until the predictions start to come true. It then recognizing or sees images in a way similar to humans’ [32].

Convolutional neural networks are distinguished from other neural networks by their superior performance with image, speech, or audio signal inputs. They have three main types of layers, which are:

- **Convolutional layer:** It is the main element of the CNN structure, consisting of filters, also called "kernels," that extract features from the input image;
- **Pooling layer:** it reduces the number of parameters the network needs to learn by inserting the image into the filter, reducing the features in the input, and streamlining the outputs;
- **Fully-connected (FC) layer:** inside this layer, image classification happens in the CNN based on the features extracted in the previous layers.

The convolutional layer is the first layer of a convolutional network. While convolutional layers can be followed by additional convolutional layers or pooling layers, the fully connected layer is the final layer. With each layer, the CNN increases in complexity, identifying greater portions of the image. Earlier layers focused on simple features, such as colors and edges. As the image data progresses through the layers of the CNN, it starts to recognize larger elements or shapes of the object until it finally identifies the intended object.

Convolutional neural networks use three-dimensional data for image classification and object recognition tasks.

2.4 COMPUTER VISION TASKS

Researchers in computer vision aspired to develop algorithms for such visual perception tasks [33]. These algorithms include different methods involved in recognizing things in images. The most common algorithms are:

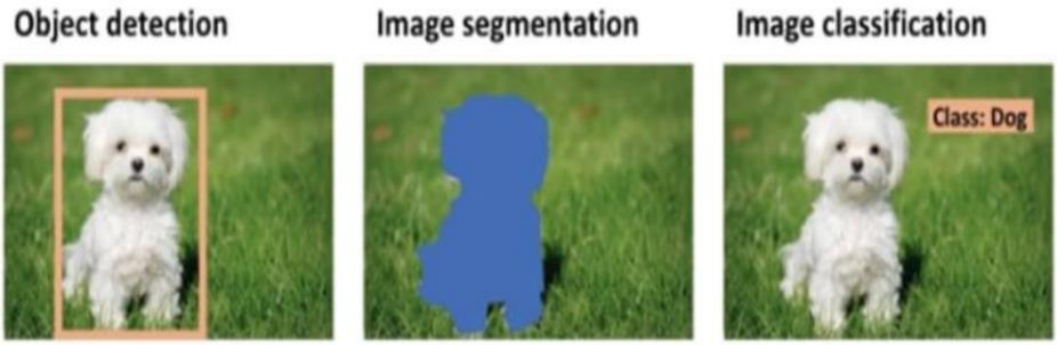


Figure II- 3 : Object detection & Image segmentation & Image classification

2.4.1 Image classification

Image classification is a process in computer vision that helps machines understand and categorize digital images. It accepts the given input images and produces an output classification for identifying whether the disease is present or not [29]. Image classification involves analyzing and interpreting visual data, training algorithms to recognize patterns, features, and structures within images, and assigning them to predefined categories or labels. This process is essential for various applications, including organizing collections of digital images, powering autonomous vehicles, and medical diagnostics. The process begins with the acquisition of digital images, which undergo preprocessing to ensure uniformity. The algorithms then analyze the visual features, extracting patterns indicative of specific classes or categories. The algorithm learns to associate these features with corresponding labels through training and iterative optimization processes. For decades, researchers have laid the path for developing advanced techniques to improve classification accuracy. In parallel with the creation of a large-scale image dataset, "ImageNet," the classification of images has changed significantly and started to show great performance in classification. This is because of the development of profound deep learning technology. Before deep learning came out, there were other methods such as designing scale-invariant features and classifications, and Support Vector Machine (SVM) learning algorithm, but all these traditional methods became weak in front of complex natural image features. Today, the classification model based on deep CNN performs much more robustly than old methods due to its effectiveness in recognizing optical patterns in pixel images with minimal pre-processing. This process is vital for various applications, including healthcare, agriculture, and autonomous systems [35].

2.4.1.1 ImageNet

The ImageNet dataset contains 14,197,122 annotated images according to the WordNet hierarchy [36]. Since 2010, the dataset has been used in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC), a benchmark in image classification and object detection. The publicly released dataset contains a set of manually annotated training images [36]. A set of test images is also released, with the manual annotations withheld. ILSVRC annotations fall into one of two categories:

- image-level annotation of a binary label for the presence or absence of an object class in the image, e.g., "there are cars in this image" but "there are no tigers;"
- Object-level annotation of a tight bounding box and class label around an object instance in the image, e.g., "There is a screwdriver centered at position (20, 25) with a width of 50 pixels and a height of 30 pixels."

The ImageNet project does not own the copyright to the images; therefore, only thumbnails and Uniform Resource Locator (URLs) of the images are provided.

2.4.2 Object Detection

Object detection is a computer technology that identifies semantic objects in digital images and videos and draws frames around those detected objects [37]. This lets us locate them in a given scene. Each object class has unique features that help classify it, such as round circles or perpendicular squares. Object class detection uses these features to classify objects, such as finding objects at a specific distance from a point or perpendicular at corners. Object detection methods are divided into two approaches: neural network-based and non-neural approaches [35]. Before deep learning, object detection was done through classical machine learning techniques, such as scale-invariant feature transform (SIFT). These algorithms would define features, then use a technique such as SVM to do the classification. Either neural network approaches or deep learning-based approaches use neural network architecture to detect objects without specifically defining features. It has applications in computer vision, image retrieval, and video surveillance.

2.4.3 Image Segmentation

Image segmentation is a computer vision technique that partitions a digital image into discrete groups of pixels—image segments—to identify the boundary of an object, simplify an image, and analyze it more efficiently. Image segmentation enables faster, more advanced

image processing. The result of segmenting is that a set of segments has a set of pixels, and each of the pixels in a region is similar with respect to any characteristic or computed property [30]. Image segmentation is divided into three categories:

- **Semantic segmentation:** is an approach to detecting, for every pixel, the belonging class [38]. For example, in a figure with many people, all the pixels belonging to persons will have the same class ID, and the pixels in the background will be classified as background;
- **Instance segmentation:** is an approach that identifies, for every pixel, the specific belonging instance of the object. It detects each distinct object of interest in the image [39]. For example, when each person in a figure is segmented as an individual object;
- **Panoptic segmentation:** combines both semantic and instance segmentation. Like semantic segmentation, panoptic segmentation is an approach that identifies, for every pixel, the belonging class. Moreover, like instance segmentation, panoptic segmentation distinguishes different instances of the same class.

2.4.4 Feature Extraction and Recognition

Feature extraction and recognition are integral components of many machines learning and computer vision tasks, including image classification [40], object detection, and facial recognition. These processes involve identifying and capturing relevant information or characteristics (features) from raw data, such as images, to enable effective analysis and interpretation by algorithms. The following points explores the difference between them:

- **Feature extraction:** Is the process of transforming raw data into a format that is more suitable for analysis. In the context of image processing and computer vision, this often involves identifying distinctive patterns, textures, edges, or other relevant attributes present in the data. These features serve as meaningful representations of the underlying data and are crucial for subsequent analysis and decision-making. In traditional computer vision approaches, feature extraction was often performed using handcrafted techniques, where domain-specific knowledge and expertise were used to design algorithms for extracting relevant features. For example, in image processing, common techniques such as edge detection, corner detection, and texture analysis were used to extract features from images. However, with the advent of deep learning, feature extraction has largely shifted towards data-driven approaches. Deep learning architectures, particularly convolutional neural networks (CNNs), are

capable of automatically learning hierarchical representations of features directly from raw data. In CNNs, lower layers typically capture simple features like edges and textures, while higher layers capture more abstract and complex features, allowing for highly effective feature extraction in tasks like image classification and object detection;

- **Feature Recognition:** Once relevant features have been extracted from the data, feature recognition involves interpreting and identifying these features to make meaningful decisions or predictions. In the context of computer vision, this often entails recognizing objects, patterns, or structures within images based on the extracted features. In image classification, for example, feature recognition involves using the extracted features to determine the category or label of an image. Similarly, in object detection, the extracted features are used to localize and identify objects within images. Feature recognition can also extend to more complex tasks such as facial recognition, where features specific to faces, such as eyes, nose, and mouth, are identified and used for identity verification or classification. Overall, feature extraction and recognition are essential stages in many machines learning and computer vision pipelines, enabling algorithms to effectively analyze and interpret raw data for a wide range of applications. Whether through traditional handcrafted techniques or modern data-driven approaches like deep learning, these processes play a critical role in unlocking the potential of machine learning and artificial intelligence.

2.4.5 Real-Time Processing

Real-time processing is the capability of a system to perform computations and generate outputs within a very short timeframe, typically in milliseconds or microseconds. This real-time capability is essential for applications where timely responses are crucial, such as autonomous driving, video surveillance, augmented reality, and live streaming for disease detection. In the context of computer vision, real-time processing involves analyzing and interpreting visual data, such as images or videos, in near real-time to extract meaningful insights or make instantaneous decisions. This requires efficient algorithms and hardware architectures that can handle the computational workload within strict time constraints [41].

2.5 TRANSFER LEARNING

Transfer learning is a machine learning technique where a model trained on one task or dataset is re-purposed or fine-tuned for another related task or dataset. Instead of training a new model from scratch, transfer learning leverages the knowledge and representations learned by the pre-trained model, which is typically trained on a large and general dataset, and applies them to a specific task with a smaller dataset [42]. The process of transfer learning typically involves several key steps:

- **Pre-trained Model Selection:** A pre-trained model, often trained on a large-scale dataset like ImageNet for image classification tasks or Word2Vec for natural language processing tasks, is chosen based on its performance and suitability for the target task;
- **Feature Extraction:** The pre-trained model is used as a feature extractor to generate high-level representations (features) from the input data. For example, in computer vision tasks, the activations of the last few layers of a convolutional neural network (CNN) can be used as features for image classification or object detection tasks;
- **Fine-tuning:** The extracted features are then used to train a new model or adapt the pre-trained model to the target task with a smaller dataset. During fine-tuning, the parameters of the pre-trained model are updated using the target dataset, while the parameters of earlier layers may be frozen or fine-tuned to a lesser extent to prevent overfitting.

2.6 VISUAL GEOMETRY GROUP 16

VGG16 is a typical deep Convolutional Neural Network (CNN) design with numerous layers, also known as ConvNet. The abbreviation VGG stands for Visual Geometry Group. VGG-16 has 16 convolutional layers, respectively. Innovative object identification models are built using the VGG architecture. The VGGNet, created as a deep neural network, outperforms benchmarks on a variety of tasks and datasets outside of ImageNet. It also remains one of the most often used image recognition architectures today. The VGG model was first proposed by Andrew Zisserman and Karen Simonyan in 2013, and a prototype was made for the 2014 ImageNet Challenge [43]. The creators of this model evaluated the networks and increased the depth using an architecture with very small (3 × 3) convolution filters, which showed a significant improvement on the prior-art configurations. They pushed the depth to 16–19 weight

layers, making it approximately 138 trainable parameters. VGG16 is a pre-trained model used in object detection and classification algorithms that is able to classify 1000 images from 1000 different categories with 92.7% accuracy. It is one of the most popular algorithms for image classification and is easy to use with transfer learning [44].

2.6.1 VGG16 Architecture

The VGG16 model begins with an input layer that takes in the raw pixel values of the image. It then proceeds through 13 convolutional layers, labeled Conv1_1 to Conv5_3, which utilize 3x3 filters to extract features. Max-pooling layers follow several convolutional layers, reducing feature map dimensions while retaining important information. After the convolutional layers, three fully connected layers further process features and produce class probabilities. The specifics of the architecture are as follows [45]:

- **Input Layer:** Receives image data, typically 224x224 pixel RGB images;
- **Convolutional (CONV) Layers:** Comprising 13 layers, these use 3x3 filters to extract image features. The number of filters increases as layers deepen, with 64 in the first block, 128 in the second, 256 in the third and fourth blocks, and finally, 512 in the fifth;
- **Activation Function:** After each convolutional layer, a Rectified Linear Unit (ReLU) activation function is applied to introduce non-linearity;
- **Max Pooling Layers:** These layers, placed at the end of each convolutional block, downsample feature maps using a 2x2 window with a stride of 2, reducing feature map dimensions and enhancing network robustness;
- **Fully Connected (FC) Layers:** Following the convolutional blocks, three fully connected layers are employed. The first two have 4096 neurons each, and the third has 1000, corresponding to the classes in the ImageNet challenge. These layers combine features learned by convolutional layers to make classification decisions;
- **Softmax Layer:** The final layer converts the output of the last fully connected layer into a probability distribution over the 1000 class labels;
- **Dropout:** Applied to the fully connected layers during training, dropout randomly sets a fraction of input units to zero, helping to prevent overfitting.

VGG-16's architecture, especially its consistent use of 3x3 filters and deep structure, demonstrated the effectiveness of depth in CNNs for image recognition tasks.

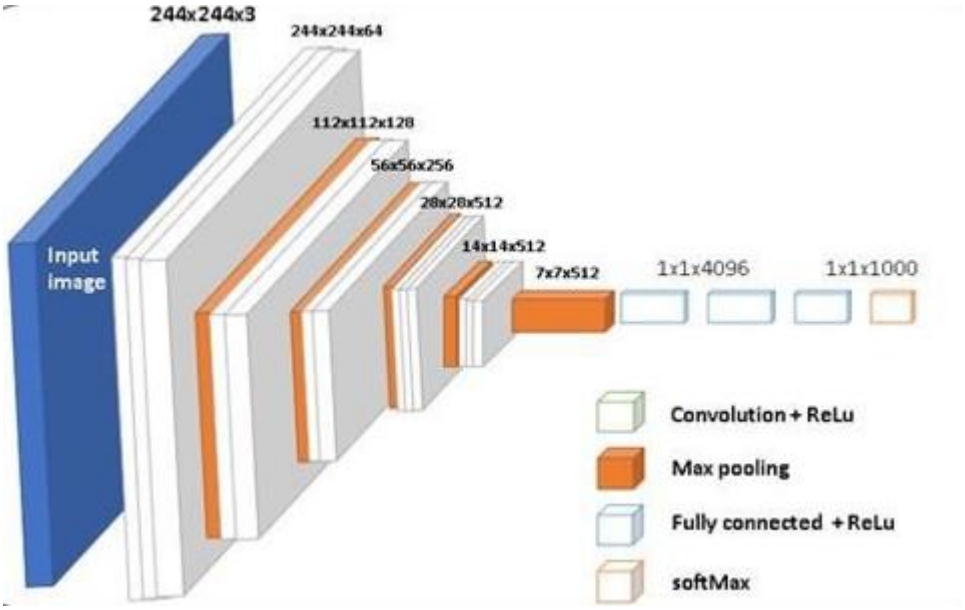


Figure II- 4 : VGG16 model Architecture

2.7 CONCLUTION

In this chapter, we provide a foundational understanding of deep learning and its application in computer vision tasks, with a specific focus on CNNs and their architecture for object detection. We have explained how CNNs extract meaningful features from images, making them ideal for tasks like object recognition and localization. We also explained the concept of transfer learning, highlighting its ability to leverage pre-trained models like VGG16, which is a pre-trained CNN model. This approach significantly reduces training time and resources, particularly when dealing with limited datasets. By fine-tuning these pre-trained networks, we can effectively adapt them to new object detection problems.

CHAPTER III: EXPLAINABLE ARTIFICIAL INTELLIGENCE

3.1 INTRODUCTION

Recently, artificial intelligence and machine learning in general have demonstrated remarkable performances in many tasks, and image processing is one of them, especially with the advent of deep learning. Along with research progress, they have encroached upon many different fields and disciplines. Some of them require a high level of accountability and, thus, transparency; the medical sector is an example. Explanations for machine decisions and predictions are thus needed to justify their reliability. This requires greater interpretability, which often means we need to understand the mechanisms underlying the algorithms. Unfortunately, the black box nature of deep learning is still unresolved, and many machine decisions are still poorly understood. In this chapter, we provide a review of interpretabilities.

3.2 EXPLAINABILITY

Explainability in deep learning refers to the ability to understand and explain the decisions made by complex neural network models. As deep learning architectures grow in size and complexity, understanding the rationale behind a model's predictions becomes increasingly challenging. Explainability aims to provide insights into how these models arrive at their decisions, enhancing transparency, interpretability, and trustworthiness. It involves answering critical questions about a model's behavior, such as why it made a specific prediction or classification, which features or patterns in the input data influenced its decision, and how confident the model is in its predictions. Achieving explainability involves various techniques and approaches. It is crucial in domains where model decisions have significant consequences, such as healthcare, finance, and autonomous systems [46].

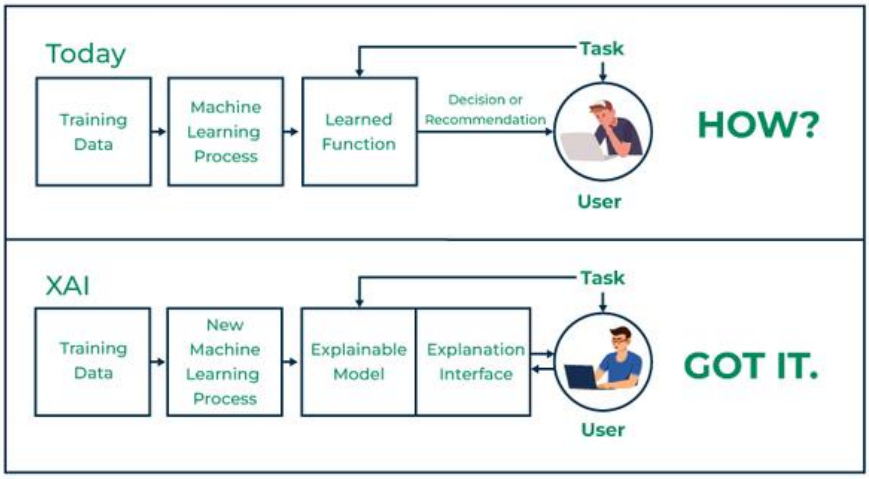


Figure III- 1 : Explainable Artificial Intelligence Concept

3.3 EXPLAINABILITY AI METHODS

Explainable AI (XAI) methods are a collection of techniques that aim to make the decision-making processes for artificial intelligence models, especially complex ones like deep learning models, more understandable by humans. These methods are important for building trust and ensuring responsible use of AI, particularly in fields like healthcare where transparency is paramount.

3.3.1 Local Interpretable Model-agnostic Explanations

Local Interpretable Model-Agnostic Explanations (LIME) is a method used to make machine learning models more understandable by humans. It explains the predictions of any black-box model by approximating it locally with an interpretable model. Essentially, LIME generates a new, simpler model that mimics the behavior of the complex model in the vicinity of a particular prediction. By doing so, it highlights which features are most influential for that specific prediction, allowing users to gain insight into the decision-making process of the model without needing to understand its internal workings. This helps in building trust and verifying the reliability of machine learning systems [47].

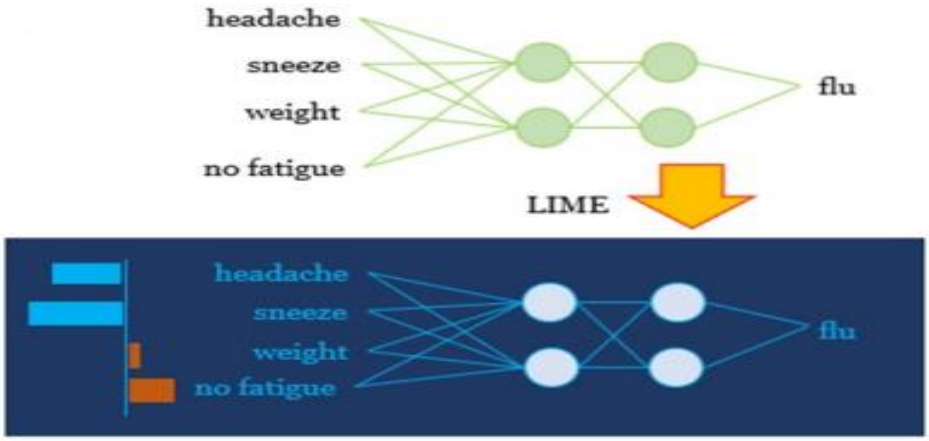


Figure III- 2 : Local Interpretable Model-Agnostic Explanations method

3.3.2 SHapley Additive exPlanations

SHapley Additive exPlanations (SHAP) is a method used to explain the output of machine learning models based on cooperative game theory. It assigns each feature an importance value, representing its contribution to the model's prediction. SHAP values are calculated using Shapley values, which fairly distribute the "payout" (prediction) among all features, ensuring that the contribution of each feature is fairly assessed considering all possible combinations. This approach provides a consistent and interpretable way to understand how

each feature influences the prediction, making it easier for users to interpret complex models accurately and transparently [48].

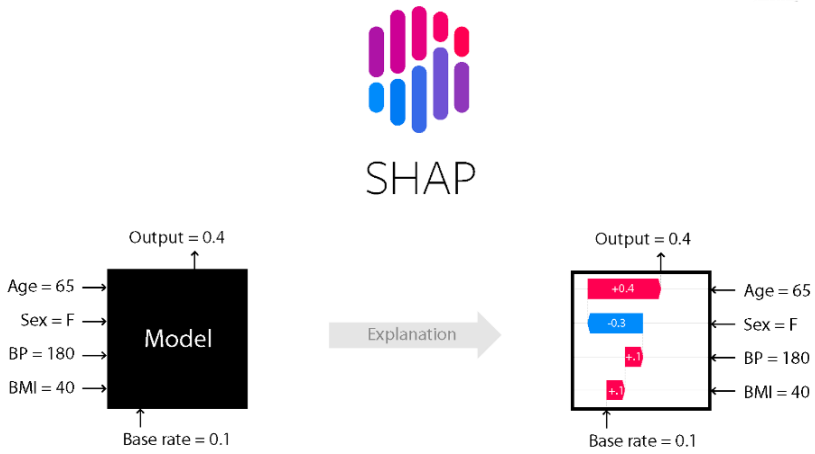


Figure III- 3 : SHapley Additive exPlanations method

SHAP and LIME are Python libraries for model explainability. SHAP uses Shapley values for feature influence scoring, considering all possible predictions and combinations of inputs. It guarantees properties like consistency and local accuracy. LIME builds sparse linear models around each prediction to explain how the black box model works in that local vicinity. However, LIME is faster than SHAP due to its ability to compute Shapley values quickly. SHAP is not optimized for all model types yet, and its tree explainer runs fast on trees like gradient boosted trees and scikit-learn, but is prohibitively slow for k-nearest neighbor models.

3.3.3 Feature maps

Feature maps are the raw data used by the interpretation methods, and the Saliency maps, Class Activation Mapping (CAM), and Gradient-weighted Class Activation Mapping (Grad-CAM) are techniques used to interpret and visualize the important regions in an input that a neural network uses for making predictions. These methods help in understanding and visualizing how CNNs make decisions [49][41].

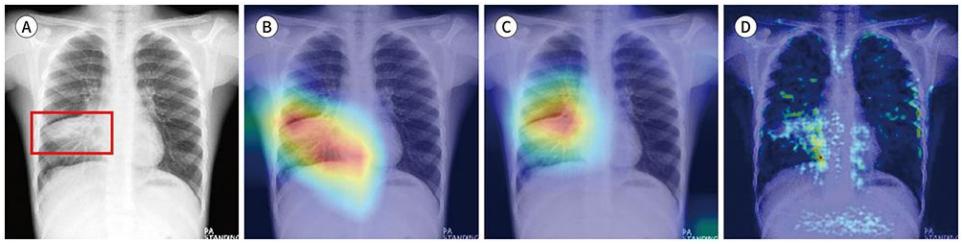


Figure III- 4 : (A) chest X-ray image of a patient with pneumonia, (B) CAM, (C) Grad-CAM, (D) saliency map

3.3.3.1 Saliency Maps

Saliency maps are a general approach to highlighting which parts of an input (e.g., an image) are most important for the model's decision. They are computed by taking the gradient of the model's output (e.g., class score) with respect to the input. This gradient indicates how much a small change in each pixel would affect the output, thus identifying the important pixels. In other words, it highlights the most influential pixels in an input image by computing the gradient of the output class score with respect to the input pixels, providing a detailed, pixel-level understanding of what drives the model's decision [49].

3.3.3.2 Class Activation Mapping

Class Activation Mapping (CAM) specifically focuses on identifying the regions of an image that are important for predicting a specific class. It generates visual explanations by using the output of the global average pooling layer before a convolutional layer. It creates a heatmap showing important regions for a specific class by computing a weighted sum of these feature maps, where the weights come from the fully connected layer corresponding to the target class [49].

3.3.3.3 Gradient-weighted Class Activation Mapping

Gradient-weighted Class Activation Mapping (Grad-CAM) Grad-CAM extends CAM by using the gradients of the class score with respect to the feature maps of any convolutional layer to produce a coarse localization map. Not just those followed by global average pooling. This method is more flexible than CAM as it does not require architectural modifications and can be applied to any CNN architecture, highlighting the regions that are important for predicting a specific class. Grad-CAM computes the gradient of the class score with respect to the feature maps of the last convolutional layer. Average these gradients to obtain weights, then use these weights to compute a weighted sum of the feature maps [49].

3.4 EXPLAINABILITY TYPES

There is no universally accepted standard for understanding machine learning explainability. While some frameworks have been proposed, different criteria are used. Network dissection is suggested for visual representation, explainability, and quantification. Interactive websites have suggested a unified framework for studying explainability. The SHAP framework defines a unified measure of feature importance. There are two major categories: perceptive explainability and explainability. Mathematical structures, present different polarities within

explainability. Perceptive explainability is challenging as the algorithm or method used to generate visual evidence may not provide useful clues for correction. Mathematical analysis of patterns can provide high-dimensional information, but abstracting fine-grained information is not discriminative with measurable certainty [50].

3.4.1 Perceptive Explainability

Humanly perceivable interpretability defines explanations that are intuitively understood by humans, often deemed "obvious." For example, an algorithm classifying an image as a "cat" can be considered obviously interpretable if it highlights the segmented patch showing the cat. However, this approach has limitations; it doesn't fully explain the algorithm's internal workings (it remains a "black box"), or it might ignore the use of background objects in the decision-making process.

3.4.1.1 Saliency

The saliency method is a technique used in the field of deep learning, particularly in the visualization of the areas within an input image that are most influential in a neural network's decision-making process. By computing the gradient of the model's output with respect to the input image, this method identifies which pixels, when adjusted slightly, would have the most significant impact on the output. The resulting gradients can then be visualized as a saliency map—a heatmap overlaid on the original image—highlighting the regions of highest importance. This visualization helps to understand and interpret how the model processes visual information and makes decisions, making it a valuable tool for debugging and improving model performance. However, saliency maps can sometimes be noisy or difficult to interpret, prompting the development of more refined techniques to improve clarity and usefulness [50].

3.4.1.2 Signal Method

Signal methods are interpretability techniques that observe the stimulation of neurons or a collection of neurons. These methods can be used to manipulate or transform the activated values of neurons into interpretable forms, such as reconstructing an image similar to the input. For example, feature maps in a layer can activate more strongly complex features, such as human faces, while feature maps in shallower layers show simple patterns. Parameters or input data can be optimized with respect to the activation values of particular neurons using methods known as activation optimization. Feature maps and inversions for input reconstructions can be related to humanely perceptible features by defining an inverse convolution map. Inversion of

image representations within layers has been used to demonstrate that CNN layers do store important information about an input image accurately. Guided backpropagation modifies the way backpropagation is performed to achieve inversion by zeroing negative signals from both the output and input signals backwards through a layer. Activation maximization involves finding input images that optimize the activation of a neuron or a collection of neurons. Research works on activation maximization include the Modified National Institute of Standards and Technology (MNIST) dataset and GoogLeNet, which uses a regularization function to visualize activation within a higher-level organization and semantically more meaningful arrangements [50].

3.4.1.3 Verbal Explainability

Explainability in machine learning models can be achieved through verbal chunks that humans naturally understand, such as sentences indicating causality or logical statements like "if A, then B." Models that produce such logical statements are considered inherently explainable. Explainability methods can be viewed as symbolic and relational systems, with specific applications in the medical field and Decision sets, or rule sets, are another approach to explainability, where each rule is a clause in disjunctive normal form (DNF) mapping inputs to outputs. For example: "rainy and grumpy or calm → dairy or vegetables." The example formally written (rainy ∧ grumpy) ∨ calm → dairy ∨ vegetables is a rule set line. The interpretability of these explanations is influenced by cognitive chunks (clauses of inputs in DNF), explanation size (length or number of lines in the rule set), and minimally by variable repetition. Tools like Model to Understand Simple English (MUSE) generate explanations in the form of decision sets by approximating black-box functions and optimizing explainability metrics. In natural language processing (NLP) problems, explanations often take the form of verbal segments. For instance, an encoder-generator framework can extract phrases like "a very pleasant ruby-red-amber color" to justify a 5-star product review. The explanation is a subset of the sentence, represented as a binary sequence indicating which words are included. This subset follows a probability distribution modeled with the hidden units of a recurrent neural network. Similar segments are generated using a filter-attribute probability density function to improve the relationship between filter activations and specific attributes [50].

3.4.2 Explainability via Mathematical Structure

Mathematical structures reveal the mechanisms of machine learning and neural network (NN) algorithms. Deeper layers store complex information, while shallower layers store simpler

information. Information theory studies interpretability by considering the information bottleneck principle. The rich ways mathematical structures enhance interpretability contribute to a comprehensive view of algorithm interpretability, potentially unifying different views under a coherent framework in the future [50].

3.4.2.1 Pre-defined Model

Pre-defined models use established mathematical formulas and algorithms to interpret and understand the behavior and decisions of machine learning models. Linearity is the simplest interpretable pre-defined model, which can be enhanced by introducing non-linearity when needed. Linear probes extract information from each neural network layer, generally performing better at deeper layers. General additive models, such as the Generalized Additive Model (GAM), General Linear Model (GLM), and ProtoAttend, are inherently interpretable. Content-specific machine learning algorithms, designed based on existing knowledge or empirical evidence, embed interpretation within the system. These can be implemented in various ways through parameter fitting. Examples include Deep Tensor Neural Networks (DTNN) for quantum many-body systems, Atomistic Neural Networks (ANNs) for quantum chemistry, Neural Networks for programmable wireless environments, TS approximation for fuzzy networks, and model-based reinforcement learning for Bayesian structures [50].

3.4.2.2 Feature Extraction

Feature extraction in explainability through mathematical structure focuses on using formal mathematical techniques to identify and understand the most relevant features that influence a model's predictions. This approach aims to provide a rigorous and transparent framework for interpreting the decision-making processes of complex machine learning models. These are the key concepts and methods involved in this approach [50]:

- **Correlation**: is a method that includes a covariance matrix and correlation coefficients after transformation by kernel functions. A kernel function transforms high-dimensional vectors such that the transformed vectors better distinguish different features in the data. It quantifies how changes in one variable are associated with changes in another. Principal Component Analysis (PCA) is an example of a kernel function that transforms vectors into principal components (PC) that can be ordered by the eigenvalues of a singular-value-decomposed (SVD) covariance matrix. Correlation is widely used in data analysis to identify patterns, trends, and

potential causal relationships between variables, facilitating insights in fields such as finance, medicine, and the social sciences;

- **Singular Vector Canonical Correlation Analysis (SVCCA)**: a method that compares and analyzes neural network representations by combining Canonical Correlation Analysis (CCA), which finds linear relationships between two sets of variables, with Singular Value Decomposition (SVD) to reduce dimensionality. It aligns the learned features of two neural network layers and quantifies their similarity, providing insights into network internal representations and their evolution across layers or models;
- **Generative adversarial networks (GAN):** are machine learning frameworks that create synthetic data samples resembling a training dataset. They consist of a generator and a discriminator trained in a game-like setup. The generator creates fake data samples, while the discriminator evaluates the samples' authenticity. GANs are used in image generation, video synthesis, and other creative applications. Studies related to interpretabilities exist, such as using encoder-decoder systems to perform multi-stage PCA;
- **Clustering:** is another method used to cluster input images based on the activation of neurons in a network. The core idea relies on the distance between objects being considered, which may appeal to human learning under the Law of Association. Clustering simply presents their similarity more sensibly in similar domains or subsets thereof.

3.4.2.3 Sensitivity

Sensitivity in explainability refers to how explanation methods that rely on localization and gradients react to small changes or perturbations in input data or model parameters. These methods rely on the notion of small changes in calculus and the neighborhood of a point in metric spaces. Some methods rely on the locality of some input x, such as a model predicting f(x) accurately for some x. Meta-predictors are interpretability methods that emphasize the importance of the variation of input x to the neural network in explaining a network. They define explanation and local explanation in terms of the response of Blackbox to some input. Techniques like LIME, TCAV, and gradient-based methods assess the local faithfulness of models by examining how slight input alterations affect predictions. While these methods help identify important features, they can also reveal model instability if minor changes lead to

significant shifts in predictions. Addressing this sensitivity is crucial for developing robust, reliable interpretability techniques that provide consistent explanations, ensuring model predictions remain trustworthy even with slight variations in data or training conditions [50].

3.5 XAI IN MEDICAL FIELD RELATED WORKS

Machine learning (ML) has recently advanced significantly in the medical field, particularly in automated diagnosis and prognosis. Various challenges have driven research into using ML and AI methods, with successful models like U-Net for medical segmentation [51]. However, U-Net and similar deep learning models are "black boxes," lacking explainability.

In medicine, explainability is crucial due to the high stakes, including risk and responsibility factors unique to the field. The inability of these models to provide accountability poses ethical and safety risks, which could be catastrophic if exploited maliciously. Thus, many studies now focus on improving explainability in medical AI, providing reviews, and raising awareness about the importance of explainability. Some argue that a degree of opaqueness is acceptable if the results are empirically verified. Researchers are categorizing explainability methods by their application and the organs targeted by diseases, though more comparative studies are needed to draw informative conclusions.

3.5.1 Perceptive explainability

Perceptive explainability identifies the ability of AI systems to provide clear, understandable explanations for their decisions and predictions in healthcare settings. It involves creating models that not only deliver accurate results but also make their reasoning transparent to medical professionals. This enhances trust, aids in clinical decision-making, and ensures that the use of AI aligns with the standards and requirements of medical practice.

3.5.1.1 Saliency

Several articles highlight the application of saliency methods for medical image explainability. CheXpert [52] uses GradCAM to visualize pleural effusion in radiographs, while CAM aids in brain tumor grading. Guided Grad-CAM and feature occlusion provide complementary heatmaps for Alzheimer’s disease classification [53]. There has even been a direct comparison between the feature maps within CNN and skin lesion images, overlaying the scaled feature maps on top of the images as a means to improve interpretability. Basic tests on GradCAM and SHAP for melanoma [54] classification in dermoscopy images highlight the need for improved heatmaps before practical use.

Other research focuses on methodological improvements. RespondCAM [55] produces heatmaps for 3D cellular images, and Multi-layer Class Activation Map (MLCAM) [56] is used for glioma localization. Multi-instance (MI) aggregation and CNN classify breast tumor tissue microarray (TMA) images, using super-pixel maps for visual interpretability. Further, activation maps assess the impact of image corruption on regions of interest, and an autofocus module enhances segmentation interpretability in pelvic CT and brain MRI using attention mechanisms [57]. It uses attention mechanism and improves it with the adaptive selection of scale with which the network ”sees” an object within an image. With the correct scale adopted by the network while performing a single task, a human observer analyzing the network can understand that a neural network is properly identifying the object, rather than mistaking the combination of the object plus the surrounding area as the object itself.

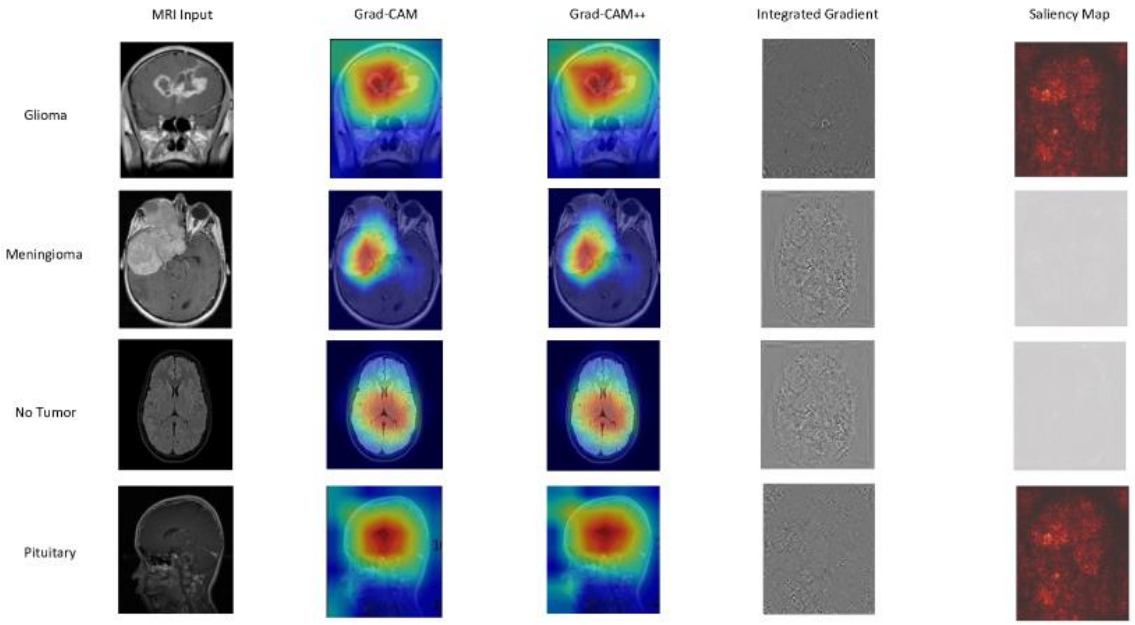


Figure III- 5 : Feature maps for brain tumor classification

3.5.2 Explainability via Mathematical Structure

Explainability via Mathematical Structure indicates the use of mathematical models and frameworks to make AI systems' decisions and predictions in healthcare more transparent and understandable. By leveraging mathematical structures, such as equations, algorithms, and logical rules, this approach aims to provide clear, precise explanations that can be easily interpreted by medical professionals, thereby improving trust, and aiding in clinical decision-making.

3.5.2.1 Pre-defined Model

Models facilitate explainability by showing how variables influence outcomes, often using parametric models designed based on observed patterns. For instance, a kinetic model [58] u for cerebral blood flow (CBF) is used in neural network training to improve image quality by denoising perfusion-weighted images. Linear models are considered interpretable due to their simplicity [99], such as logistic regression, which identifies a negative correlation between asthma and pneumonia death risk.

3.5.2.2 Feature Extraction

Explainability in neural networks can be achieved through various diverse methods that illustrate the breadth of approaches to making neural networks and complex models interpretable, each tailored to specific types of data and applications. As detailed in the following summaries:

- **Vanilla CNN and Separable Models:** A vanilla CNN can be made interpretable by using a separable model, achieved by polynomial-transforming scalar variables, thus generating interpretable weights [59];
- **FMRI Analysis with Graph-Based Methods:** Functional MRI is analyzed using correlation-based functional graphs, which are clustered into super-graphs. These super-graphs, consisting of subnetworks, are deemed interpretable and processed through a convolutional layer [60];
- **Correlation-Based Feature Extraction:** Discrete wavelet transform (DWT) is used for feature extraction from EEG data for epilepsy classification [61]; involving a fuzzy relation similar to correlation coefficients. Frame singular value decomposition (F-SVD) [62] is also introduced for EMG data classification, involving various processing techniques;
- **Clustering techniques:** Are employed for explainability in various ways. Variational autoencoders (VAE) predict hypertrophic cardiomyopathy (HCM) [63] using Laplacian Eigenmap for dimensionality reduction and explainability. A K-nearest-neighbor method [64] customized with CNN and triplet loss, is used for melanoma classification by visually comparing queried images with training images;
- **Sensitivity-based methods:** The Unit Ball Surface Sampling (UBS) [65] metric addresses shortcomings in the Br score for classifying mammographic nodules.

Guideline-based Additive eXplanation (GAX) [66] employs LIME-like perturbation and SHAP for diagnosing CT lung images, and it is compared with other explainability methods like Grad-CAM and SHAP.

This table below mentions the related works in explainability and categorizes them by the organs affected by the diseases. The rows are arranged according to the focus of the explainability [49]:

	Organ	**Paper**
APPLICATION	Brain	[53], [58], [67], [68]
	Lung	[52], [69]
METHODOLOGY	Brain	[70], [57], [71], [56], [72],[62],[60], [61]
	Breast	[73]
	Skin	[64]
	Heart	[63]
COMPARISON	Brain	[74]
	Lung	[66]
	Skin	[54]

Table III- 1 : Related works

3.6 EXPLAINABILITY CRITERIA

To evaluate the quality and effectiveness of explanations provided by machine learning models. There are principles or criteria that ensure that the explanations are not only accurate and trustworthy but also actionable and understandable for users, and these are the most important criteria:

- Reliability ensures that explanations are consistent and dependable, involving the stability and robustness of the explanations so they do not change drastically with small changes in input data. It also reflects the faithfulness of the explanation to the actual model behavior.
- Causality focuses on the ability of explanations to reflect causal relationships rather than mere correlations, helping to understand whether and how certain features causally influence the model's predictions. This understanding is important for making decisions based on the model's outputs and for grasping the underlying mechanisms.

- Usability pertains to how easily users can understand and act on the explanations, involving the clarity, relevance, and accessibility of the explanations. It ensures that the explanations are actionable, providing insights that users can use to make informed decisions.

3.7 LOCALIZATION FOR EXPLAINABILITY

Involves using techniques such as bounding boxes to highlight specific regions of input data (e.g., images) that are most influential in a model's decision-making process. Bounding boxes are drawn around these important areas to visually represent the model's focus. The accuracy of this localization is often measured using Intersection over Union (IoU), which quantifies the overlap between the predicted bounding box and the ground truth, ensuring that the identified regions accurately correspond to the significant features relevant to the model's output.

3.7.1 Bounding BOX

Bounding boxes are rectangular boxes that surround an object and specify its position. They are used in computer vision and image processing to specify the location of objects within an image. They are a fundamental component in various tasks such as object detection, image segmentation, and tracking. where the aim is to identify the position and type of multiple objects in the image [75]. Depending on the convention followed, the main parameters that specify a bounding box are:

- **Class:** What is in the object inside the box? For example, car, truck, person etc.;
- **(x1, y1):** Corresponds to the x and y coordinate of the top left corner of the rectangle;
- **(x2, y2):** Corresponds to the x and y coordinate of the bottom right corner of the rectangle;
- **(xc, yc):** Corresponds to the x and y coordinate of the center of the bounding box;
- **Width:** Represents the width of the bounding box;
- **Height:** Represents the height of the bounding box;
- **Confidence:** Indicates how likely the object is present in that box. For example, a confidence level of 0.9 would indicate that there is a 90% chance that object exists in that box.

3.7.2 Intersection over Union

Intersection over Union (IoU) is a popular metric to measure localization accuracy and compute localization errors in object detection models. It calculates the amount of overlap between two bounding boxes—a predicted bounding box and a ground truth bounding box [76]. The greater the region of overlap, the greater the IOU [77]. IoU used in non-max suppression, which is used to eliminate multiple boxes that surround the same object based on which box has the highest confidence. It's used in numerous computer vision and object detection applications, such as autonomous vehicles, security systems, and medical imaging, where we train a model to output a box that fits perfectly around an object [76].

The calculation of the overlap between the ground-truth bounding box and the predicted bounding box in the numerator. Mathematically written as:

$$Intersection\ over\ Union\ (IoU) = \frac{|A\ \cap\ B|}{|A|\ \cup\ |B|}$$

But for binary classification, it is written as:

$$Intersection\ over\ Union\ (IoU) = \frac{TP}{TP + FN + FP}$$

Where:

- TP = True Positive.
- FN = False Negative
- FP = False Positive

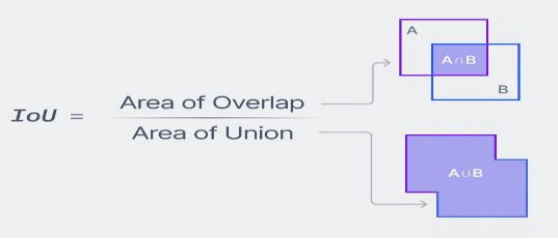


Figure III- 6 : The visual representation of IoU

The IoU score will be high if there is much overlap between the anticipated and ground truth boxes. In contrast, a low overlap will result in a low IoU score. An IoU score of 1 indicates a perfect match between the projected box and the ground truth box, whereas a score of 0 means no overlapping between the boxes.

Figure III- 7 : IoU score representation

The IoU metric is helpful since it offers a numerical assessment of how well a model identifies items in an image. Additionally, while training the model, we can choose a minimum IoU score needed for a predicted box to be regarded as an accurate positive detection, which allows using IoU to set a threshold for object detection. The trade-off between detection accuracy and false positives can be managed by choosing a suitable threshold. The IOU returned value is of type float, which lies between 0 and 1. There is no one-size-fits-all recommended threshold for IoU, as it largely depends on the specific object detection task and dataset. However, a common threshold used in practice is 0.5, meaning that a predicted box must have an IoU of at least 0.5 with a ground truth box to be considered a true positive detection. This threshold can be adjusted based on the desired trade-off between precision and recall. For example, increasing the threshold would result in fewer false positives but may also miss some true positives. It's essential to evaluate the model's performance on a validation set using different IoU thresholds to choose the most appropriate one for the task.

3.8 CONCLUSION

This chapter explored the Explainable Artificial Intelligence (XAI), particularly within the medical field. We emphasized the importance of understanding how AI models arrive at their decisions, especially in healthcare settings where patient well-being is paramount. We discussed various XAI methods and their advantages, including techniques that reveal the inner workings of the model and those that focus on explaining specific predictions like bounding boxes and Intersection over Union (IoU) for object detection. By employing these methods, we can gain valuable insights into the model's reasoning, foster trust, and facilitate clinical adoption by healthcare professionals.

CHAPTER IV: EXPLAINING PNEUMONIA DISEASE THROUGH CAUSAL LEARNING

4.1 INTRODUCTION

After presenting the theory of deep learning and explainability in the previous chapters, this part will be devoted to the implementation of a system for the detection of pneumonia using the VGG16 model and the causal explanation of the convolutional neural network method. For this, we will use a group of images selected from the database, which contains a collection of pneumonia chest x-ray images.

4.2 DATASET OVERVIEW

For this study, we utilized two publicly available dataset on Kaggle, “RSNA Pneumonia Detection Challenge”, and “Chest X-Ray Images (Pneumonia)” by Paul Timothy Mooney.

The “RSNA Pneumonia Detection Challenge” dataset comprises 26,684 chest X-ray images in DICOM format. This format includes comprehensive metadata such as patient information, image acquisition details, and pixel data. The dataset is labeled to indicate whether the images contain pneumonia, providing a clear classification task for developing and testing machine learning models for pneumonia detection [78].

To work with this dataset, we converted the images from DICOM format to JPG. This conversion is necessary because many common image processing libraries and tools are optimized for formats like JPG, facilitating easier manipulation and analysis of the data for training and testing machine learning models for pneumonia detection. For this part we utilize this code:

```python
# Function to convert and copy files
def convert_and_copy(filenames, dest_dir):
    for filename in filenames:
        # Read the DICOM file
        ds = pydicom.dcmread(os.path.join(dicom_dir, filename + '.dcm'))
        # Get the image array
        img = ds.pixel_array
        # Normalize the image to the range [0, 255] if necessary
        img = cv2.normalize(img, None, 0, 255, cv2.NORM_MINMAX)
        # Ensure the image has 3 channels
        if len(img.shape) == 2:  # grayscale image
            img = cv2.cvtColor(img, cv2.COLOR_GRAY2BGR)
        # Create the destination directory if it doesn't exist
        os.makedirs(dest_dir, exist_ok=True)
        # Write the JPEG image
        cv2.imwrite(os.path.join(dest_dir, filename + '.jpg'), img)
```

Figure IV- 1 : Function to convert the format of the images

The table below illustrates the structure of our learning base and the classes used:

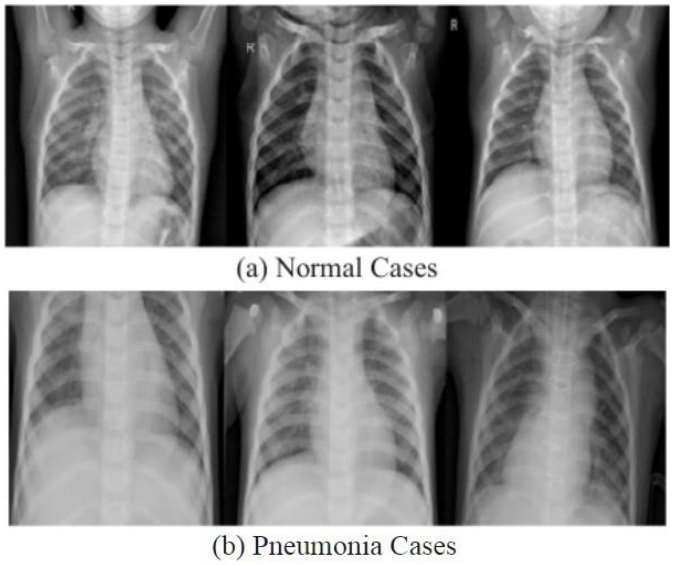
CATEGORY	TRAINING	TEST	VALIDATION	TYPE
PNEUMONIA	4914 images	1349 images	1349 images	JPG
NORMAL	14470 images	3101 images	3101 images	JPG

Table IV- 1 : "RSNA Pneumonia Detection Challenge" Dataset Overview

The "Chest X-Ray Images (Pneumonia)" dataset consists of 5,863 CXR images categorized into two classes: Normal and Pneumonia. The images are in JPG format and are divided into training, testing, and validation sets. This dataset is valuable for developing and testing machine-learning models for pneumonia detection and diagnosis. It is intended for educational and research purposes [79]. Chest X-ray images (anterior-posterior) were selected from retrospective cohorts of pediatric patients of one to five years old from Guangzhou Women and Children's Medical Center, Guangzhou. All chest X-ray imaging was performed as part of the patients' routine clinical care [80]. The table below illustrates the structure of our learning base and the classes used:

CATEGORY	TRAINING	TEST	VALIDATION	TYPE
PNEUMONIA	3875 images	390 images	8 images	JPG
NORMAL	1341 images	234 images	8 images	JPG

Table IV- 2 : "Chest X-Ray Images (Pneumonia)" Dataset Overview



(a) Normal Cases

(b) Pneumonia Cases

Figure IV- 2 : Examples from the datasets

4.3 MODEL ARCHITECTURE

We employ a Convolutional Neural Network (CNN) architecture, specifically VGG16, for our image classification task. The model consists of multiple convolutional layers, strategically designed to enhance classification performance. Initially, we load the VGG16 model with pre-trained weights from ImageNet, excluding the top layers. The input shape is set to (224, 224, 3). Next, we introduce custom fully connected layers: flattening the output, adding a dense layer with 64 units and ReLU activation, followed by a dropout layer with a rate of 0.33. Finally, we append a dense layer with 1 unit and sigmoid activation, named 'final_dense'. To fine-tune the model, we freeze all layers in the VGG16 base. After configuring these layers, we define our model and print the summary as illustrated in Table 3.2. This setup serves as a solid foundation for pneumonia detection in chest X-rays.

Layer (type)	Output Shape	parameter
input_1 (InputLayer)	(None, 224, 224, 3)	0
block_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
average_pooling2d (AveragePo	(None, 3, 3, 512)	0
flatten (Flatten)	(None, 4608)	0
dense (Dense)	(None, 64)	294976
dropout (Dropout)	(None, 64)	0
final_dense (Dense)	(None, 1)	65

Table IV- 3 : Architecture of the Proposed Model for VGG16

4.4 EXPLANATION FRAMEWORK

In this section, we will explain each part of the process, including dataset augmentation, training outcomes, and the application of the CexCNN method. It will also cover the generation of heatmaps, both prior to and following the application of CexCNN. This comparative analysis will illustrate the method's role in improving interpretability.

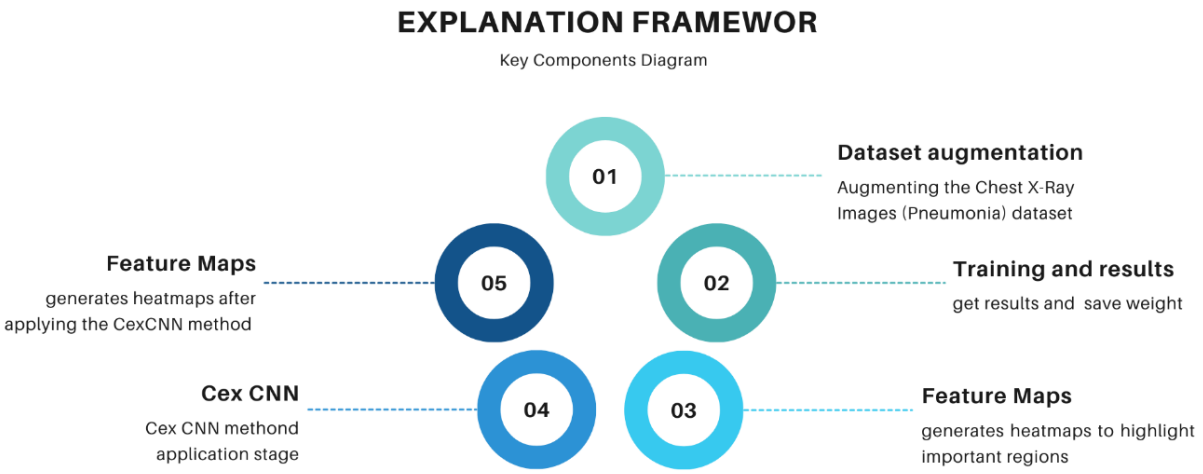


Figure IV- 3 : Framework explanation

4.4.1 Dataset augmentation

To enhance the diversity and effectiveness of the dataset, we applied data augmentation techniques during the preprocessing phase. The training data was augmented using the ImageDataGenerator class, with transformations such as rescaling, shear range, zoom range, and horizontal flip to create varied versions of the original images. Specifically, images were rescaled by a factor of 1./255, sheared by 0.2, and zoomed by 0.2, with horizontal flips enabled to introduce variability. The training, validation, and test data were loaded with target sizes set to 224x224 pixels. The training generator processed batches of 64 images with binary class mode and shuffling enabled. Similarly, the testing generator utilized a batch size of 64, while the validation generator used a smaller batch size of 16. Both testing and validation datasets did not shuffle the data. This strategic augmentation and preprocessing pipeline aimed to enhance the model's robustness and generalization capabilities by exposing it to a broader range of image variations.

4.4.2 Training and results

Training of VGG16 model is performed on Python 3.10.13 on Kaggle. The model is trained using TensorFlow (v.1.13.1). The model is trained with Adam optimizer.

The "Chest X-Ray Images (Pneumonia)" dataset: VGG16 model did well with a 92% accuracy. This high level of accuracy suggests that the model is highly effective in distinguishing between 'PNEUMONIA' and 'NORMAL' cases in chest X-rays. Also, a 91% precision, 97% recall, and a 94% f1-score.

The model has exhibited remarkable accuracy in its predictions, as demonstrated by the high confidence levels in its probability scores. This indicates that the model's predictions are not only reliable but also reflect a true understanding of the underlying patterns in the data, confirming its effectiveness in diagnosing conditions such as pneumonia from chest X-ray images.

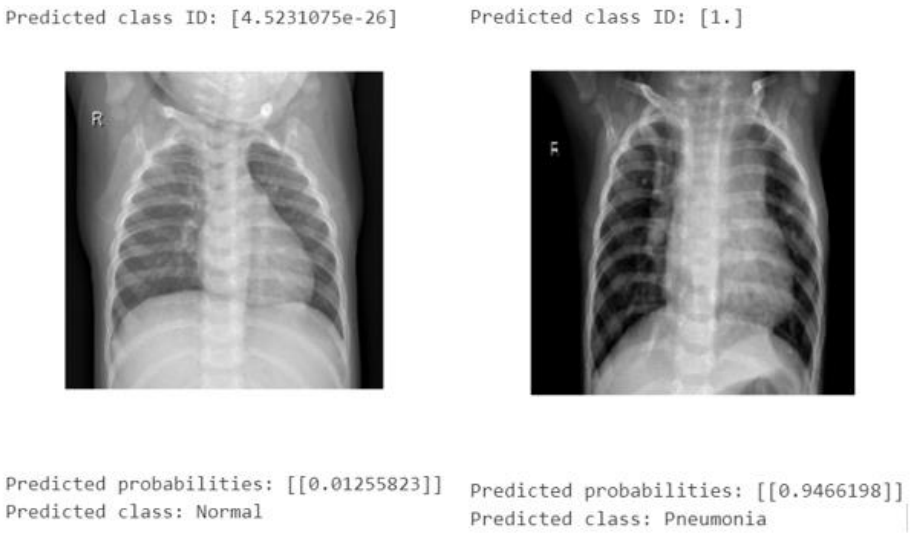


Figure IV- 4 : Examples of class prediction for the "Chest X-Ray Images (Pneumonia)" dataset

The "RSNA Pneumonia Detection Challenge" dataset: The model has shown considerable accuracy in its predictions, achieving a confidence level of 78%. Also, a 77% precision, 89% recall, and an 81% f1-score. This indicates that while there is room for improvement, the model can still reliably identify patterns in the data and make reasonably accurate predictions. Its performance in diagnosing conditions such as pneumonia from chest X-ray images demonstrates a solid understanding of the underlying patterns.

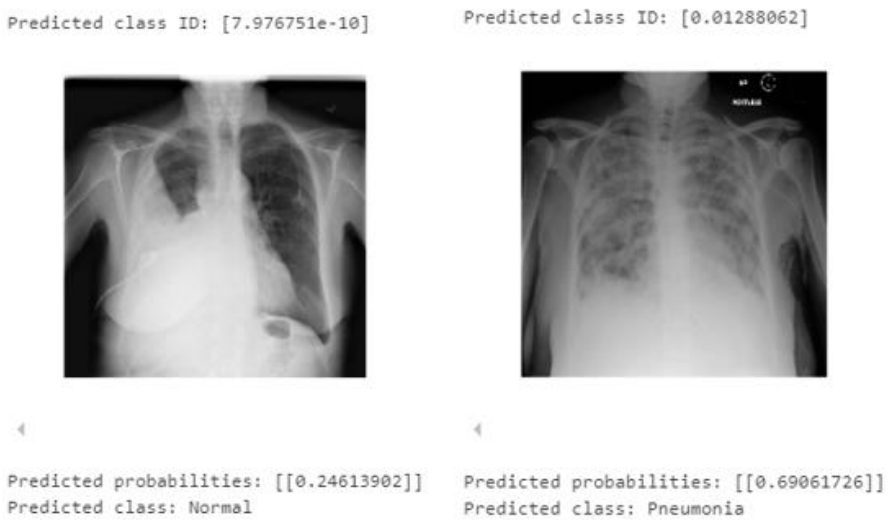


Figure IV- 5 : Examples of class prediction for the "RSNA Pneumonia Detection Challenge" dataset

4.4.3 Visualizing Heatmap and Bounding Box before CexCNN Method

In this section, we generate heatmaps which are known as feature maps to highlight important regions for a chest X-ray (CXR) image selected from the dataset before applying the CexCNN method. This feature maps indicates the areas on a chest X-ray that the model focuses on to detect pneumonia. But as we see it is not very accurate in pinpointing pneumonia-affected areas.

The "Chest X-Ray Images (Pneumonia)" dataset: The most intense activation is shown in yellow to red, suggesting this is the primary region of interest.

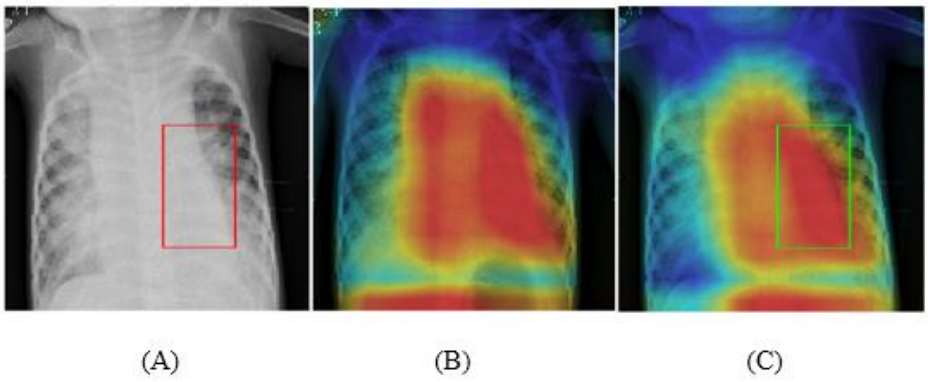


Figure IV- 6 : (A) Pneumonia chest X-ray, (B) CAM, (C) Bounding box on CAM of "Chest X-Ray Images (Pneumonia)" dataset before the CexCNN method

The "RSNA Pneumonia Detection Challenge" dataset: The most intense activation is shown in green to blue, suggesting this is the primary region of interest.

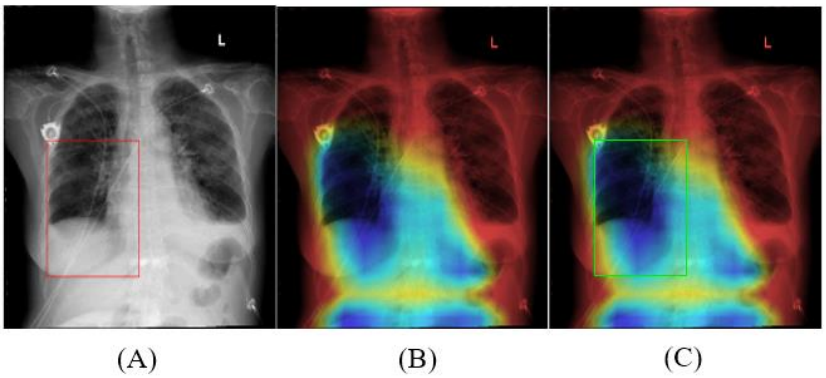


Figure IV- 7 : (A) Pneumonia chest X-ray, (B) CAM, (C) Bounding box on CAM of "RSNA Pneumonia Detection Challenge" dataset before the CexCNN method

4.4.4 Causal explanation of CNN

CexCNN (Causal Explanation of Convolutional Neural Networks) is a causal explanation technique that combines the notions of causality, responsibility, and blame to provide insights into the decision-making process of Convolutional Neural Networks (CNNs). The key concept of CexCNN is identifying the actual causes of a CNN's decision, which are the filters in each convolutional layer. The causal learning process in CexCNN assigns a degree of responsibility to each filter based on its importance to the related class. These responsibilities are then projected back to the input image to compute the "blame" for each region, indicating the most important explanations for the CNN's decision. Since filters are identified at each convolutional layer, and these layers have a hierarchical structure, the filters at lower levels have a causal effect on the output of the last layer. This suggests that CNNs inherently express causality in their decision-making process [81].

4.4.4.1 Causality code overview

To overcome limitations in advanced instructions and configurations for implementing causal explanation CNN techniques on Kaggle, we transitioned our model training to Google Colab, benefiting from its flexibility and compatibility. By importing pre-trained weights from Kaggle, we resumed model refinement and applied the Causal Explanation CNN method [82]. This strategic shift empowered deeper insights into our model's predictions, enhancing result robustness and reliability. Leveraging the strengths of both platforms facilitated efficient model development while providing meaningful explanations for our model's decisions.

This code iterates through a list of the image filenames, loads each image, preprocesses it, and predicts class probabilities using a pre-trained model. It stores the original filename and

calculates the accuracy of predictions. Additionally, it initializes a dictionary for weight-related information and sets the learning phase to 0 for prediction:

```
imgCount = 0
for imgFileName in imagesFileNames:
    imgPath = folder + imgFileName
    img = image.load_img(imgPath, target_size=(224, 224))
    x = image.img_to_array (img)
    x = np.expand_dims (x, axis=0)
    x = preprocess_input (x)
    preds = model.predict(x)
    originalName = imagesFileNames[imgCount]
    if '.jpg' in originalName:
        originalName = originalName.replace('.jpg','')
    orig_acc = preds[0][0]
    print('origin acc', orig_acc)
    from keras.layers import Lambda
    weights_dic = {}
    K.set_learning_phase(0)
```

Figure IV- 8 : Prediction Class Probabilities Code

This loop iterates over 512 values, initializing each element of an array to 0. It defines a custom layer using a Lambda function, which scales the tensor by the corresponding array element. This custom layer is inserted into the model at position 18, which is after the last layer of the model. The modified model predicts class probabilities for an image, and the accuracy difference from the original prediction is calculated and stored to assess the impact of pruning certain filters from the model. Finally, the array element is set to 1, indicating an iterative adjustment of filter weights to observe their impact on accuracy:

```
for i in range(512):
    arryWeights_last_After[i]=0
    filt = i
    def custom_layer_last(tensor):
        return tensor * arryWeights_last_After
    lambda_layer = Lambda (custom_layer_last, name="lambda_New")
    new_model = insert_intermediate_layer_in_keras (model, 18, lambda_layer)
    preds = new_model.predict(x)
    New_accu = preds[0][0]
    print ('acc after purning (%.2f%%)', New_accu)
    print('difference: ', (orig_acc - New_accu))
    tuple = orig_acc - New_accu, 1
    weights_dic[filt] = tuple
    arryWeights_last_After[i]=1
```

Figure IV- 9 : Iterative Filter Pruning Impact Code

This code sorts the dictionary “weights_dic” based on its values and stores the sorted result in “weights_dic_sort”. It then updates “dctOfAllDictionClasses” with the sorted dictionary paired with the original image name. Additionally, it saves two dictionaries, “lstOfAllDictionClasses” and “dctOfAllDictionClasses”, into JSON files:

```
weights_dic_sort = sorted (weights_dic.items (),
                           key=lambda kv: kv[1],
                           reverse=True)
pairKeyValue = {originalName: weights_dic_sort}
dctOfAllDictionClasses.update(pairKeyValue)
    import json
    with open('FIltersRespAll', 'w') as fout: json.dump(str(lstOfAllDictionClasses),
                                                        fout)
    with open('/content/drive/My Drive/Colab Notebooks/Filters', 'w') as fout:
        json.dump(str(dctOfAllDictionClasses), fout)
    print('Last Image ID:', imagesFileNames[imgCount])
    imgCount+=1
```

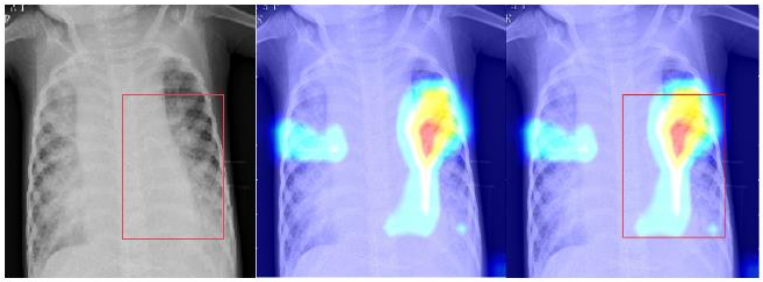
Figure IV- 10 : Sorted Weights Code

After creating a sorted dictionary where the keys are filter indices and the values are weights it assesses the impact of individual filters by calculating their importance and sorting them accordingly. This provides insights into which filters contribute most significantly to the model's predictions. Additionally, it performs filter pruning based on importance values and defines a custom layer that adjusts the contribution of individual elements in subsequent neural network layers, aiming to optimize the model's performance by selectively including or excluding certain filters. After applying these steps, a heatmap is generated.

4.4.5 Visualizing Heatmap and Bounding Box after CexCNN Method

In this section, we generate feature maps and bounding box to highlight important regions for a chest X-ray (CXR) image Selected from the dataset after applying the CexCNN method. This feature maps highlights the areas of interest with varying colors, indicating the levels of activation, which helps in understanding the model's focus areas for pneumonia detection.

The "Chest X-Ray Images (Pneumonia)" dataset: The application of the CexCNN method have refined the model's interpretability, potentially leading to more accurate and clinically relevant insights.



(A) (B) (C)

Figure IV- 11 : (A) Pneumonia chest X-ray, (B) CAM, (C) Bounding box on CAM of "Chest X-Ray Images (Pneumonia)" dataset after the CexCNN method

The "RSNA Pneumonia Detection Challenge" dataset: The same applies to the "RSNA Pneumonia Detection Challenge" dataset, where the CexCNN method similarly refines the model's interpretability and improves its accuracy and clinical relevance.

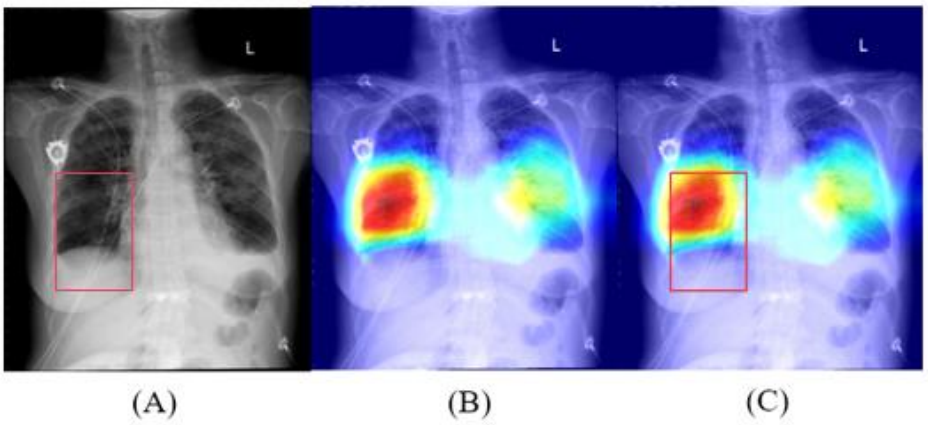


Figure IV- 12 : Pneumonia chest X-ray, (B) CAM, (C) Bounding box on CAM of "RSNA Pneumonia Detection Challenge" dataset after the CexCNN method

4.4.6 Comparing Feature Maps

Comparing the two feature maps before and after CexCNN (Figure 3-4 and Figure 3-8), the feature maps before applying the CexCNN method showed high activation in areas not necessarily indicative of pneumonia, leading to potential inaccuracies. After applying the CexCNN method, the feature maps appear more focused, with refined areas of activation that are likely more representative of the true regions affected by pneumonia. This suggests that the CexCNN method enhances the model's ability to discern relevant features for accurate diagnosis.

4.5 IMPLEMENTATION TOOLS

The following technologies were used to implement the VGG 16 system and conduct the experimental study.

4.5.1 Programming language (Python)

Python is a high-level scripting language used for text processing, system administration, and internet-related tasks, with a small core language that is easy to master and supports module additions for extensive functionality. It is a true object-oriented language available on many platforms, with even a Java-based interpreter, making it an excellent solution for internet-based problems. Python, developed in the early 1990s by Guido van Rossum at CWI in Amsterdam and later at CNRI in Virginia, was designed to be easy for beginners yet powerful for advanced

users. Its small, clean syntax and thorough implementation of object-oriented programming, along with support for traditional programming styles, make Python an excellent first programming language that retains the advanced capabilities needed by experienced users [83].

Figure IV- 13 : PYTHON LOGO

4.5.2 Kaggle

Kaggle is an online community platform founded in 2010 by Anthony Goldbloom and Jeremy Howard, later acquired by Google in 2017, aimed at data scientists and machine learning enthusiasts. It enables users to collaborate, publish datasets, use GPU integrated notebooks, and compete in data science challenges. With over 8 million registered users, Kaggle offers powerful cloud-based resources, allowing up to 30 hours of GPU and 20 hours of TPU usage per week. Users can upload, download, and explore datasets, notebooks, and discussion topics, with activities scored and ranked on a live leaderboard. Kaggle serves as a valuable resource for both professionals and learners in their data science journey [84].

Figure IV- 14 : Kaggle LOGO

4.5.3 Google Colaboratory

Google Colab is a cloud-hosted version of Jupyter Notebook from Google Research, allowing Python developers to write and execute code using only a web browser without the need for local installations or hardware upgrades. It provides free access to storage, memory, CPUs, GPUs, and TPUs, making it ideal for machine learning programmers, data scientists, AI researchers, and Python learners. Colab notebooks can integrate executable code, rich text, HTML, LaTeX, images, and data visualizations, facilitating comprehensive presentations of machine learning or data science projects [85].

Figure IV- 15 : Google Colaboratory LOGO

4.5.4 TensorFlow

TensorFlow is an end-to-end opensource platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries, and community resources that lets researchers push the state-of-the-art in ML, and gives developers the ability to easily build and deploy ML-powered applications [86].

Figure IV- 16 : TensorFlow LOGO

4.5.5 Why TensorFlow

TensorFlow offers multiple levels of abstraction to suit different needs, from the high-level Keras API for easy model building and training to eager execution for immediate iteration and debugging. For large-scale ML tasks, the Distribution Strategy API enables distributed training across various hardware configurations. TensorFlow facilitates seamless production deployment on servers, edge devices, and the web, with specialized tools like TFX for production pipelines, TensorFlow Lite for mobile and edge inference, and TensorFlow.js for JavaScript environments. It supports advanced model creation with the Keras Functional API and Model Subclassing API, and features an extensive ecosystem of add-on libraries such as Ragged Tensors, TensorFlow Probability, Tensor2Tensor, and BERT for experimentation and enhanced functionality [87].

4.5.6 Numpy, sklearn, matplotlib, seaborn, skimage

These libraries are essential for scientific computing, data analysis, and visualization within the Python ecosystem, making them invaluable tools for researchers and practitioners across various fields.

4.6 CONCLUSION

In conclusion, this chapter provided a comprehensive overview of the dataset used in the study, detailed the architecture of the VGG16 neural network model, and discussed the training process and results. The significance of individual filters was assessed, leading to insights into their contributions to the model's predictions through the generation of Class Activation Maps. Additionally, the chapter examined a causal interpretation of the Convolutional Neural Network (CNN), including an overview of the causality code. The comparison of the feature maps for pneumonia detection in chest X-rays highlighted that the feature maps before applying the CexCNN method showed high activation in areas not necessarily indicative of pneumonia, leading to potential inaccuracies. After applying the CexCNN method, the feature maps and the bounding boxes appeared more focused, with refined areas of activation that are likely more representative of the true regions affected by pneumonia. This suggests that the CexCNN method enhances the model’s ability to discern relevant features for accurate diagnosis. Finally, the chapter covered the generation of heatmaps and demonstrated the application of the causality method to enhance Class Activation Maps, providing a clearer understanding of the model's decision-making proce

GENERAL CONCLUSION

In this thesis, we have demonstrated the effectiveness of the VGG16 deep learning model for detecting pneumonia from CXR images with high accuracy. By integrating explainability methods such as Causal Explanation CNN (CexCNN), we enhanced the interpretability of the VGG16 model, providing healthcare professionals with clear insights into the decision-making process. These techniques highlight the regions of the images that the model considers important, offering causal explanations and visual evidence for its predictions. This combination of a powerful pre-trained CNN architecture and explainability techniques not only improves diagnostic reliability but also builds trust in AI-based systems.The development of this model was driven by the need for faster diagnosis, reduced workload for radiologists, and improved accessibility to quality healthcare in remote areas. We emphasize that our model is not meant to replace doctors but rather to assist them in diagnosis, serving as valuable tools to enhance their efficiency and accuracy. The results suggest that, with further validation, the VGG16 model, augmented with CexCNN, can be effectively integrated into clinical practice, aiding in the early and accurate detection of pneumonia, and ultimately improving patient outcomes.

BIBLIOGRAPHIE

[1] "Influenza (seasonal)." Accessed: Jun. 30, 2024. [Online]. Available: https://www.who.int/health-topics/influenza-seasonal?gad_source=1&gclid=CjwKCAjwvvmzBhA2EiwAtHVrb25mKsEHn3yxB3L-uL3JXJbcV7Ys-1WRhSNjAcKoi1eX2k2SZQUUOBoCwEQQAvD_BwE#tab=tab_1

[2] "Newborn health." Accessed: Jun. 30, 2024. [Online]. Available: https://www.who.int/europe/news-room/fact-sheets/item/newborn-health

[3] B. Jones, N. Dean, R. Wunderink, M. Sockrider, H. Kulkarni, and C. Dela Cruz, "What is pneumonia?," *American Journal of Respiratory and Critical Care Medicine*, vol. 193, no. 1. American Thoracic Society, pp. P1–P2, Jan. 01, 2016. doi: 10.1164/rccm.1931P1.

[4] W. S. Lim, "Pneumonia—Overview," *Encycl. Respir. Med.*, vol. 4, p. 185, Jan. 2022, doi: 10.1016/B978-0-12-801238-3.11636-8.

[5] "Bronchopneumonia: Symptoms, causes, and treatment." Accessed: Mar. 18, 2024. [Online]. Available: https://www.medicalnewstoday.com/articles/323167

[6] V. A. Zinserling, V. V. Swistunov, A. D. Botvinkin, L. A. Stepanenko, and A. E. Makarova, "Lobar (croupous) pneumonia: old and new data," *Infection*, vol. 50, no. 1, pp. 235–242, Feb. 2022, doi: 10.1007/S15010-021-01689-4.

[7] N. P. Dueck, S. Epstein, T. Franquet, C. C. Moore, and J. Bueno, "Atypical pneumonia: Definition, causes, and imaging features," *Radiographics*, vol. 41, no. 3, pp. 720–741, 2021, doi: 10.1148/rg.2021200131.

[8] S. Shoar and D. M. Musher, "Etiology of community-acquired pneumonia in adults: a systematic review," *Pneumonia*, vol. 12, no. 1, 2020, doi: 10.1186/s41479-020-00074-3.

[9] L. Lux, R. Posey, and L. Daniels, "Pharmacokinetic / Pharmacodynamic Measures for Guiding Antibiotic Treatment for Pneumonia," no. 136, 2014.

[10] L. Papazian, M. Klompas, and C. E. Luyt, "Ventilator-associated pneumonia in adults: a narrative review," *Intensive Care Med.*, vol. 46, no. 5, pp. 888–906, 2020, doi: 10.1007/s00134-020-05980-0.

[11] C. C. Chang, A. C. Cheng, and A. B. Chang, “Over-the-counter (OTC) medications to reduce cough as an adjunct to antibiotics for acute pneumonia in children and adults,” *Cochrane Database of Systematic Reviews*, vol. 2014, no. 3. John Wiley and Sons Ltd, Mar. 10, 2014. doi: 10.1002/14651858.CD006088.pub4.

[12] P. Pahal, V. Rajasurya, and S. Sharma, “Typical Bacterial Pneumonia,” *StatPearls*, Jul. 2023, Accessed: Mar. 18, 2024. [Online]. Available: https://www.ncbi.nlm.nih.gov/books/NBK534295/

[13] N. Miyashita, “Atypical pneumonia: Pathophysiology, diagnosis, and treatment,” *Respir. Investig.*, vol. 60, no. 1, pp. 56–67, Jan. 2022, doi: 10.1016/J.RESINV.2021.09.009.

[14] “Medscape Registration.” Accessed: Jun. 06, 2024. [Online]. Available: https://emedicine.medscape.com/article/300455-overview?form=fpf

[15] “What Causes Pneumonia? | American Lung Association.” Accessed: Mar. 07, 2024. [Online]. Available: https://www.lung.org/lung-health-diseases/lung-disease-lookup/pneumonia/what-causes-pneumonia

[16] “Aspiration Pneumonia - Symptoms and Causes.” Accessed: Jun. 06, 2024. [Online]. Available: https://www.pennmedicine.org/for-patients-and-visitors/patient-information/conditions-treated-a-to-z/aspiration-pneumonia

[17] “Pneumonia Causes and Risk Factors.” Accessed: Mar. 20, 2024. [Online]. Available: https://www.verywellhealth.com/pneumonia-causes-risk-factors-770691

[18] S. A. Harmon *et al.*, “Artificial intelligence for the detection of COVID-19 pneumonia on chest CT using multinational datasets,” *Nat. Commun.*, vol. 11, no. 1, pp. 1–7, 2020, doi: 10.1038/s41467-020-17971-2.

[19] “COVID-19 and Pneumonia: Risks, Symptoms & Treatments | Ada Health.” Accessed: Mar. 19, 2024. [Online]. Available: https://ada.com/covid/covid-19-symptoms-and-pneumonia/

[20] NCGC, “Diagnosis and management of community- and hospital-acquired pneumonia in adults,” no. December, 2010, [Online]. Available: https://www.nice.org.uk/guidance/cg191/evidence/full-guideline-193389085?fbclid=IwAR3Z-7Kp2-cT6RoUCyPe6QTgpHnHTbLgRGs3BLRwOylaI99cTW-6ZW87VmQ

[21] “Pulse Oximetry | Johns Hopkins Medicine.” Accessed: Mar. 18, 2024. [Online]. Available: https://www.hopkinsmedicine.org/health/treatment-tests-and-therapies/pulse-oximetry

[22] “Pneumonia - Diagnosis and treatment - Mayo Clinic.” Accessed: Mar. 18, 2024. [Online]. Available: https://www.mayoclinic.org/diseases-conditions/pneumonia/diagnosis-treatment/drc-20354210

[23] O. Access, “[56] Research_in_Medical_Imaging_Using_Image_Processing 2019.pdf”.

[24] “Computed Tomography (CT) Scan | Johns Hopkins Medicine.” Accessed: Mar. 18, 2024. [Online]. Available: https://www.hopkinsmedicine.org/health/treatment-tests-and-therapies/computed-tomography-ct-scan

[25] “Information on Gadolinium-Based Contrast Agents | FDA.” Accessed: Mar. 18, 2024. [Online]. Available: https://www.fda.gov/drugs/postmarket-drug-safety-information-patients-and-providers/information-gadolinium-based-contrast-agents

[26] S. Anis *et al.*, “An overview of deep learning approaches in chest radiograph,” *IEEE Access*, vol. 8, pp. 182347–182354, 2020, doi: 10.1109/ACCESS.2020.3028390.

[27] B. U. M. B N, R. I. S, N. Jain, K. C S, and N. K R, “Detection of Pneumonia Using Chest X-Ray Images and Image Processing Algorithms - A Comparative Study,” *J. Image Process. Artif. Intell.*, vol. 7, no. 3, 2021, doi: 10.46610/joipai.2021.v07i03.005.

[28] “Does Pneumonia Always Show on Chest X-rays? – Radiology In Plain English.” Accessed: Jun. 06, 2024. [Online]. Available: https://radiologyinplainenglish.com/does-pneumonia-always-show-on-chest-x-rays/

[29] A. K. Sangaiah, “Deep Learning and Parallel Computing Environment for Bioengineering Systems,” *Deep Learn. Parallel Comput. Environ. Bioeng. Syst.*, pp. 1–260, Jan. 2019, doi: 10.1016/C2018-0-00906-8.

[30] “What is Computer Vision? | IBM.” Accessed: May 11, 2024. [Online]. Available: https://www.ibm.com/topics/computer-vision

[31] “The 100 Most Popular Computer Vision Applications in 2024 - viso.ai.” Accessed: May 02, 2024. [Online]. Available: https://viso.ai/applications/computer-vision-applications/

[32] “What are Convolutional Neural Networks? | IBM.” Accessed: May 28, 2024.

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

[33] S. Albawi, T. A. Mohammed, and S. Al-Zawi, “Understanding of a convolutional neural network,” *Proc. 2017 Int. Conf. Eng. Technol. ICET 2017*, vol. 2018-Janua, pp. 1–6, Jul. 2017, doi: 10.1109/ICENGTECHNOL.2017.8308186.

[34] “Convolutional Neural Networks: A Comprehensive Guide | by Jorgecardete | The Deep Hub | Medium.” Accessed: Jun. 06, 2024. [Online]. Available: https://medium.com/thedeephub/convolutional-neural-networks-a-comprehensive-guide-5cc0b5eae175

[35] X. Feng, Y. Jiang, X. Yang, M. Du, and X. Li, “Computer vision algorithms and hardware implementations: A survey,” *Integration*, vol. 69, pp. 309–320, Nov. 2019, doi: 10.1016/J.VLSI.2019.07.005.

[36] J. Deng, W. Dong, R. Socher, L.-J. Li, Kai Li, and Li Fei-Fei, “ImageNet: A large-scale hierarchical image database,” pp. 248–255, Mar. 2010, doi: 10.1109/CVPR.2009.5206848.

[37] S. Dasiopoulou, V. Mezaris, I. Kompatsiaris, V. K. Papastathis, and M. G. Strintzis, “Knowledge-assisted semantic video object detection,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 10, pp. 1210–1224, Oct. 2005, doi: 10.1109/TCSVT.2005.854238.

[38] D. Guo, Y. Pei, K. Zheng, H. Yu, Y. Lu, and S. Wang, “Degraded Image Semantic Segmentation with Dense-Gram Networks,” *IEEE Trans. Image Process.*, vol. 29, pp. 782–795, 2019, doi: 10.1109/TIP.2019.2936111.

[39] J. Yi, P. Wu, M. Jiang, Q. Huang, D. J. Hoeppner, and D. N. Metaxas, “Attentive neural cell instance segmentation,” *Med. Image Anal.*, vol. 55, pp. 228–240, Jul. 2019, doi: 10.1016/J.MEDIA.2019.05.004.

[40] S. Wang, G. Wen, and H. Cai, “Feature extraction and face recognition algorithm,” *ICNC-FSKD 2017 - 13th Int. Conf. Nat. Comput. Fuzzy Syst. Knowl. Discov.*, pp. 1910–1914, Jun. 2018, doi: 10.1109/FSKD.2017.8393059.

[41] “A Review of Different Interpretation Methods (Part 1: Saliency Map, CAM, Grad-CAM) | by Mohammadreza (Reza) Salehi | Medium.” Accessed: Jun. 06, 2024. [Online]. Available: https://mrsalehi.medium.com/a-review-of-different-interpretation-methods-in-deep-learning-part-1-saliency-map-cam-grad-cam-3a34476bc24d

[42] S. J. Pan and Q. Yang, “A Survey on Transfer Learning,” 2009, doi: 10.1109/TKDE.2009.191.

[43] “Computer Vision: Do You Know VGG 16? | by Chinmay Bhalerao | Towards AI.” Accessed: Jun. 06, 2024. [Online]. Available: https://pub.towardsai.net/do-you-know-vgg-16-f7b8f9a5e550

[44] S. V. Militante, N. V. Dionisio, and B. G. Sibbaluca, “Pneumonia and COVID-19 Detection using Convolutional Neural Networks,” *Proceeding - 2020 3rd Int. Conf. Vocat. Educ. Electr. Eng. Strength. Framew. Soc. 5.0 through Innov. Educ. Electr. Eng. Informatics Eng. ICVEE 2020*, 2020, doi: 10.1109/ICVEE50212.2020.9243290.

[45] “Everything you need to know about VGG16 | by Great Learning | Medium.” Accessed: Jun. 06, 2024. [Online]. Available: https://medium.com/@mygreatlearning/everything-you-need-to-know-about-vgg16-7315defb5918

[46] G. P. Reddy and Y. V. P. Kumar, “Explainable AI (XAI): Explained,” *2023 IEEE Open Conf. Electr. Electron. Inf. Sci. eStream 2023 - Proc.*, 2023, doi: 10.1109/ESTREAM59056.2023.10134984.

[47] S. Mishra, B. L. Sturm, and S. Dixon, “LOCAL INTERPRETABLE MODEL-AGNOSTIC EXPLANATIONS FOR MUSIC CONTENT ANALYSIS”, Accessed: May 10, 2024. [Online]. Available: https://code.soundsoftware.ac.uk/projects/SoundLIME

[48] “Welcome to the SHAP documentation — SHAP latest documentation.” Accessed: May 09, 2024. [Online]. Available: https://shap.readthedocs.io/en/latest/

[49] A. Chaddad, J. Peng, J. Xu, and A. Bouridane, “Survey of Explainable AI Techniques in Healthcare,” *Sensors (Basel).*, vol. 23, no. 2, Jan. 2023, doi: 10.3390/S23020634.

[50] E. Tjoa and C. Guan, “A Survey on Explainable Artificial Intelligence (XAI): Toward Medical XAI,” *IEEE Trans. Neural Networks Learn. Syst.*, vol. 32, no. 11, pp. 4793–4813, Nov. 2021, doi: 10.1109/TNNLS.2020.3027314.

[51] A. Chaddad, J. Peng, J. Xu, and A. Bouridane, “Survey of Explainable AI Techniques in Healthcare,” *Sensors (Basel).*, vol. 23, no. 2, Jan. 2023, doi: 10.3390/S23020634.

[52] J. Irvin *et al.*, “CheXpert: A Large Chest Radiograph Dataset with Uncertainty Labels

and Expert Comparison," *33rd AAAI Conf. Artif. Intell. AAAI 2019, 31st Innov. Appl. Artif. Intell. Conf. IAAI 2019 9th AAAI Symp. Educ. Adv. Artif. Intell. EAAI 2019*, pp. 590–597, Jan. 2019, doi: 10.1609/aaai.v33i01.3301590.

[53] Z. Tang *et al.*, "Interpretable classification of Alzheimer's disease pathologies with a convolutional neural network pipeline," *Nat. Commun.*, vol. 10, no. 1, Dec. 2019, doi: 10.1038/S41467-019-10212-1.

[54] K. Young, G. Booth, B. Simpson, R. Dutton, and S. Shrapnel, "Deep neural network or dermatologist?," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11797 LNCS, pp. 48–55, Aug. 2019, doi: 10.1007/978-3-030-33850-3_6.

[55] G. Zhao, B. Zhou, K. Wang, R. Jiang, and M. Xu, "Respond-CAM: Analyzing deep models for 3D imaging data by visualizations," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11070 LNCS, pp. 485–492, 2018, doi: 10.1007/978-3-030-00928-1_55.

[56] M. Izadyyazdanabadi *et al.*, "Weakly-Supervised Learning-Based Feature Localization in Confocal Laser Endomicroscopy Glioma Images".

[57] Y. Qin *et al.*, "Autofocus Layer for Semantic Segmentation," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11072 LNCS, pp. 603–611, May 2018, doi: 10.1007/978-3-030-00931-1_69.

[58] C. Ulas, G. Tetteh, S. Kaczmarz, C. Preibisch, and B. H. Menze, "DeepASL: Kinetic Model Incorporated Loss for Denoising Arterial Spin Labeled MRI via Deep Residual Learning".

[59] A. E. U. Cerna *et al.*, "Interpretable Neural Networks for Predicting Mortality Risk using Multi-modal Electronic Health Records," Jan. 2019, Accessed: May 05, 2024. [Online]. Available: https://arxiv.org/abs/1901.08125v1

[60] Y. Yan, E. Solarz, J. Zhu, C. Sripada, M. Duda, and D. Koutra, "Groupinn: Grouping-based interpretable neural network for classification of limited, noisy brain data," *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, pp. 772–782, Jul. 2019, doi: 10.1145/3292500.3330921.

[61] O. Kocadagli and R. Langari, "Classification of EEG signals for epileptic seizures using hybrid artificial neural networks based wavelet transforms and fuzzy relations,"

Expert Syst. Appl., vol. 88, pp. 419–434, Dec. 2017, doi: 10.1016/J.ESWA.2017.07.020.

[62] A. Hazarika, M. Barthakur, L. Dutta, and M. Bhuyan, "F-SVD based algorithm for variability and stability measurement of bio-signals, feature extraction and fusion for pattern recognition," *Biomed. Signal Process. Control*, vol. 47, pp. 26–40, Jan. 2019, doi: 10.1016/J.BSPC.2018.08.010.

[63] C. Biffi *et al.*, "Learning Interpretable Anatomical Features Through Deep Generative Models: Application to Cardiac Remodeling," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11071 LNCS, pp. 464–471, Jul. 2018, doi: 10.1007/978-3-030-00934-2_52.

[64] N. C. F. Codella, C.-C. Lin, A. Halpern, M. Hind, R. Feris, and J. R. Smith, "Collaborative Human-AI (CHAI): Evidence-Based Interpretable Melanoma Classification in Dermoscopic Images", Accessed: May 05, 2024. [Online]. Available: http://isic-archive.com

[65] H. Yeche, J. Harrison, and T. Berthier, "UBS: A Dimension-Agnostic Metric for Concept Vector Interpretability Applied to Radiomics," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11797 LNCS, pp. 12–20, 2019, doi: 10.1007/978-3-030-33850-3_2.

[66] P. Zhu and M. Ogino, "Guideline-Based Additive Explanation for Computer-Aided Diagnosis of Lung Nodules," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11797 LNCS, pp. 39–47, 2019, doi: 10.1007/978-3-030-33850-3_5.

[67] M. Pisov *et al.*, "Incorporating Task-Specific Structural Knowledge into CNNs for Brain Midline Shift Detection," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11797 LNCS, pp. 30–38, 2019, doi: 10.1007/978-3-030-33850-3_4.

[68] N. Prentzas, A. Nicolaides, E. Kyriacou, A. Kakas, and C. Pattichis, "Integrating machine learning with symbolic reasoning to build an explainable ai model for stroke prediction," *Proc. - 2019 IEEE 19th Int. Conf. Bioinforma. Bioeng. BIBE 2019*, pp. 817–821, Oct. 2019, doi: 10.1109/BIBE.2019.00152.

[69] R. Caruana, Y. Lou, J. Gehrke, P. Koch, M. Sturm, and N. Elhadad, "Intelligible

Models for HealthCare: Predicting Pneumonia Risk and Hospital 30-day Readmission," *Knowl. Discov. Data Min.*, vol. 2015-August, pp. 1721–1730, Aug. 2015, doi: 10.1145/2783258.2788613.

[70] X. Li, N. C. Dvornek, J. Zhuang, P. Ventola, and J. S. Duncan, "Brain Biomarker Interpretation in ASD Using Deep Learning and fMRI," *Med. Image Comput. Comput. Assist. Interv.*, vol. 11072, pp. 206–214, 2018, doi: 10.1007/978-3-030-00931-1_24.

[71] B. Letham, C. Rudin, T. H. McCormick, and D. Madigan, "Interpretable classifiers using rules and Bayesian analysis: Building a better stroke prediction model," *Ann. Appl. Stat.*, vol. 9, no. 3, pp. 1350–1371, Nov. 2015, doi: 10.1214/15-AOAS848.

[72] E. Varol, A. Sotiras, K. Zeng, and C. Davatzikos, "Generative discriminative models for multivariate inference and statistical mapping in medical imaging," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11072 LNCS, pp. 540–548, Jul. 2018, doi: 10.1007/978-3-030-00931-1_62.

[73] K. Suzuki *et al.*, Eds., "Interpretability of Machine Intelligence in Medical Image Computing and Multimodal Learning for Clinical Decision Support," vol. 11797, 2019, doi: 10.1007/978-3-030-33850-3.

[74] S. Haufe *et al.*, "On the interpretation of weight vectors of linear models in multivariate neuroimaging," *Neuroimage*, vol. 87, pp. 96–110, Feb. 2014, doi: 10.1016/J.NEUROIMAGE.2013.10.067.

[75] "Basics of Bounding Boxes. What is a bounding box? | by Vineeth S Subramanyam | Analytics Vidhya | Medium." Accessed: May 08, 2024. [Online]. Available: https://medium.com/analytics-vidhya/basics-of-bounding-boxes-94e583b5e16c

[76] "IOU (Intersection over Union). What is IOU? | by Vineeth S Subramanyam | Analytics Vidhya | Medium." Accessed: May 15, 2024. [Online]. Available: https://medium.com/analytics-vidhya/iou-intersection-over-union-705a39e7acef

[77] "Intersection over Union (IoU): Definition, Calculation, Code." Accessed: May 14, 2024. [Online]. Available: https://www.v7labs.com/blog/intersection-over-union-guide#h1

[78] "RSNA Pneumonia Detection Challenge | Kaggle." Accessed: Jun. 30, 2024. [Online]. Available: https://www.kaggle.com/c/rsna-pneumonia-detection-challenge/

[79] “Chest X-Ray Images (Pneumonia).” Accessed: Mar. 09, 2024. [Online]. Available: https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia

[80] “Dataset Search.” Accessed: Mar. 21, 2024. [Online]. Available: https://datasetsearch.research.google.com/search?query=x-ray&docid=L2cvMTFqc2RkNnozdA%3D%3D

[81] H. Debbi, “Causal Explanation of Convolutional Neural Networks,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12976 LNAI, pp. 633–649, 2021, doi: 10.1007/978-3-030-86520-7_39.

[82] “HichemDebbi/CexCNN: CexCNN is a causal explanation technique for Convolutional Neural Networks (CNNs). CexCNN helps to identify the most important features in the input image tha contributed to a CNN’s decision. Besides, it shows good results in Weakly Su.” [Online]. Available: https://github.com/HichemDebbi/CexCNN/

[83] P. Spector, “Introduction to Python Programming Course Notes,” 2005.

[84] “What is Kaggle? | DataCamp.” Accessed: Jun. 01, 2024. [Online]. Available: https://www.datacamp.com/blog/what-is-kaggle

[85] “Google Colab: Everything You Need to Know - Geekflare.” Accessed: Jun. 01, 2024. [Online]. Available: https://geekflare.com/google-colab/

[86] “TensorFlow | Google Open Source Projects.” Accessed: Jun. 01, 2024. [Online]. Available: https://opensource.google/projects/tensorflow

[87] “Why TensorFlow.” Accessed: Jun. 01, 2024. [Online]. Available: https://www.tensorflow.org/about

ABSTRACT

Pneumonia detection from chest X-ray images is achieved using the pre-trained VGG16 deep learning model, known for its ability to extract relevant features from images. The process involves feeding the X-ray images into the VGG16 model, which then analyzes and classifies them to detect signs of pneumonia. To build trust in the model's decisions, explainability techniques such as Causal Explanation CNN (CexCNN) and CAM are employed. These techniques provide visual explanations by highlighting the regions of the X-ray images that the model focuses on when making its predictions. CexCNN offers casual robust explanations, indicating which features are most influential in the decision-making process, thus leading to gain confidence in the CNN model. Which is very crucial especially in the medical field.

Keywords: Deep learning, CNN, CexCNN, VGG16, ImageNet, Computer vision, heatmaps, Boundingbox, X-Ray.

ملخص

اكتشاف الالتهاب الرئوي يتم من خلال صور الأشعة السينية للصدر باستخدام نموذج التعلم العميق VGG16 المُدرب مسبقًا، والمعروف بقدرته على استخراج الميزات ذات الصلة من الصور. تتضمن العملية إدخال صور الأشعة السينية في نموذجVGG16 ، الذي يقوم بعد ذلك بتحليلها وتصنيفها للكشف عن علامات الالتهاب الرئوي. لبناء الثقة في قرارات النموذج، يتم استخدام تقنيات التفسير مثل التفسير السببي CNN (CexCNN) و CAM. توفر هذه التقنيات تفسيرات مرئية من خلال تسليط الضوء على مناطق صور الأشعة السينية التي يركز عليها النموذج عند وضع تنبؤاته. تقدم CexCNN تفسيرات قوية غير رسمية، تشير إلى الميزات الأكثر تأثيرًا في عملية صنع القرار، مما يؤدي إلى اكتساب الثقة في نموذج CNN.وهو أمر بالغ الأهمية خاصة في المجال الطبي.

Résumé

La pneumonie est détectée à partir d'images radiographiques pulmonaires à l'aide du modèle d'apprentissage profond pré-entraîné VGG16, connu pour sa capacité à extraire des caractéristiques pertinentes des images. Le processus consiste à saisir des images radiographiques dans le modèle VGG16, qui les analyse et les classe ensuite pour détecter les signes de pneumonie. Pour renforcer la confiance dans les décisions du modèle, des techniques d'interprétation telles que l'explication causale CNN (CexCNN) et CAM sont utilisées. Ces techniques fournissent des interprétations visuelles en mettant en évidence les zones des images radiographiques sur lesquelles le modèle se concentre lors de ses prédictions. CexCNN fournit des explications informelles robustes, indiquant quelles caractéristiques sont les plus influentes

dans le processus de prise de décision, ce qui permet de gagner en confiance dans le modèle CNN. C’est extrêmement important, surtout dans le domaine médical.