People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research

IBN KHALDOUN UNIVERSITY OF TIARET

Dissertation

Presented to:

FACULTY OF MATHEMATICS AND COMPUTER SCIENCE
DEPARTEMENT OF COMPUTER SCIENCE

in order to obtain the degree of:

MASTER

Specialty: Software Engineering

Presented by:

- BENALI Bochra Yamina
- BENAOUM Karima Haifaa

On the theme:

Proposal and Evaluation of a Deep Learning Model for image segmentation

---

Defended publicly on 10 / 06 /2025 in Tiaret in front the jury composed of:

Mr DAOUD Mohamed Amine	M.C.B	Tiaret University	Chairman
Mr CHENINE Abdelkader	M.A.A	Tiaret University	Supervisor
Mr ZIOUAL TAHAR	M.A.A	Tiaret University	Examiner

2024-2025

Acknowledgments

First and foremost, we would like to thank Allah for granting us the will, health, patience, and strength to complete this humble work.

We sincerely thank our supervisor, **Dr. Chenin AEK**, for his valuable guidance and support. We also warmly thank the members of the jury, **Dr. DAOUD Mohamed Amine** and **Dr. Zioual Tahar**, for honoring us with their presence and for their constructive remarks and evaluation of our work.

Our gratitude extends to all the faculty members and staff of the department for the knowledge, support, and opportunities they have provided throughout our academic journey.

We are deeply grateful to our families and friends for their collaboration, moral support, and encouragement during challenging moments.

Finally, we express our heartfelt appreciation to our families, whose love, patience, and constant support have been our greatest source of strength throughout our studies.

.

Dedication

This thesis is dedicated to the soul of my beloved ***grandmother****, who lost her battle with breast cancer. Her memory lives on through this work. She dreamed of seeing me graduate this year, but her light faded too soon.*

To the most important person in my life the pillars of my strength and inspiration*,* ***my beloved parents (Lahcen & Linda)****. Your unwavering support and encouragement throughout all these years have been the foundation of my journey. Even in my moments of weakness and despair, you lifted me up every time I fell and gave me everything I needed to succeed.*
Your sacrifices, patience, and endless love made this achievement possible.

To my brothers, ***Salah****,* ***Abdel Illah*** *thank you for your constant support, patience, and encouragement.*

To my wonderful ***friends,*** *thank you for bringing joy, happiness, and unforgettable memories to this journey, your presence made even the hardest days lighter.*

Finally, I thank everyone who supported me throughout this path.

Benaoum Karima Haifaa

Dedication

*To my beloved parents(**Bachir & Khalida**), my everlasting support and strength;*
to my dear mother, the symbol of patience and unconditional love, and to my cherished father, whose kindness lights my path;
*to my siblings **Djinan**, **AEK**, and **Chakib**, who stood by me every step of the way this success is for you.*

My heartfelt thanks also go to my friends for all the beautiful memories we shared.
I lovingly remember my grandfather, who left us this year and always wished to see me succeed
may this work be counted among his good deeds, and I ask every reader to say a prayer for his mercy.

Finally, I thank myself for the resilience and determination that brought me to this moment

Benali Bochra Yamina

Abstract

Breast cancer represents a major global health challenge, particularly due to its impact on women's health. Improving diagnostic capabilities is essential to ensure timely intervention and reduce complications. Yet, the traditional manual analysis of mammographic images often proves to be time-intensive and susceptible to variability among medical professionals. This thesis explores a deep learning-based approach for the semantic segmentation of mammographic images, aiming to enhance the detection and localization of suspicious breast lesions. In addition to developing a robust segmentation technique, this work includes the integration of the system into a mobile user interface, enabling practical testing and accessibility. The proposed approach illustrates the growing role of artificial intelligence in advancing medical imaging and supporting early breast cancer diagnosis.

Keywords

Breast Cancer, Medical Imaging, Mammography, Deep Learning, Semantic Segmentation, Artificial Intelligence, Mobile Application, Computer-Aided Diagnosis, Image Processing, Neural Networks

الملخص

يمثل سرطان الثدي تحديًا صحيًا عالميًا، نظرًا لتأثيره الكبير على صحة النساء. يُعدّ تحسين وسائل التشخيص أمرًا ضروريًا لضمان التدخل في الوقت المناسب وتقليل التعقيدات المحتملة. إلا أن تحليل صور تصوير الثدي بالطريقة التقليدية اليدوية يستغرق وقتًا طويلاً وقد يعاني من التباين بين الأطباء. يهدف هذا البحث إلى اعتماد منهج قائم على التعلم العميق لتجزئة صور الثدي بشكل دقيق، مما يعزز من القدرة على كشف وتحديد مواضع الآفات المشتبه بها. كما يتضمن العمل تطوير واجهة تطبيق مخصص لاختبار النظام في بيئة عملية وسهلة الاستخدام، مما يُبرز إمكانيات الذكاء الاصطناعي في دعم أنظمة التصوير الطبي وتشخيص سرطان الثدي في مراحله المبكرة.

الكلمات المفتاحية

سرطان الثدي، التصوير الطبي، تصوير الثدي الشعاعي، التعلم العميق، التجزئة الدلالية، الذكاء الاصطناعي، تطبيق الهاتف المحمول، التشخيص بمساعدة الحاسوب، معالجة الصور، الشبكات العصبية.

Résumé

Le cancer du sein constitue un défi majeur de santé publique en raison de son incidence élevée chez les femmes. L’amélioration des outils diagnostiques s’avère essentielle pour permettre une prise en charge rapide et réduire les complications. Cependant, l’analyse manuelle des mammographies reste une tâche longue et sujette à des divergences d’interprétation. Ce mémoire propose une approche fondée sur l’apprentissage profond pour la segmentation sémantique des images mammaires, pour objectif de faciliter l’identification précise des lésions suspectes. En complément, une interface mobile a été développée afin de permettre l’évaluation pratique du système dans un environnement accessible. Ce travail souligne le potentiel de l’intelligence artificielle pour renforcer les outils d’aide au diagnostic et contribuer à la détection précoce du cancer du sein.

Mots-clés

Cancer du sein, Imagerie médicale, Mammographie, Apprentissage profond, Segmentation sémantique, Intelligence artificielle, Application mobile, Diagnostic assisté par ordinateur, Traitement d’image, Réseaux de neurones.

Table of Contents

List of Figures

List of Tables

Glossary of Abbreviations and Acronyms

AI Artificial Intelligence

ML Machine Learning

DL Deep Learning

MLP Multi-layer Perceptrons

GD Gradient Descent

CNN Convolutional Neural Network

GPU Graphical Processing Unite

ReLU Rectified Linear Unit

Adam Adaptive Moment Estimation

IOU Intersection Over Union

DC Dice Coefficient

FP False positive

FN False Negative

TN True Negative

TP True Positive

General Introduction

Context and Background

Breast cancer remains one of the most prevalent and life-threatening diseases affecting women worldwide. Early detection plays a vital role in increasing survival rates and improving treatment outcomes. Mammography, as the primary imaging technique for breast cancer screening, is an essential diagnostic tool, providing detailed X-ray images that enable radiologists to identify potential abnormalities such as masses, micro calcifications, and architectural distortions. However, the manual interpretation of mammographic images is time-consuming requiring extensive expertise and is subject to inter-observer variability, leading to challenges in consistent and accurate diagnosis.

The advent of artificial intelligence and deep learning technologies has opened new avenues for improving medical image analysis, particularly in the domain of mammogram interpretation. Semantic segmentation, which involves pixel-level classification of medical images, plays a pivotal role in computer-aided diagnosis (CAD) systems by enabling precise localization and isolate suspicious regions such as masses or tumors, enabling more accurate diagnosis and treatment planning.

This thesis focuses on the integration of AI techniques for preprocessing and semantic segmentation of medical images, specifically mammography images. The aim is to develop an efficient and robust model capable of accurately segmenting breast masses while minimizing computational cost. Accurate segmentation of mammographic structures can greatly enhance diagnostic accuracy, reduce interpretation time, and offer valuable quantitative measurements to support clinical decision-making.

To validate the practical usability of the proposed model, a mobile application was developed using Flutter, enabling users to test the segmentation system directly on mammographic images.

Problematic

The early detection of breast cancer significantly increases the chances of successful treatment and patient survival. However, manual interpretation of mammographic images is often time-consuming, subject to inter-observer variability, and prone to diagnostic errors. Traditional image processing and segmentation techniques lack the robustness and precision needed to accurately identify abnormal regions in complex breast tissue structures. These limitations highlight the need for more reliable, consistent, and efficient methods for analyzing and segmenting mammographic images.

Proposed approach

To address these challenges, this thesis proposes a hybrid deep learning architecture for mammogram image segmentation, combining MobileNetV2 as a lightweight encoder with

ResUNet as the decoder. This specific combination leverages MobileNetV2's computational efficiency alongside ResUNet's powerful segmentation capabilities through residual connections. Furthermore, to facilitate practical testing and usability, we developed a mobile application using Flutter, which communicates with the segmentation model via a Flask API. This integration enables real-time evaluation of the model's performance on mammographic images directly from mobile devices, bridging the gap between research and clinical application.

Plan

This thesis consists of four chapters in addition to general introduction and general conclusion:

- **Chapter I: Overview of Breast Cancer**

  This chapter provides a comprehensive introduction to breast cancer. It covers the anatomy and physiology of the breast, main risk factors contributing to breast cancer development, and the different types of breast cancer. Additionally, it reviews current diagnostic techniques used in clinical practice, followed by an explanation of the various stages of breast cancer progression.

- **Chapter II: Image Processing**

  This chapter provides an overview of image processing, including an explanation of what constitutes an image and the different levels of image processing. This chapter then discusses the main stages involved in image processing workflows. Furthermore, it reviews traditional image segmentation approaches, highlighting their strengths and limitations in medical image analysis.

- **Chapter III: Deep learning & Segmentation**

  An introductory exploration of the fundamentals of artificial intelligence (AI), machine learning (ML), and deep learning(DL), emphasizing their roles in medical image analysis. It describes the architecture and mechanisms of neural networks and explores the main types of segmentation techniques used in deep learning. The chapter also discusses existing segmentation model architectures and introduces key concepts such as transfer learning and the use of pretrained models to improve performance and reduce training time.

- **Chapter IV: Experiments and Realization**

  In this chapter, we delve deeply into the practical implementation of the proposed hybrid deep learning architecture, MobileNetV2-ResUNet, specifically designed for mammogram image segmentation. The chapter details the dataset used, preprocessing steps, and the architectural design of the model, including training strategy and evaluation results. Finally, the chapter presents the deployment of the trained model through a mobile to enable real-time testing on mammographic images.

Chapter I : Overview of Breast Cancer

I.1 Introduction

Breast cancer is a major public health concern and the leading cause of cancer-related deaths among women. While it is less common in younger women, approximately 25% of cases occur before the age of 50. Mass screening programs primarily target women over 50, while younger women undergo annual gynecological check-ups, including clinical breast examinations. Additionally, women aged 40 to 50 may benefit from individualized mammography screening.

As the most prevalent cancer in women, breast cancer represents the foremost oncological pathology and a leading cause of premature mortality. In this chapter, we will explore this widespread global disease in detail, covering its definition, historical background, classification, risk factors, screening methods, diagnostic approaches, treatment options, and follow-up care.

I.2 Understanding the breast

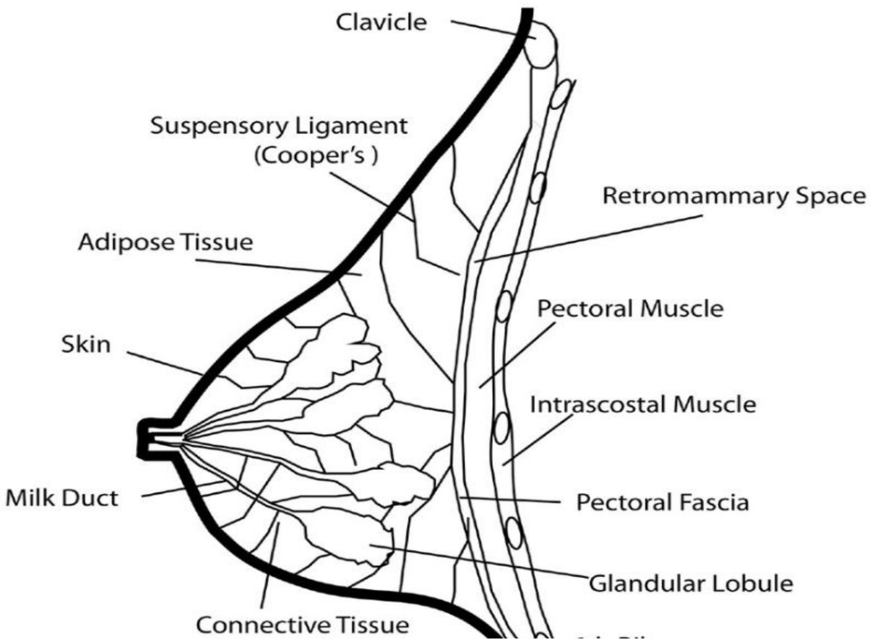


Figure I. 1 : Anatomical Cross-Section of the Human

This sagittal (side) view of the female breast shows the main anatomical structures that are important in understanding breast cancer. **Milk ducts** carry milk from the **Glandular lobules** to the nipple and are a common place where breast cancer begins (ductal carcinoma). The **glandular lobules**, which produce milk, can also be affected (lobular carcinoma). Fat tissue gives shape to the breast. **Cooper's** ligaments support the breast and can become involved in cancer, causing dimpling of the skin. Behind the breast, **muscles** (**pectoral** and **intercostal**), the **Retromammary space**, and ribs are shown. Understanding this anatomy helps doctors in imaging, surgery, and treatment planning for **breast cancer** [1]. As shown in (**Figure I.1)**

I.3 Definition

Breast cancer is a disease characterized by the abnormal and uncontrolled growth of certain breast cells. These cells multiply excessively and usually form a mass known as a tumor. There are various types of breast cancer, each with different patterns of progression. Some types are considered aggressive and develop rapidly, while others progress more slowly.

Initially, cancerous cells may remain confined to the breast. However, they can also spread to other parts of the body through the bloodstream or lymphatic system—a process known as metastasis, which marks a more serious stage of the disease. The most common sites of metastasis are the lungs, liver, bones, and brain.

Breast cancer often develops over several months or even years before becoming detectable. It is the most frequently diagnosed cancer among women, accounting for more than one-third of all new female cancer cases. **[2]**

There are two main categories of tumors :

Benign tumors	Malignant tumors (cancers)
Well-defined	Poorly defined
Encapsulated	Non-encapsulated
Histologically similar to the tissue of origin	Partially or not similar to the tissue of origin (dedifferentiation, aberrant differentiation)
Regular (uniform) cells	Irregular cells (cancer cells)
Slow growth	Rapid growth
Displacement of adjacent tissues without their destruction	Invasion of surrounding tissues
No local recurrence after complete surgical excision	Possible recurrence after presumed complete excision
No metastasis	metastasis

Table I. 1 : Criteria for Differentiating Between Benign and Malignant Tumors **[3]**

I.4 The main risk factors for breast cancer

Multiple risk factors have been associated with the development of breast cancer. These factors can be classified as **non-modifiable** (e.g., age, genetics) and **modifiable** (e.g., lifestyle choices).

a) **Age**: The risk of developing breast cancer increases with age, particularly after the age of 50

b) **Genetic mutations**: Inherited mutations in genes such as **BRCA1** and **BRCA2** significantly increase the lifetime risk of breast cancer. Women with a BRCA1 mutation have a 55–65% risk, while BRCA2 mutations confer about a 45% risk by age 70. **[4]**

c) **Family history**: Having a first-degree relative (mother, sister, daughter) with breast cancer roughly doubles the risk.

d) **Radiation exposure**: Women who received radiation therapy to the chest, especially during adolescence, are at increased risk

e) **Hormonal factors**:

   a. Early menstruation (before age 12).
   b. Late menopause (after age 55).
   c. Hormone replacement therapy (HRT) with estrogen and progesterone.
   d. Never having been pregnant or having a first pregnancy after age 30.

f) **Lifestyle factors**: Obesity and physical inactivity , Alcohol consumption (higher intake increases risk) and Smoking. **[5]**

I.5 Types of breast cancer

There are numerous types of breast cancer and numerous ways to describe them. The specific cells in the breast that become cancer determine the type of breast cancer. **[6]** The most common types of breast cancer are:

I.5.1 Non-Invasive (In Situ) Breast Cancer

a) Lobular Carcinoma In Situ (LCIS)

Lobular carcinoma in situ (LCIS) is an uncommon condition in which abnormal cells form in the milk glands (lobules) in the breast. Lobular carcinoma in situ (LCIS) isn't cancer. But being diagnosed with LCIS indicates that you have an increased risk of developing breast cancer. **[7]**

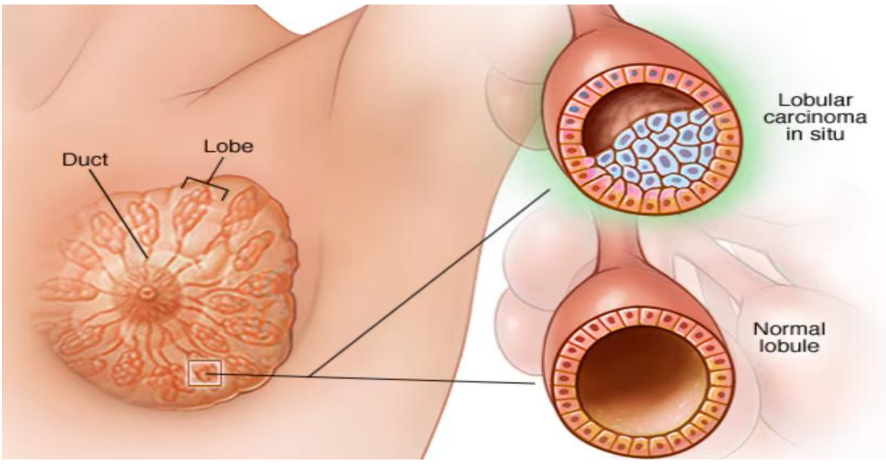


Figure I. 2 : Lobular Carcinoma In Situ

b) Ductal carcinoma in situ (DCIS)

DCIS, also called intraductal carcinoma or stage 0 breast cancer, is a non-invasive form: the duct-lining cells have become cancerous but remain inside the ducts and haven't spread into the surrounding breast tissue. **[6]**

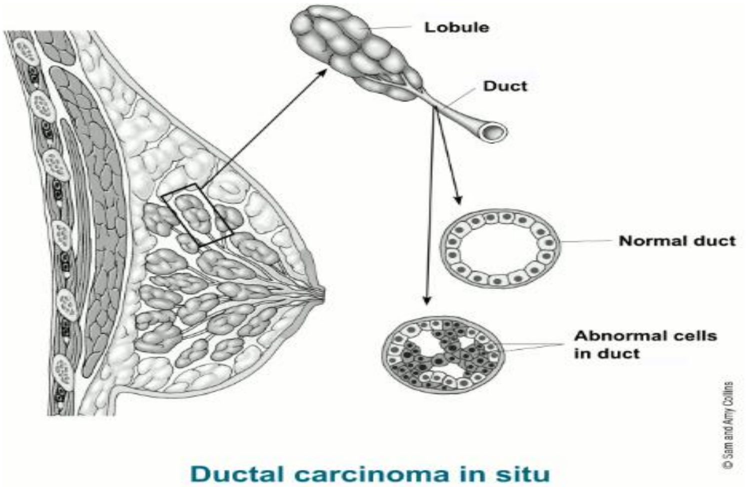


Figure I. 3 : Ductal carcinoma in situ

I.5.2 Invasive breast cancers

a) Invasive Ductal Carcinoma (IDC)

Invasive ductal carcinoma (IDC) is the most common form of invasive breast cancer, making up about 80 % of cases. It begins in a milk duct, then breaks through the duct wall to grow into nearby breast tissue, where it can enter the lymphatic system or bloodstream and spread to other parts of the body.**[6]**

b) Invasive Lobular Carcinoma (ILC)

Invasive lobular carcinoma (ILC) accounts for about 10% of invasive breast cancers. It starts in the milk-producing lobules and can spread to other parts of the body. ILC may be harder to detect on physical exams and imaging, such as mammograms, than invasive ductal carcinoma. In about 20% of cases, it affects both breasts at the time of diagnosis. **[6]**

I.6 Diagnostic Techniques for Breast Cancer

I.6.1 Mammography

is a specialized imaging technique that uses low-dose X-rays to capture detailed internal views of the breast, enabling the detection of small tumors and micro calcifications that may indicate early-stage breast cancer. It remains the most reliable method for breast cancer screening, particularly effective in identifying abnormalities before physical symptoms appear. Recommended especially for women over 45—who face higher risks due to hormonal changes, menopause, and weakened immunity mammography significantly improves early detection and treatment outcomes. Its benefits include clearer imaging of dense breast tissue, better localization of abnormalities, fewer unnecessary biopsies, and an increased ability to detect multiple tumors, ultimately contributing to reduced breast cancer mortality through earlier intervention **. [8]**

I.6.2 Breast Ultrasound

Breast ultrasound is a non-invasive imaging technique primarily used as a complementary diagnostic tool rather than a routine screening method for breast cancer. It is particularly valuable for examining breast changes such as palpable lumps that may not be visible on mammograms, especially in women with dense breast tissue. Ultrasound helps differentiate between fluid-filled cysts, which are usually benign, and solid masses that may require further investigation. Additionally, it plays a crucial role in guiding biopsy procedures by accurately targeting suspicious areas or swollen lymph nodes. Performed with a handheld transducer that emits sound waves and captures echoes to generate real-time images, ultrasound is painless, free from radiation exposure, widely accessible, and more affordable than many alternative imaging methods, as illustrated in (**FigureI.4)**, which presents a comparison between ultrasound and mammography.**[9]**

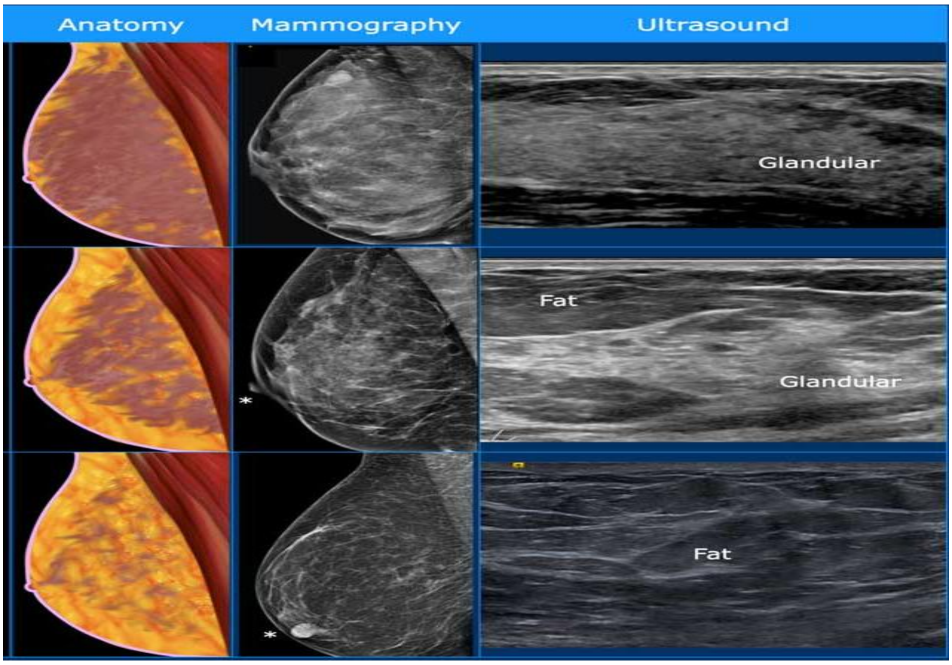


Figure I. 4 : Comparison Mammogram & Ultra sound images

I.6.3 Breast MRI (Magnetic Resonance Imaging)

Breast magnetic resonance imaging (MRI) is a highly sensitive imaging technique that uses powerful magnets, radio waves, and a computer to produce detailed images of the breast's internal structures. It is often used in conjunction with mammography for detecting breast cancer, particularly in individuals at high risk due to factors such as a strong family history, dense breast tissue, or inherited gene mutations like BRCA1 and BRCA2. Breast MRI is also valuable after a cancer diagnosis to assess the extent of disease or evaluate the opposite breast. In some cases, it is used to investigate suspected implant ruptures or other complex breast conditions. Due to its high resolution, breast MRI can identify abnormalities that may not be visible on other imaging modalities, making it a crucial tool in both screening and treatment planning for breast cancer. **[10]**

I.6.4 Biopsy

A breast biopsy is a diagnostic procedure in which a small sample of breast tissue is removed and examined under a microscope by a pathologist to determine whether it is cancerous or benign. It is typically recommended when imaging tests—such as mammograms, ultrasounds, or MRIs—reveal a suspicious area, or when physical signs such as lumps, thickening, or abnormal nipple changes are detected. The biopsy provides a definitive diagnosis, guiding further medical decisions such as the need for surgery, treatment, or continued monitoring.**[11]**

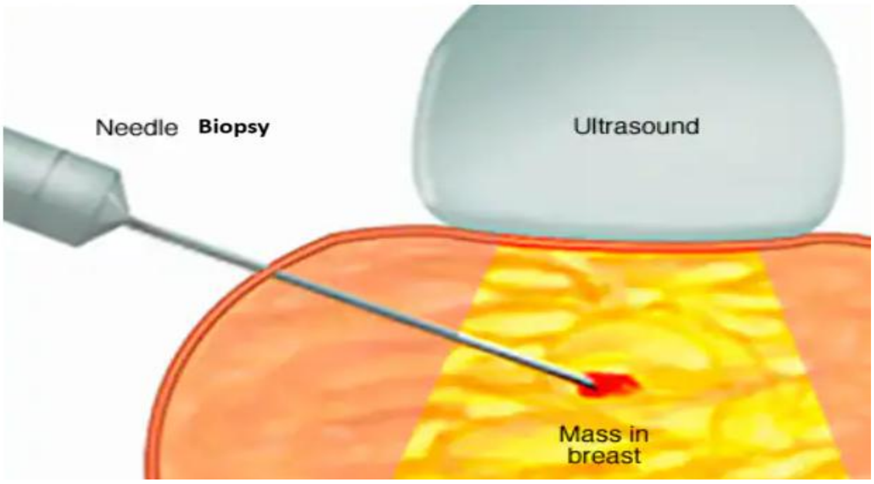


Figure I. 5 : Biopsy

I.7 Stages of Breast Cancer

There are five stages of breast cancer, including zero through four,The stage of breast cancer is also described by the "TNM" system:

- **T: Tumor** size (in centimeters)
- **N: Number** of nearby lymph nodes with cancer
- **M:** Whether the cancer has **metastasized** or spread to other organs of the body (0 = no spread, 1 = it has spread). **[12]**

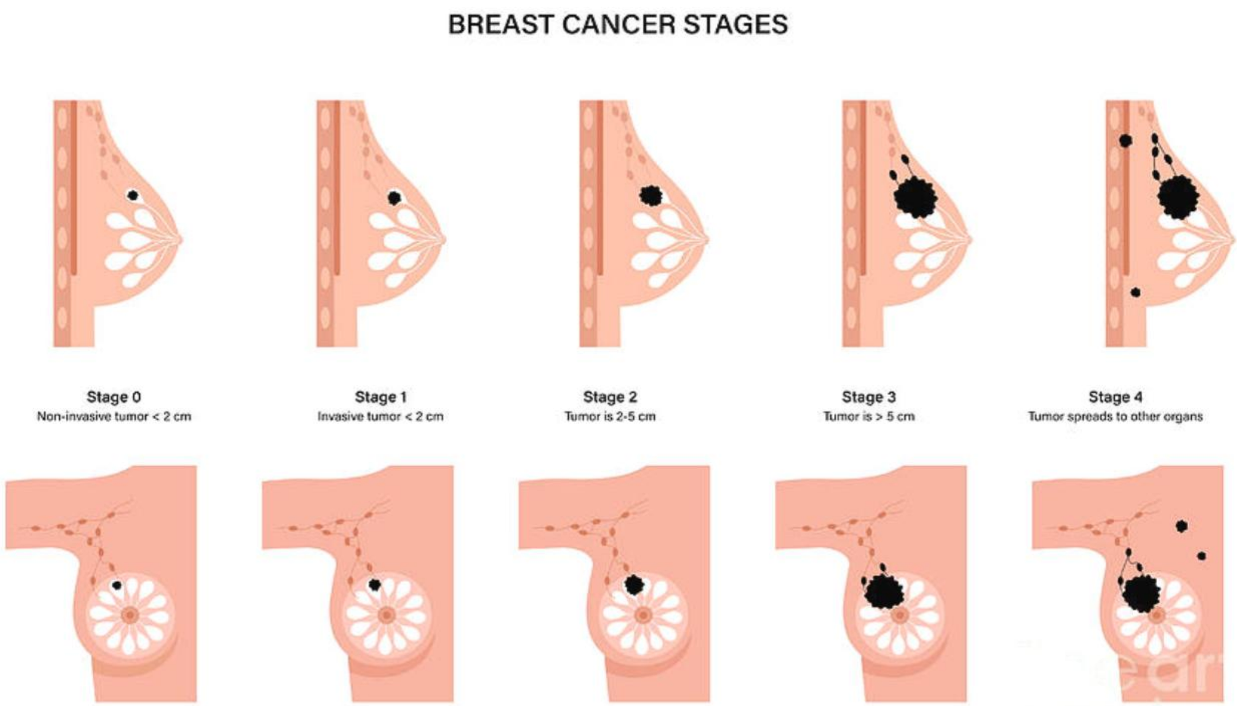


Figure I. 6 : Stages of Breast Cancer

1) Stage 0

The disease is only in the ducts or lobules of the breast. It has not spread to the surrounding tissue. It is also called noninvasive cancer.

2) Stage 1

The tumor is small, measuring up to 2 cm, and may or may not have spread to a few nearby lymph nodes. At this stage, the cancer is invasive but still limited to the breast area, meaning it has not spread to distant parts of the body.

3) Stage 2

At this stage, the tumor may be between 2 and 5 cm, or cancer may have spread to several nearby lymph nodes. The cancer is still considered early but shows greater local involvement than Stage I. Treatment aims to remove the tumor and prevent further spread.**[13]**

4) Stage 3

the tumor may be larger than 5 cm and has often spread to many lymph nodes or nearby tissues like the chest wall or skin. It has not yet spread to distant organs.

5) Stage 4

Stage 4 means the cancer has spread to distant parts of the body such as bones, lungs, liver, or brain. It is the most advanced stage and is considered incurable but treatable. **[14]**

I.8 Treatment Options

The treatment of breast cancer depends on the stage, type, hormone receptor status, and the overall health of the patient. Most patients receive a combination of treatments to ensure the best outcomes. Below are some treatment options:

I.8.1 Surgery

Surgery is often the first step in treatment, especially for early-stage breast cancer. There are two main types:

- **Lumpectomy**: Removal of the tumor and a small margin of surrounding tissue.
- **Mastectomy**: Removal of the entire breast, sometimes including nearby lymph nodes. **[15]**

I.8.2 Radiation Therapy

Radiation uses high-energy rays to target and destroy cancer cells. It is usually given after surgery to eliminate remaining cancer cells in the breast, chest wall, or lymph nodes. **[16]**

I.8.3 Chemotherapy

Chemotherapy targets rapidly dividing cells, such as cancer cells. However, it also affects healthy cells that grow quickly, which can cause side effects like fatigue, low white blood cell counts, early menopause, and hair loss.**[17]**

I.8.4 Immunotherapy

Immunotherapy strengthens the immune system, helping it to better detect and destroy cancer cells. However, this increased immune activity can sometimes cause the body to attack healthy cells as well, leading to inflammation. **[18]**

I.9 Conclusion

In summary, breast cancer remains a major public health issue, affecting millions of women worldwide. This chapter provided an overview of its definition, risk factors, types, diagnostic methods, and available treatments. Early detection through regular screening and awareness plays a key role in improving survival rates and treatment effectiveness. The variety of treatment options—ranging from surgery to immunotherapy—reflects the complexity and individuality of each case. In the following chapter, we will explore the application of image processing techniques in breast cancer detection. By integrating advanced medical imaging and analysis tools, we aim to enhance diagnostic precision and contribute to more personalized, accurate, and timely interventions.

Chapter II : Image Processing

II.1 Introduction

Image processing is a vast and continuously evolving field that has experienced significant advancements in recent decades. It plays a crucial role in various domains such as **medicine, robotics, surveillance, remote sensing**, and many others.

Image processing encompasses a set of **methods and techniques** aimed at analyzing, modifying, and interpreting an image to extract relevant information or enhance its visual quality for better human perception. Among these tasks, **segmentation** plays a central role, as it allows an image to be divided into **homogeneous regions or meaningful objects**, facilitating their identification and analysis.

The main goal of image processing is to **optimize these operations**, making them **more efficient, precise, and adaptable** to various applications.

In this chapter, we will first introduce the **fundamental concepts** of image processing to establish a solid understanding of the different approaches. Then, we will explore **classical segmentation techniques**, which form the foundation for more advanced methods.

The objective of this chapter is to provide a **comprehensive understanding of classical segmentation techniques**, highlighting their principles and how to works .

II.2 What is image processing

Image processing, a branch of signal processing, encompasses all operations performed on images and videos. These operations serve two main purposes: improving image quality (through contrast enhancement, noise removal, or blur correction) and extracting useful information like segmentation and edge detection. These techniques optimize image readability and facilitate interpretation **[19] .** see (**Figure II.1**)

Figure II. 1 : Image before and after enhancement **[20]**

Before any image processing, preprocessing operations aim to enhance image quality.

Image processing objectives can be divided **into three levels [21]**

II.3 Image processing levels

II.3.1 Low-level Processing

- Input: Image

- Output: Image

- Focuses on basic image transformations

Operations: Image enhancement, color correction, noise reduction, filtering, brightness adjustment, etc.

Function: Preprocessing and image manipulation to improve quality before further analysis**.**

II.3.2 Mid-level Processing

- Input: Image

- Output: Measurements/attributes

- Extracts specific features from images

-Extracts numerical image characteristics (edges, texture)

-Analyzes without interpretation

Operations: Image segmentation, thresholding, edge detection, median filtering, binarization, and morphological techniques.

Function: Feature extraction and segmentation to identify objects or structures in the image.

II.3.3 High-level Processing

- Input: Image

- Output: High-level description

- Provides advanced interpretation of image content

Operations: Use of statistical methods, machine learning, neural networks, and deep learning.

Function: Image recognition and interpretation, such as classification or analysis of detected objects.

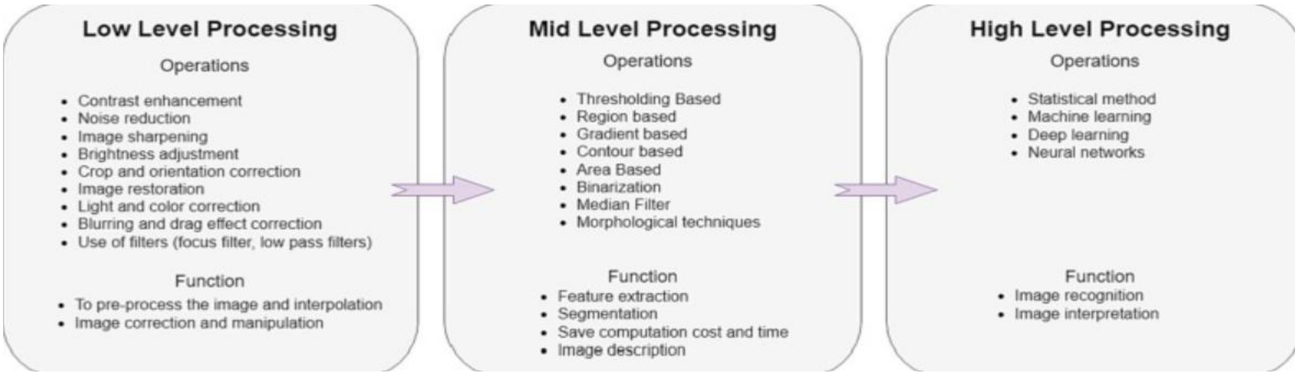


Figure II. 2 : Image processing levels

These processes can be broadly categorized into two types:

1. Image-to-Image transformations: producing modified images as output

2. Feature extraction: generating attributes or measurements as output

II. 4 What is an image

An image is a planar representation of an object, mathematically expressed as a 2D function f(x, y). At coordinates (x0, y0), f(x0, y0) represents the intensity or gray level value. This creates a continuous 2D process from physical measurements.

When generated by physical processes, f(x, y) corresponds to radiated energy from sources like: Light waves (emission and reflection) , Infrared radiation , X-rays , Ultrasound. **[22]**

Example

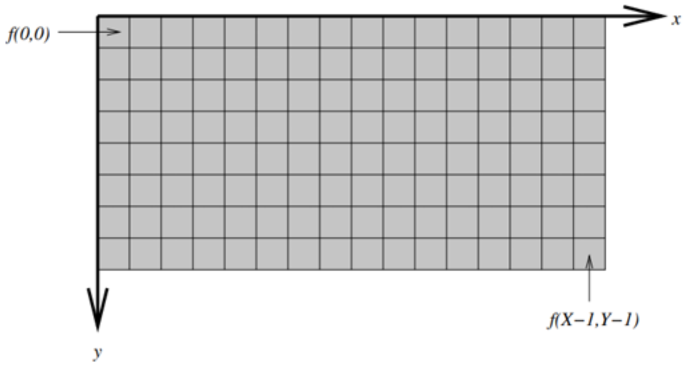


Figure II. 3 : A rectangular digital image of resolution 16 x 8

When the amplitude values of ff and the coordinates (x,y)(x, y) are discrete quantities, the image is called a digital image. A digital image is therefore composed of a finite number of elements, each having a specific location and value. These elements are called pixels (a contraction of **PIC**ture **EL**ement).

An image can be represented in three dimensions (3D)using x, y, and z coordinates. In this format, pixels are organized as a matrix. This structure is known as an RGB image, where R represents red, G stands for green, and B signifies blue. For grayscale images, there is only a single channel, which means z equals 1. **[23]**

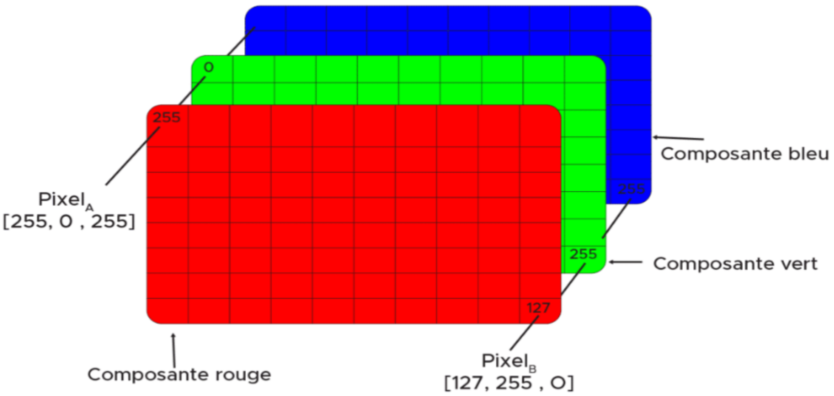


Figure II. 4 : RGB image

II.5 Image Data Type

Different types of image data storage formats are determined by both image content and storage requirements.

II.5.1 Binary images

are the simplest form of digital images, consisting of 2D arrays where each pixel is assigned either a 0 or 1 value. Also known as logical images, they represent black pixels with 0 (off) and white pixels with 1 (on), making them ideal for simple document scanning and pattern recognition tasks.

II.5.2 Grayscale images

are represented by 2D arrays where each pixel value corresponds to an intensity level. These images store varying degrees of gray between black and white, with the range of possible values determined by the image's bit resolution. The specific format used for storage depends on the application and system requirements.

II.5.3 RGB color images

are more complex, using 3D arrays where each pixel contains three numerical values representing the red, green, and blue color channels. These images can be conceptualized as three separate 2D planes, with dimensions of C (columns) by R (rows) by 3 (channels). The data is typically stored as sequential integers in channel order (R0 G0 B0, R1 G1 B1, etc.) and can be accessed using coordinates in the format I(C, R, Channel). **[24]**

Grayscale images **Binary images** **RGB color images**

$I(x,y) \in [0..255]$ $I(x,y) \in \{0 , 1\}$ $I_R(x,y)$; $I_G(x,y)$; $I_B(x,y)$

Figure II. 5 : types of image

II.6 The main stages of image processing

Figure II. 6 : Flow chart of the image-processing steps

II.6.1 Image Acquisition

Image acquisition initiates the processing sequence as a critical foundation step. It encompasses capturing images while considering multiple essential factors: problem requirements, image specifications, storage capacity, capture timing, resolution, lighting conditions, and camera parameters. The quality of images obtained during this phase directly influences the success of all subsequent processing stages. **[25]**

II.6.2 Image preprocessing

During preprocessing, the raw image undergoes various transformations to prepare it for analysis. This stage focuses on optimizing the image while preserving essential information, typically involving RGB to grayscale conversion, noise reduction through filtering, and contrast enhancement. These operations aim to reduce computational complexity while maintaining image integrity for further processing. **[25]**

II.6.3 Feature Localization

Feature localization involves identifying and isolating specific regions of interest within the processed image through **segmentation techniques**. This stage is particularly crucial in specialized applications like medical imaging, where precise detection of structures is essential. For instance, in ophthalmology, accurate optic disk detection enables early diagnosis of conditions like glaucoma.

II.6.4 Feature extraction

The feature extraction stage focuses on obtaining relevant information from previously identified regions of interest. This process involves analyzing various characteristics such as texture, shape, edges, and intensity patterns. The extracted features provide essential data that serves as input for subsequent analysis and classification tasks.

II.6.5 Display & interpretation

The final stage encompasses analyzing the extracted features and presenting findings. This involves evaluating key results and displaying them in appropriate formats for decision-making purposes. In medical applications, this might include detecting abnormalities, diagnosing conditions, or monitoring treatment effectiveness through visual or numerical representations of the processed data. **[25]**

Segmentation is a key element in the **localization of features** before extracting information for final analysis **[26]**

Segmentation is an essential step in **image processing** that involves dividing an image into homogeneous regions. Its purpose is to extract objects of interest.

II.7 Traditional Approach of Segmentation

Classical segmentation techniques As shown in **(Figure II. 7)** **[27]**

II.7.1 Image segmentation

is a technique that divides a digital image into distinct groups of pixels (image segments) to facilitate object detection and related tasks. By breaking down the complex visual data of an image into specific-shaped segments, image segmentation enables faster and more advanced image processing. **[28]**

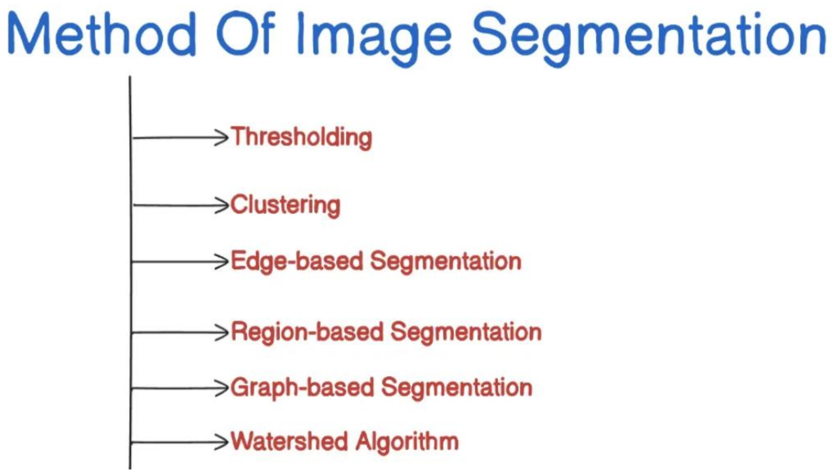


Figure II. 8 : Traditional Approach of Segmentation

II.7.2 Threshold Method

Thresholding represents a fundamental technique in image segmentation that converts grayscale images into binary format through a straightforward yet effective process. This method operates by establishing a threshold value that divides pixels into two distinct classes: those with values above the threshold are assigned a value of 1 (appearing as white), while those below are set to 0 (appearing as black). The process, commonly known as binarization, is particularly effective when there exists a significant contrast between the pixel values of the target classes, making it straightforward to select an appropriate threshold value. This conversion to a binary map serves as an essential preprocessing step for more advanced image processing algorithms, such as contour detection and identification, which specifically require binary input images to function effectively. **[29]**Thresholding can be more intuitively described as follows: **[30]**

$$
\begin{array}{l}
(x,y) = 1, \quad (x,) \geq T \\
\quad\quad 0, \quad (x,) < T
\end{array}
$$

I(x,y) is the pixel value of the grayscale image at position (x,y);
B(x,y) is the pixel value of the binary image at position (x,y);
T is the chosen threshold.

Figure II. 9 : Threshold Method

Within image processing, thresholding methodologies can be taxonomically bifurcated into **global** and **local** thresholding, contingent upon the modality of threshold selection and its subsequent application **[31].** As shown in **(Figure II.2)**

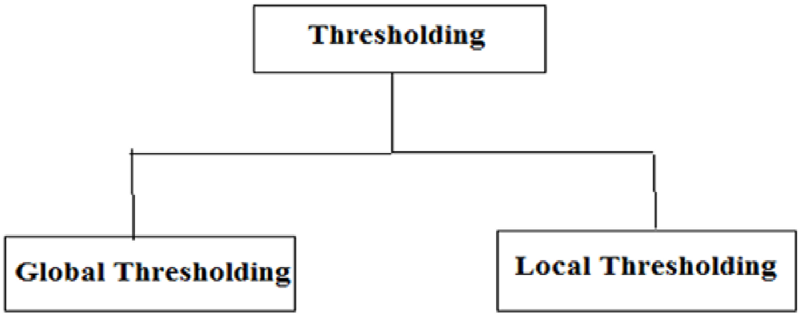


Figure II. 10 : Thresholding Techniques

a) Global Thresholding

Global thresholding is a segmentation approach that employs a single, uniform threshold value determined from either the image's homogeneous characteristics or its overall histogram analysis. This technique proves most effective when there exists a clear distinction in grayscale distribution between the foreground (region of interest) and background regions. Several established methods fall under global thresholding, including Otsu's Method, Iterative Thresholding, Minimum Error Thresholding, and Entropy-based Thresholding, each offering different approaches to determining the optimal threshold value. **[30]**

b) Local Thresholding

Local thresholding techniques represent an advanced approach to image segmentation, particularly effective in handling images with non-uniform illumination caused by shadows or directional lighting, where traditional uniform thresholding methods often fail. These techniques employ sophisticated algorithms that analyze local image characteristics within specific neighborhoods, determining threshold values based on statistical parameters such as luminance, contrast, and texture patterns of surrounding pixels. **[30]**

II.7.3 Edge Based Method

Edge-based segmentation is a technique used in image processing to identify and separate the edges of an image from the background. The method involves detecting the abrupt changes in intensity or color values of the pixels in the image and using them to mark the boundaries of the objects. Two common tasks in this method are **edge detection** and **edge linking.[32]**

Original image After edge-segmented

Figure II. 11 : Edge Based Method

a) Edge Detection

In edge detection, we need to find the pixels that are edge pixels of an object. There are many object detection methods such as Sobel operator, Laplace operator, Canny, etc. **[33]**

Edge Detection Methods

1. **The Roberts Detection**

The Roberts Cross operator is a fast and simple method for computing the **2D spatial gradient** of an image, effectively highlighting regions with high spatial frequency, which often correspond to edges. Typically applied to grayscale images, both as input and output, it estimates the absolute magnitude of the spatial gradient at each pixel, providing a measure of intensity changes in the image. **[33]**

How It Works

- The operator computes the gradient in both **horizontal (Gx)** and **vertical (Gy)** directions.
- The magnitude of the gradient determines the presence of an edge.
- It highlights **sharp intensity changes**, indicating object boundaries.

$$\begin{array}{|c|c|} \hline +1 & 0 \\ \hline 0 & -1 \\ \hline \end{array} \qquad \begin{array}{|c|c|} \hline 0 & +1 \\ \hline -1 & 0 \\ \hline \end{array}$$

Gx Gy

Roberts Cross convolution kernels

Gradient Magnitude Calculation

Once the gradients Gx and Gy are computed, the edge strength is determined using:

$$|G| = \sqrt{Gx^2 + Gy^2}$$

Alternatively, for faster computation:

$$|G| = |Gx| + |Gy|$$

2. The Prewitt Detection

It also detects vertical and horizontal edges of an image using **3×3 convolution kernels**. It is one of the best ways to detect the orientation and magnitude of an image. It uses the kernels or masks .

How It Works

Mx = Mask for detection of vertical edges

When we apply this mask on the image it prominent vertical edges. It simply works like as first order derivate and calculates the difference of pixel intensities in a edge region. As the center column is of zero so it does not include the original values of an image but rather it calculates the difference of right and left pixel values around that edge. This increase the edge intensity and it become enhanced comparatively to the original image.

My= Mask for detection of horizontal edges

The horizontal edge detection mask operates by emphasizing edges oriented horizontally in an image through a difference calculation of pixel intensities. The mask's structure, featuring a center row of zeros, specifically ignores the original edge values while computing the intensity differences between pixels above and below the edge. This design enhances the visibility of edges by amplifying sudden intensity changes. The mask follows derivative principles, characterized by opposing signs within the mask and a sum that equals zero. These masks are standardized, meaning their values are fixed and cannot be modified, unlike some other operators where mask values might be adjustable. **[34]**

$$M_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad M_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

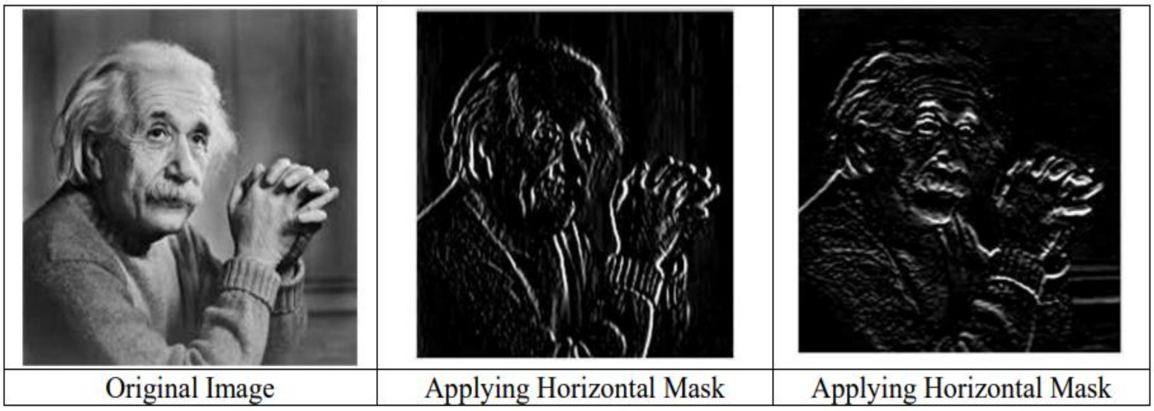


Figure II. 12 : Example of Prewitt Detection

3. The Sobel Detection

The **Sobel operator** is very similar to the **Prewitt operator**, as both are **derivative masks** used for **edge detection**. **[34]** Like the Prewitt operator, the Sobel operator is designed to detect two types of edges in an image:

- **Vertical edges**
- **Horizontal edges**

Difference from the Prewitt Operator

The key distinction between the **Sobel** and **Prewitt** operators lies in the **mask coefficients**. In the Sobel operator, these coefficients are not fixed and can be adjusted based on specific requirements, provided they still satisfy the properties of derivative masks.

Vertical Edge Detection

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

The **vertical mask** of the Sobel operator functions similarly to that of the Prewitt operator, with one major difference: it assigns **"2" and "-2"** to the center of the first and third columns. This modification enhances the detection of **vertical edges** when applied to an image.

Horizontal Edge Detection

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

Similarly, the **horizontal mask** detects edges in the **horizontal direction**. The distinction here is that the **zero column is oriented horizontally**, and the center values of the first and third rows are **2 and -2**. This configuration ensures that when convolved with an image, it emphasizes **horizontal edges** more prominently. **[34] See (Figure II.3)**

Original Image	Applying Horizontal Mask	Applying Horizontal Mask

Figure II. 13 : Example of Sobel Detection

4. Laplacian Operator

The Laplacian operator is also a derivative operator used to detect edges in an image. The key difference between the Laplacian and other operators like Prewitt, Sobel, Robinson, and Kirsch is that these are **first-order derivative masks**, whereas the Laplacian is a **second-order derivative mask**. This operator is further classified into two types: the **Positive Laplacian Operator** and the **Negative Laplacian Operator**. **[35]**

Another difference between Laplacian and other operators is that unlike other operators Laplacian didn't take out edges in any particular direction but it take out edges in following classification.

- **Inward Edges**
- **Outward Edges**

How Laplacian operator works

- **Positive Laplacian Operator**

In Positive Laplacian we have standard mask in which center element of the mask should be negative and corner elements of mask should be zero.

0	1	0
1	-4	1
0	1	0

Positive Laplacian Operator is use to take out outward edges in an image.

- **Negative Laplacian Operator**

In negative Laplacian operator we also have a standard mask, in which center element should be positive. All the elements in the corner should be zero and rest of all the elements in the mask should be -1

0	-1	0
-1	4	-1
0	-1	0

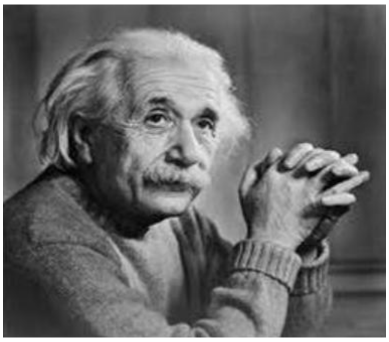
Negative Laplacian operator is use to take out inward edges in an image.

The Working Principle of the Laplacian Operator

The Laplacian operator functions as a derivative operator that emphasizes gray-level discontinuities while minimizing regions of gradual gray-level variation in images. When applied, it creates images featuring grayish edge lines and discontinuities against a dark background, effectively identifying both inward and outward edges.

The application of Laplacian filters follows specific rules. A key consideration is that positive and negative Laplacian operators cannot be applied simultaneously to the same image. When using the positive Laplacian operator, the resulting image must be subtracted from the original to obtain the sharpened image. Conversely, when applying the negative Laplacian operator, the resulting image must be added to the original to achieve the sharpened effect.

Each operator type produces different effects: the positive operator identifies transitions from light to dark (inward edges), while the negative operator highlights transitions from dark to light (outward edges). This dual capability makes the Laplacian operator particularly versatile in edge detection applications.



Original image

applying Positive Laplacian Operator **applying Negative Laplacian Operator**

Figure II. 14 : Example of Laplacian Operator

b) Edge Linking:

to refine the edge detection by linking the adjacent edges and combine to form the whole object. The edge linking can be performed using any of the two methods below:

- **Local Processing:**

This method uses gradient and direction to link neighboring edges. If two edges have similar direction vectors, they can be connected. **[36]**

- **Global processing:**

This approach relies on the **Hough Transform (HOG) (** Hough transform: The Hough transform is an incredible tool that lets you identify lines. Not just lines, but other shapes as well**)** to detect and link edges across the entire image. **[36]**

II.7.4 Region Based Method

Region-based segmentation is a technique used in image processing to divide an image into distinct regions based on similarity criteria, such as color, texture, or intensity. The primary goal of segmentation is to partition an image into coherent regions. Unlike other approaches, region-based segmentation techniques directly identify regions by grouping pixels into clusters based on their similarities, then merging or splitting these regions until the desired level of segmentation is achieved. Generally There are two methods for Region Based Segmentation **[37].** As shown in **(Figure II.4) [38]**

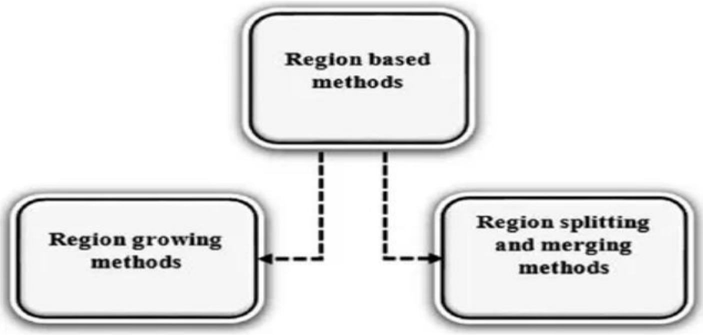


Figure II. 15 : Methods for Region Based Segmentation

a) Region Growing

This method recursively grows segments by including neighboring pixels with similar characteristics. It uses the difference in gray levels for gray regions and the difference in textures for textured images. **[38]**

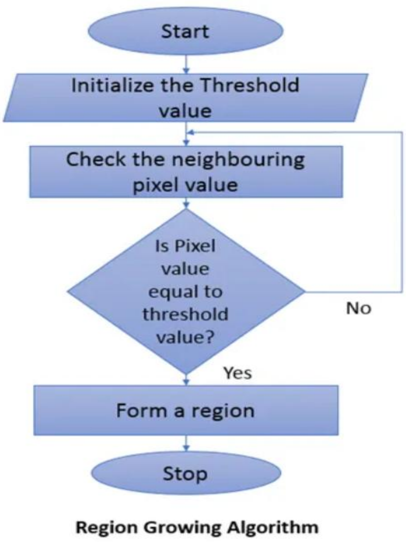


Figure II. 16 : Region Growing algorithm

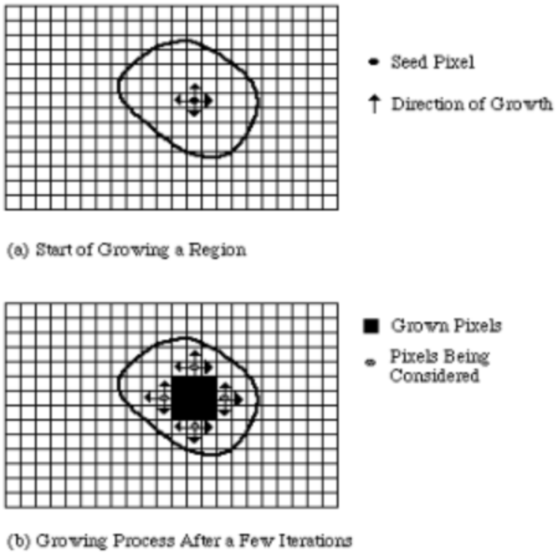


Figure II. 17 : Region Growing Method

b) Region Splitting and Merging

Split and merge segmentation is a region-based technique that recursively divides an image into smaller regions until a stopping condition is satisfied, after which it combines similar regions to create larger ones. This method works by splitting the image into smaller blocks or regions and then merging adjacent regions that meet specific similarity criteria, such as color or texture. Split and merge segmentation is a straightforward and efficient approach for image segmentation, but it may struggle with complex images containing irregular or overlapping regions. **[39] As shown in (Figure II.5)**

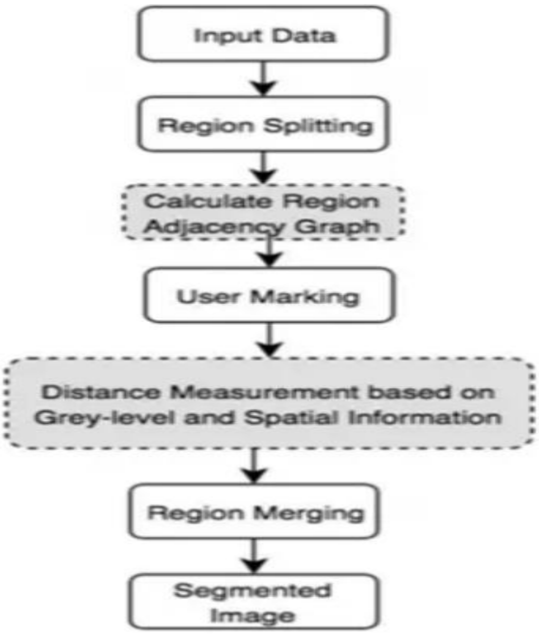


Figure II. 18 : Region Splitting and Merging **[40]**

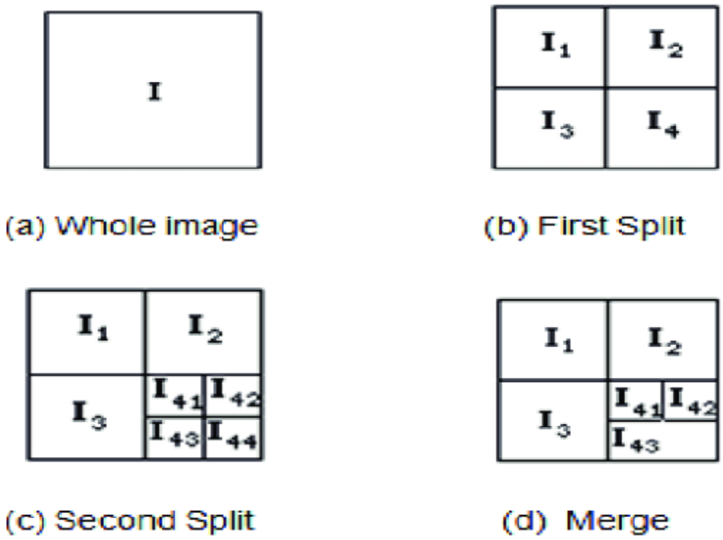


Figure II. 19 : Example of region splitting and merging

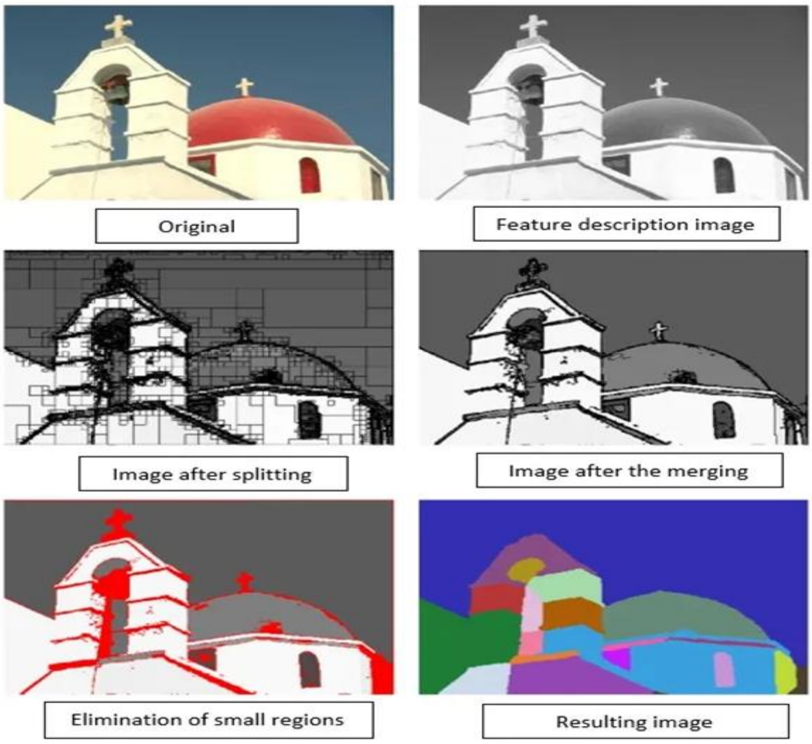


Figure II. 20 : Example of Region Splitting and Merging

II.7.5 Clustering Method

Clustering is one of the most popular techniques used for image segmentation, as it can group pixels with similar characteristics into clusters or segments. The main idea behind clustering-based segmentation is to group pixels into clusters based on their similarity, where each cluster represents a segment. This can be achieved using various clustering algorithms, such as K means clustering, mean shift clustering.

The process of image segmentation by clustering can be carried out using two methods. **[41]**

- **Agglomerative clustering :**

each pixel starts as an **individual cluster**. Then, the **closest clusters** (with the smallest inter-cluster distance) are gradually merged. This process continues until the final clusters are formed.

- **Divisive clustering :**

In **divisive clustering**, all pixels start in **one large cluster**. The cluster is then **split** into smaller groups based on the largest inter-cluster distance. This process repeats until the **optimal number of clusters** is reached. **[41]**

a) K-means clustering

K-Means Clustering is an **agglomerative clustering** method where the labels of pixels or data points are unknown beforehand. The number of clusters is set to **K**, and these clusters are formed by grouping pixels with similar characteristics.

Steps of K-Means Clustering

1. Define a value for **K** (number of clusters). **[42]**
2. Select a feature from each pixel (e.g., RGB values).
3. Group similar pixels based on a distance metric (e.g., Euclidean distance).
4. Use cluster centers to refine the grouping iteratively.
5. Set a threshold to stop the algorithm when changes become minimal.

Figure II. 21 : Showing the result of segmenting the image at k=2,4,10.

II.7.6 Watershed Based Method

The **Watershed Algorithm** is a classical image segmentation technique based on the concept of watershed transformation. This segmentation process uses the similarity between adjacent pixels as a key criterion to group pixels with similar spatial positions and gray values. And **The Watershed Algorithm** is used when segmenting images with touching or overlapping objects. It excels in scenarios with irregular object shapes, gradient-based segmentation requirements, and when marker-guided segmentation is feasible. **[43]**

Interprets the gradient magnitude of an image as a topographic surface, where high gradient magnitudes are interpreted as edges.

How the Watershed Algorithm Works

The Watershed Algorithm segments an image using topographic information, treating pixel intensity as elevation. It identifies **catchment basins** and **watershed lines** to separate objects.

Steps of the Watershed Algorithm

1. **Marker Placement**: Markers are placed at local minima (lowest intensity points) to serve as starting points.
2. **Flooding Process**: The image is flooded with colors from the markers, filling catchment basins until object boundaries are reached.
3. **Catchment Basin Formation**: As flooding progresses, distinct regions are formed and assigned unique colors.
4. **Boundary Identification**: The borders between regions define object boundaries, aiding in object recognition and image analysis. **[48]**

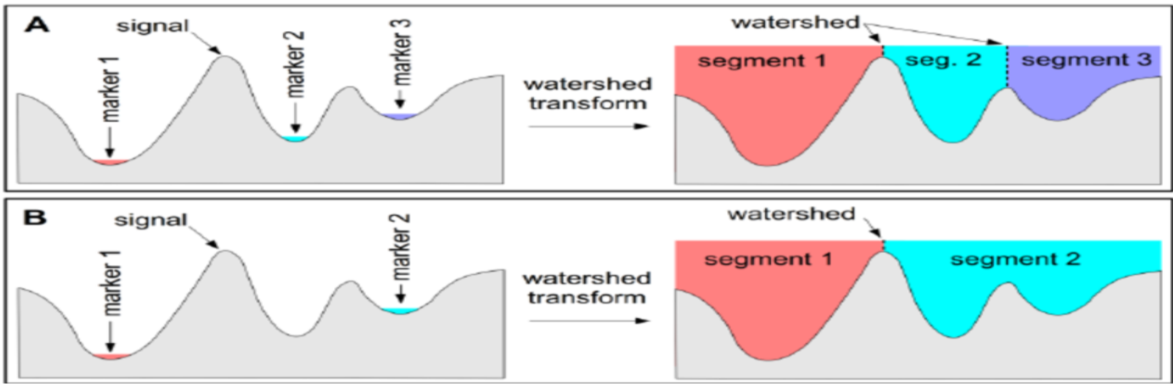


Figure II. 22 : Examples of the watershed transform

II.7.7 Graphic-based segmentation

Graph-based segmentation converts an image into a graph structure for advanced analysis and efficient segmentation. In this approach, pixels are represented as nodes, and their relationships are defined by edge weights. Graph theory algorithms are then applied to partition the image into meaningful regions. **[44]**

How Graph-Based Segmentation Works

Graph-based segmentation follows a series of steps to divide an image into meaningful regions:

1. **Graph Representation:** The image is represented as a graph where each pixel is a node. Edge weights are assigned based on pixel similarity, such as color or intensity differences.

2. **Graph Partition:** Graph partitioning is a segmentation method that splits a graph into separate regions by minimizing the edge weights between them. Common algorithms include **spectral clustering**, **normalized cuts**, **graph cuts**, and **minimum spanning tree** methods. This step plays a key role in graph-based segmentation (GBS), helping to produce accurate and meaningful segments.**[45]**

3. **Segment Refinement:** Segments are adjusted by merging or splitting them based on factors like size, shape, or texture. **[44]**

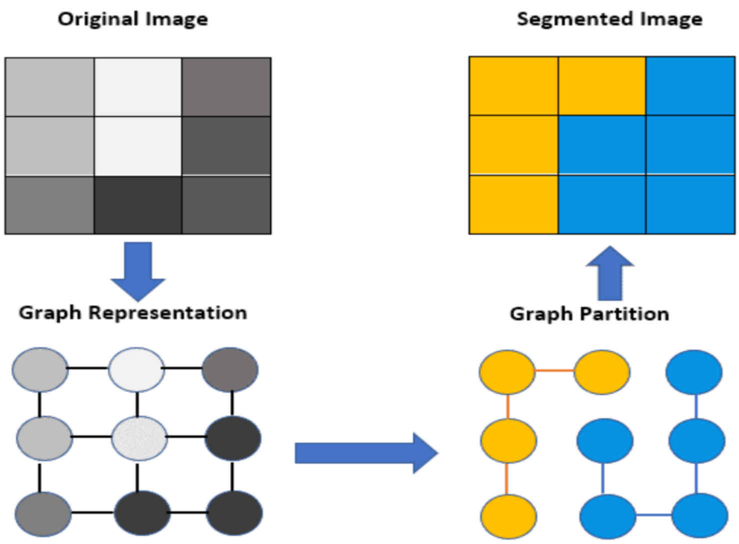


Figure II. 23 : Graph-Based Segmentation

II.8 Conclusion

In this chapter, we have introduced the fundamental concepts of image processing and emphasized the importance of segmentation as a crucial step in analyzing and interpreting images. We explored classical segmentation techniques, which serve as the foundation for more advanced methods used in modern applications. Understanding these techniques is essential for developing more sophisticated approaches that improve accuracy, efficiency, and adaptability in various domains. The knowledge gained from classical segmentation methods provides a strong basis for exploring more advanced techniques, such as machine learning-based and deep learning-based segmentation, which will be discussed in the following chapters.

Chapter III: Deep learning & Segmentation

III.1 Introduction

Artificial Intelligence (AI) is at the center of today's digital world, enabling machines to perform tasks requiring human intelligence.AI needs data to learn, adapt, and improve, it works depends directly on the quality and variety of the data that it learns from, much like humans rely on experience to develop skills. Data can come from various sources such as sensors on machines, smart devices, databases, computer records, or websites.

AI is an combination of Machine Learning**(**ML**)** techniques and Deep Learning **(**DL**)**decisions. (ML) allows systems to learn patterns from data, while (DL)based on artificial neural networks has shown outstanding results in fields requiring complex data processing. Among (DL) applications, **image segmentation** stands out, especially in medical imaging. Deep learning-based segmentation (DLS) methods, such as convolutional neural networks (CNNs) and encoder-decoder architectures, enable accurate segmentation and localization of specific regions within images, which is crucial for tasks such as identifying tumor boundaries.

In this chapter, we'll explore the interesting world of AI, machine learning(ML), with a focus on Deep learning-based segmentation (DL). **[46]**

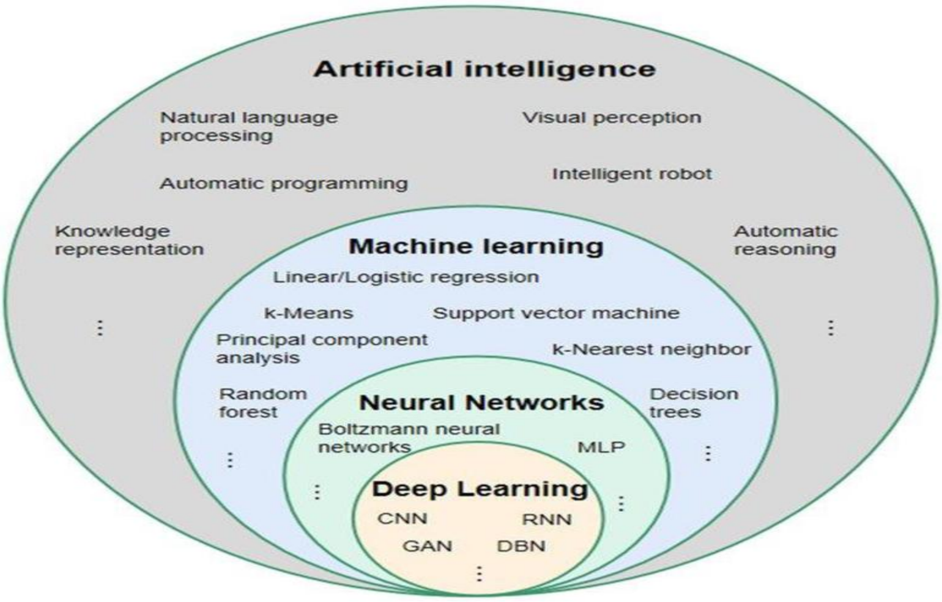


Figure III. 1 : Relation between AI, ML, NN, and DL.

III.2 Artificial intelligence

III.2.1 Definition

Artificial Intelligence is machine-displayed intelligence that simulates human behavior or thinking and can be trained to solve specific problems through a set of rules(algorithm). One of the known definitions of AI, was formulated by the pioneer **Marvin Lee Minsky** as « the science of making machines do things that would require intelligence if done by men. »

III.2.2 Types of Artificial Intelligence

Artificial Intelligence (AI) can be classified into various types depending on its capabilities and functions. Below are the key categories:

a) Based on Capabilities

- **Narrow AI (Weak AI)**

Currently, this is the most common type of AI. Narrow AI is designed for performing specific tasks, frequently surpassing human capabilities for example in image recognition software, spam filters, and chess-playing computers. **[47]**

- **General AI**

Known as general AI or powerful AI. that AI has the ability to understand, learn, and do multiple activities and serve as realistic, intelligent helpers to people in daily life is the aim of artificial general intelligence design. **[48]**

- **Superintelligence AI**

Super AI, often known as artificial superintelligence (ASI), is the stuff of science fiction. It is predicted that if AI reaches the level of general intelligence, it would learn so quickly and will surpass even human intelligence in all aspects. **[48]**

b) Based on Functionality

- **Reactive machines AI**

Reactive Machines are AI systems that can react to inputs, but they lack the capacity to store memories, learn from mistakes. They are also limited in recollect previous outcomes or decisions, they only work with presently available data to perform a very specific task **[48]**

- **Limited Memory AI**

AI can store historical data to create predictions. It constructs a small, temporary knowledge base of its own, and uses that information to complete tasks. It allows machines to learn from experience and use data to improve accuracy, like **Chatbots with context or self-driving cars. [47]**

- **Artificial Intelligence Theory**

The term **«theory of mind»** describes that AI can sense and understand other people's feelings. The phrase is taken from psychology to describe how people may read other people’s emotions and use that information to forecast their own behavior in the future. **[48]**

- **Self-Aware AI**

One of the ultimate objectives in AI development is self-aware AI. Since self-aware AI would not only be able to experience other people's emotions but will also have a sense of self, it is believed that once this technology is developed, AI machines will be uncontrollable. **[48]**

III.2.3 Application of AI in Healthcare

AI has become an essential part of modern healthcare society, enhancing medical practice by speeding up the pace of research or providing doctors with advanced tools to better assist patients. A few examples are giving in the following subsections:

- **AI in medical imaging**

Artificial neural networks can detect disease symptoms just as accurately as many radiologists. With powerful computational resources, clinicians can store more medical images, making it easier to track a patient’s history. This crucial information also plays a key role in the treatment process.

- **Research assistance**

AI allows researchers to access large pools of data from various sources, which can be used by clinicians all over the world. Software companies and start-ups are providing AI tools to help track the progress of patients, recover crucial diagnosis data, and contribute to this information through shared networks. 46 Social media apps are used to share data worldwide and connect to other similar healthcare agencies for learning and sharing information. **[49]**

- **Error reduction**

AI can improve the safety of patients and AI safety tools can ensure accurate decision-making with improved error detection and drug management. AI-powered tools have made life easy for physicians and healthcare workers**.**

- **Early disease detection and treatment plans**

Machine Learning models analyze patient data to identify the factors and alert clinicians in cases of emergencies. For instance, diagnosing breast cancer scans and finding signs that might be easily overlooked, also it can assist clinicians by analyzing past cases to helps finding the best treatment for each patient based on what worked for similar conditions.

III.3 Machine Learning(ML)

IBM has been a leader in AI research since the field's early days in the **1950s**, when **Arthur Samuel** developed a checker player that learned from experience, this work was one of the earliest and most influential examples of machine learning **[50].**

Machine learning is a subset of AI, allows systems to automatically learn and improve from experiences by analyzing data to make prediction and decisions. It has become a crucial tool for a variety of cutting-edge technologies.

III.3.1 Machine learning approaches and algorithms

Machine learning methods are divided into categories according to their purpose and the main are the follow

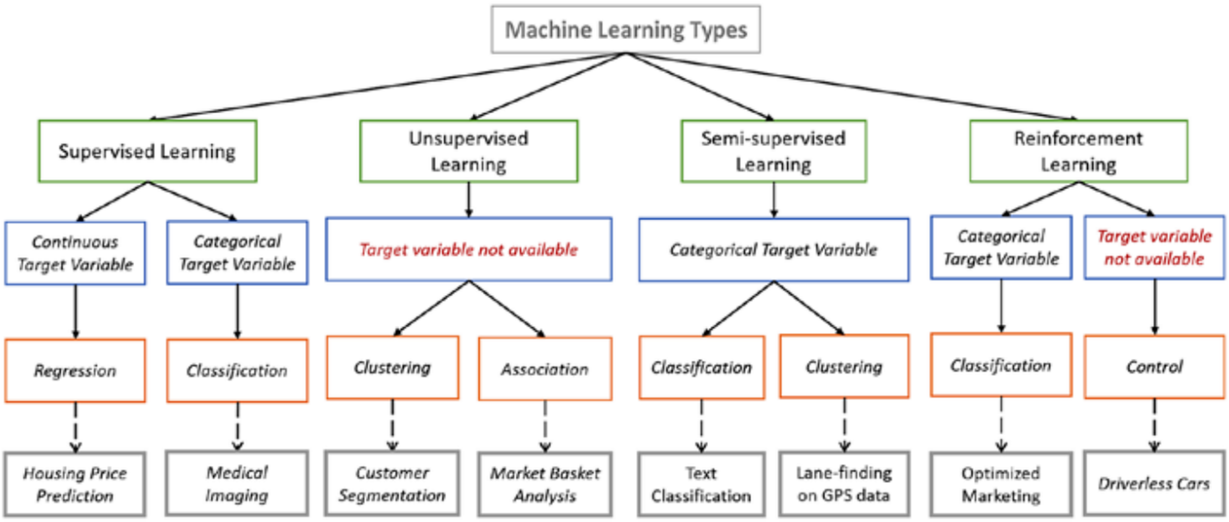


Figure III. 2 : Different types of ML algorithms and real life applications **[51]**

a) Supervised Learning

Supervised machine learning is a type of machine learning where the model is trained on a labeled dataset (i.e., the target or outcome variable is known). It is commonly used for risk assessment, image recognition, predictive analytics and fraud detection. **[52]** Supervised learning comprises algorithms such as: classification and regression.

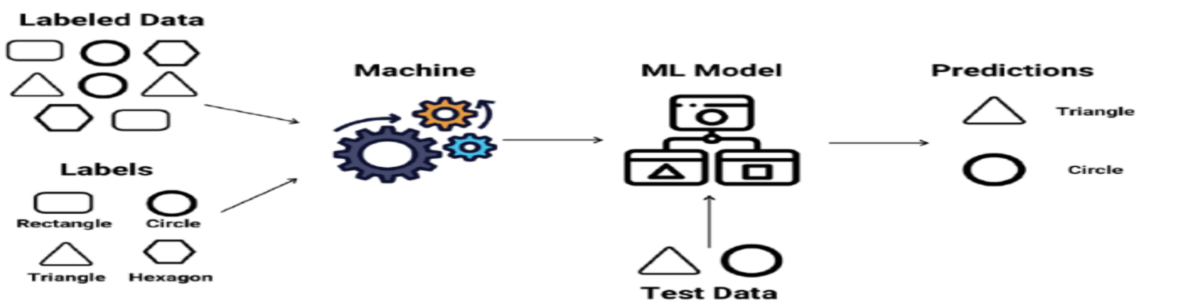


Figure III. 3 : Example of Supervised machine learning **[53]**

- **Classification**: predict categorical output variables (e.g., “junk” or “not junk”) by labeling pieces of input data. Classification algorithms include logistic regression, k-nearest neighbors and support vector machines (SVMs), among others.

- **Regression**: predict output values by identifying linear relationships between real or continuous values (e.g., temperature, salary). Regression algorithms include linear regression, random forest and gradient boosting, as well as other subtype.

b) Unsupervised learning

Unsupervised learning is the training of machine using information that is neither classified nor labeled and allowing the algorithm to act on that information without guidance. Here the task of machine is to group unsorted information according to similarities, patterns and differences without any prior training of data. **[52]**

Unsupervised learning problems can be categorized into clustering, association and dimensionality reduction. The figure below illustrates the process.

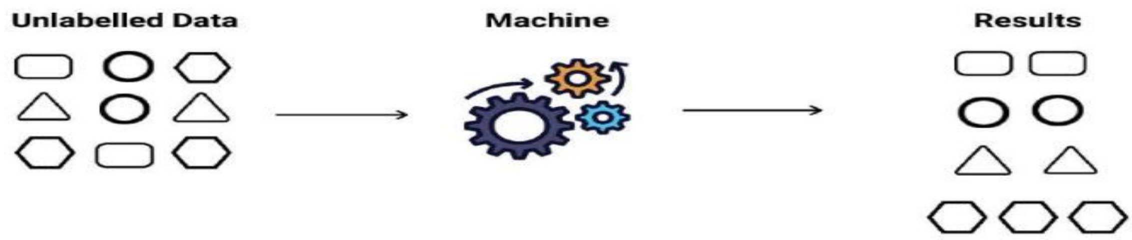


Figure III. 4 : Example of Unsupervised machine learning **[53]**

III.4 Neural Networks

Neural network consists of neurons, which are the fundamental units akin to brain cells. These neurons receive inputs, process them, and produce an output. They are organized into distinct layers: An Input Layer that receives the data, several Hidden Layers that process this data, and an Output Layer that provides the final decision or prediction**. [54]**

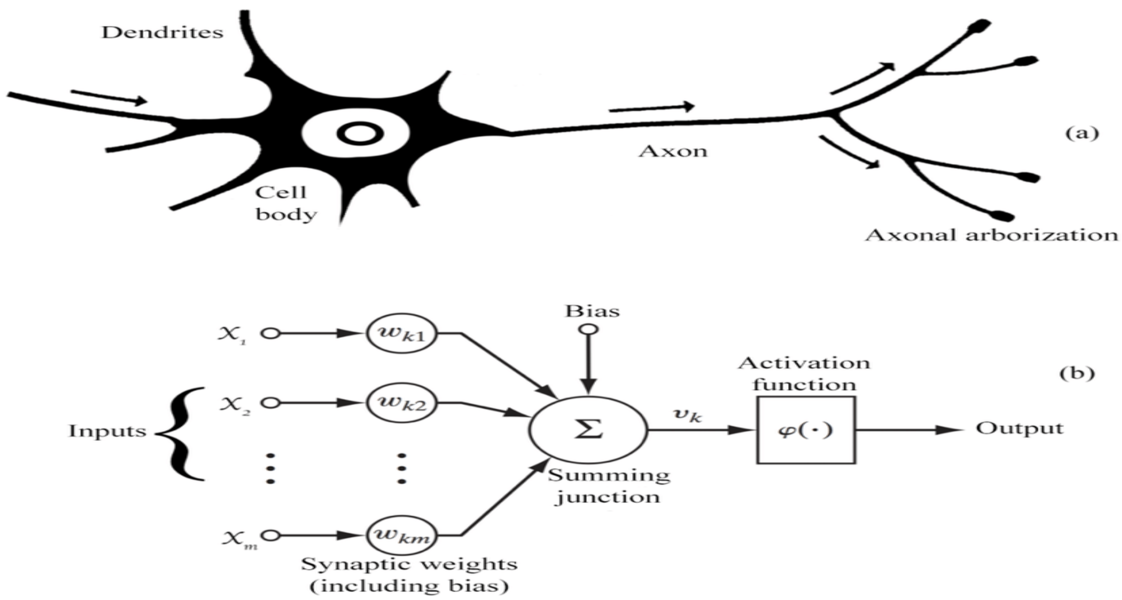


Figure III. 5 : Similarity between biological and artificial neural networks

The artificial neuron is composed of seven basic elements, namely: **[55]**

- **Multiple inputs signals** ($x_1, x_2, \ldots, x_n$) represent values from the external environment corresponding to the variables of a particular application. The input signals are usually normalized to enhance the computational efficiency of learning algorithms.
- **Synaptic weights** ($w_{k_1}, w_{k_2}, \ldots, w_{k_n}$) are the values used to weight each input variable, quantifying their relevance to the neuron functionality.
- **Linear aggregator** ($\sum$) computes a weighted sum of the input signals using their corresponding synaptic weights to produce an activation voltage.
- **Activation threshold or bias** (b) sets the threshold required for the output activation. The neuron fires only if the result from the linear aggregator meets or exceeds this threshold.
- **Activation potential** (u_k) is the result produced by the difference between the linear aggregator and the activation threshold. If this value is positive, i.e. if $u \geq b$, then the neuron produces an excitatory potential; otherwise, it will be inhibitory response.
- **Activation function** (g) whose goal is limiting the neuron output within a predefining range ensuring stability and meaningful signal propagation.
- **Output signal** (y) consists on the final value produced by the neuron to a given set of input signals and can also serve as an input for other interconnected neurons.

III.4.1 Neural Network Architecture

Neural Network Architecture composes of three layers, each playing a crucial role in processing and transforming data. Below is an explanation of these layers and their functions.

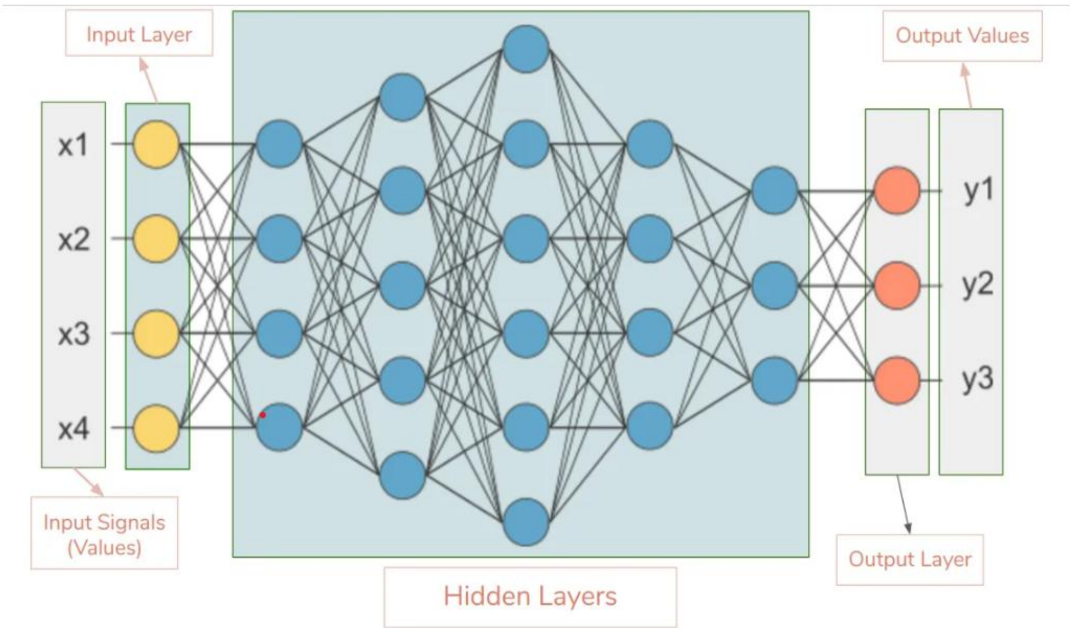


Figure III. 6 : Basic neural network layout

Input Layer: This is the first layer in the neural network. It takes input signals(values) and passes them to the next layer. It doesn't apply any operations on the input and has no weights and biases values associated. In our network we have 4 input signals x1, x2, x3, x4. **[56]**

Hidden Layer: Hidden layers have neurons(nodes) which apply different transformations to the input data. One hidden layer is a collection of neurons stacked vertically(Representation). In our image given below we have 5 hidden layers. All the neurons in a hidden layer are connected to each and every neuron in the next layer, hence we have a fully connected hidden layers.

Output Layer: This is the last layer in the network that receives input from the last hidden layer. With this layer we can get desired number of values and in a desired range. In this network we have 3 neurons in the output layer and it outputs y1, y2, y3.

III.4.2 Neural Network Process

As the network learns, weights and biases are adjusted to determine the strength of input signals. This process repeats until the network gets better at making accurate predictions. **[57]**

1. Initialization: Randomly initializing the weights and biases of the neural network as initial values to start the learning process.

2. Forward Propagation: The neural network takes the input data and processes it layer by layer. Each neuron calculates a value based on mathematical operations, including matrix multiplications and activation functions. This process, called **forward propagation**, generates the network's predicted output.

3. Loss Calculation: The model compares its predicted output to the actual target (ground truth) using a **loss function**, which measures how far the predictions are from the true values.

4. Backpropagation: The backpropagation algorithm is used to calculate the gradients of the loss function with respect to the network's weights and biases. It involves computing the derivative of the loss function with respect to each parameter, which indicates how the loss changes with respect to small changes in the parameters. The gradients calculated during backpropagation show the direction in which the loss increases the most. To minimize the loss, the network updates its parameters using optimizers.

5. Iteration: Steps 2 to 4 are repeated for each batch of training data multiple times (epochs) until the neural network's performance on the training data reaches a satisfactory level or converges to a solution.

III.4.3 Activation function

An **activation function** is a mathematical rule in neural networks that determines whether a neuron should be activated. It takes the weighted sum of a neuron's inputs and transforms it into an output, controlling how much signal moves to the next layer. Activation functions are essential because they allow neural networks to learn **non-linear relationships** in data. Choosing the right activation function is crucial, as it directly impacts the model's performance and accuracy.

The figure (**III.7**) illustrates the two types of activation functions: **[58]**

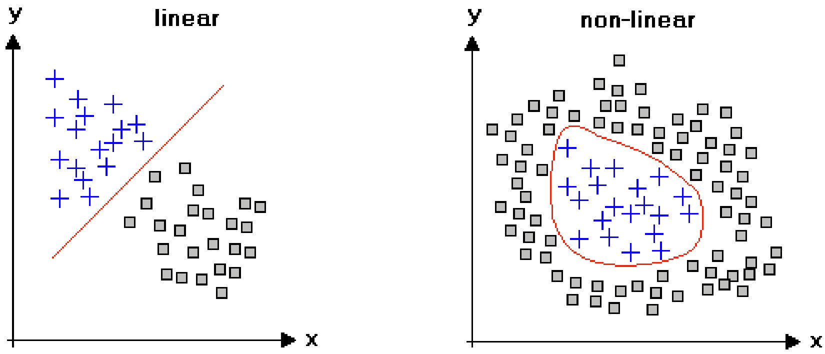


Figure III. 7 : Types of Activation functions

a) Linear Activation function: The simplest activation function that returns the input value unchanged, it ensures that the neural network can only recognize linear relationships in the data. This limits its performance immensely, as no more complex structures can be learned from the data this function is rarely used in deep neural networks, but only in simpler, linear models or in the output layer for regressions.

Linear activation function formula: f(x)=x

b) Non-Linear Activation Function: Introducing non-linear activation functions allows neural networks to model more complex relationships and learn non-linear decision boundaries.

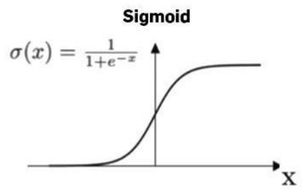
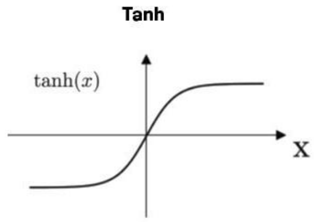
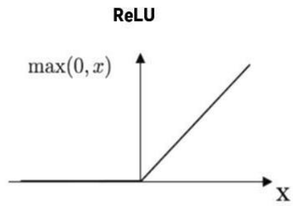
Activation Function	**Definition**	**Equation**	**Plot**
Sigmoid	Ensures that the input value is mapped to a range between 0 and 1,useful for objects recognition in images or classifying medical diagnosis	$f(x) = \frac{1}{1 + e^{-x}}$	
Hyperbolic tangent (Tanh)	Transforms the input value into the range between -1 and 1, used when negative inputs should be mapped to negative outputs	$f(x) = tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$	
Rectified Linear Unit (ReLU)	It keeps positive values and sets negative input values equal to zero in order to achieve good results and preventing vanishing gradients.	$f(x) = \max(x, 0)$	

Table III. 1 : Activation Functions

[59]

III.5 Deep learning

Deep learning (DL) is a branch of machine learning that uses multilayer neural networks, it emulates structure of the nervous system, where interconnected neurons transmit information by analyzing vast amounts of unstructured data, these models repeatedly perform tasks and get better, similar to how human improve through practice.

III.5.1 Types of Deep Learning Networks

a) Convolutional Neural Network (CNN)

A convolutional neural network (CNN) is a deep learning model that processes structured grid data such as images and videos. It automatically extracts features from input data to complete a specific task. CNNs are widely used for purposes such as video recognition (e.g., facial recognition), medical imaging (e.g., detecting cancerous tumors), self-driving cars (e.g., identifying road signs), and natural language processing (e.g., text classification).

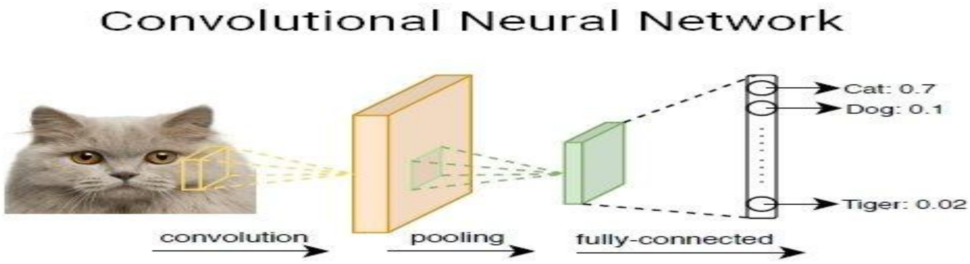


Figure III. 8 : Concept of Convolutional Neural Network (CNN)

The CNN architecture can be divided into three components. Here’s an overview of the three layers **[60].** As shown in (**Figure III.8**)

- **Convolutional Layers**

This is the first layer used to extract various features from the input images. In order to create a feature map, this layer applies a collection of filters (kernels) to the input image, each filter slides (convolves) over the image. This aids in identifying a variety of characteristics, including patterns, textures, and edges.

The general equation of the convolutional layer can be expressed as in the (1) **[61].**

$$Activation\ map = Input * Filter$$
$$= \sum_{y=0}^{columns}\left(\sum_{x=0}^{rows} Input(x-p, y-q)Filter\ (x,y)\right) \quad (1)$$

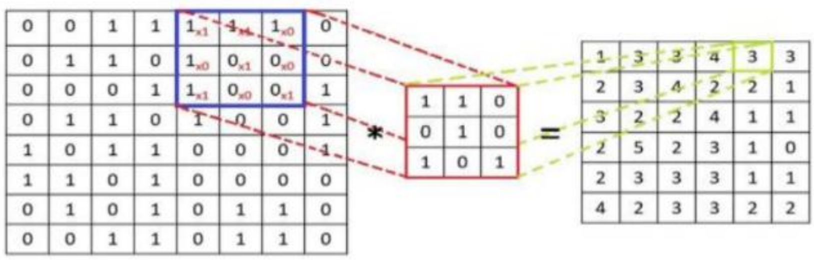


Figure III. 9 : Convolutional Layer

- **Pooling Layers**

This layer's main goal is to decrease the convolved feature map's size in order to reduce computing loads and the number of parameters. This is accomplished by working separately on each feature map and reducing the connections across layers. Pooling can result in a variety of forms: Max pooling and average pooling. **Fig (III.10)** represents a simple operation in dimension reduction of an activation map using the max-pooling function. **[61]**

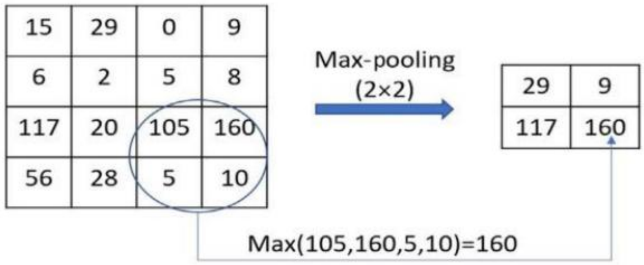


Figure III. 10 : Pooling Layer

- **Fully Connected Layers**

The fully connected layer, also known as the convolutional output layer, is the third layer in a CNN that receives input from the final convolutional or pooling layer, flattens it, and then passes it to the next layer. After multiple convolutional and pooling layers, the flattened output is fed into one or more fully connected (dense) layers, leading to the final output layer, which makes the classification or prediction.

In CNN layers, an activation function is applied to the filtered output, introducing non-linearity to the network. One commonly used activation function is the Rectified Linear Unit (ReLU) **[61]**.

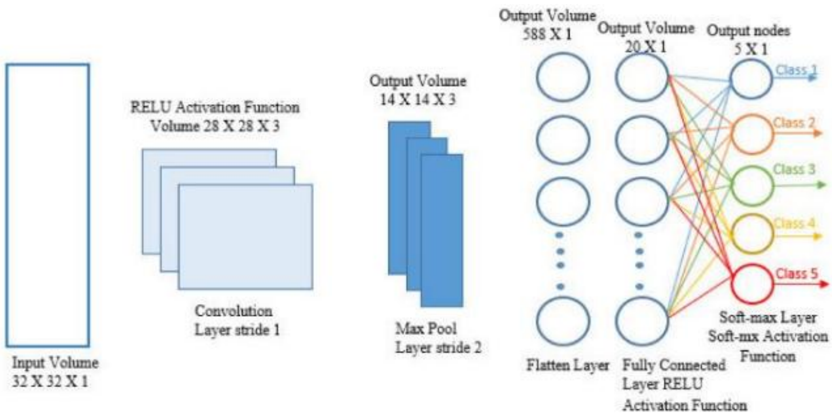


Figure III. 11 : Fully Connected layers

III.6 Transfer Learning

Transfer learning involves retraining a previously trained model (base model) on a new dataset from the current problem (target) domain. Depending on the similarity of the target domain and the domain where the base model is trained (usually called source domain), transfer learning can be feature extraction or fine-tuning. **[62]**

- **Feature extraction** is usually applied when the target domain dataset is scanty and similar to the source domain. This is achieved by replacing the last fully connected layer of the base model architecture with a new layer corresponding to the target output, initializing the other layers with the weights from the previous training scenario, and retraining only the newly added layer.

- **Fine-tuning** is applied either when the dataset is scanty or when the problem domains are different. This is achieved by replacing the last layer of the base model with a new layer corresponding to the target output, initializing the other layers with weights from the previous training scenario, and training the entire network again.

III.6.1 Pre-trained CNN models

a) VGG 19

is a deep convolutional neural network (CNN) developed by the Visual Geometry Group at the University of Oxford, representing an extension of the original VGG model. Known for its architectural complexity and strong feature learning capabilities, VGG19 captures rich visual representations through its depth but requires significant computational resources for training. It is widely used as a base model for transfer learning and fine-tuning on domain-specific datasets, and remains a benchmark in computer vision, especially for complex image recognition tasks **[63] .**

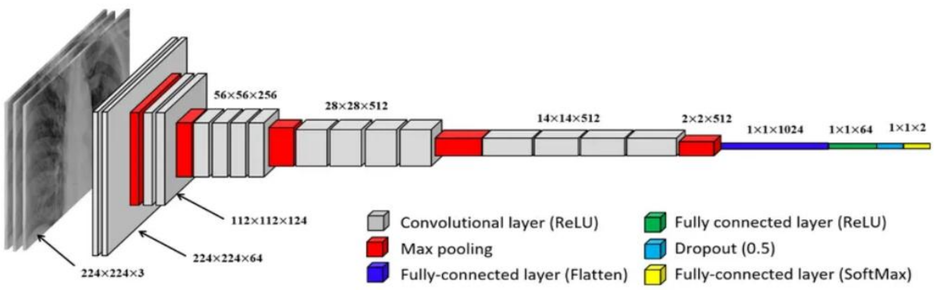


Figure III. 12 : The VGG19 Architecture

b) Google Net

also known as **Inception**, is a convolutional neural network (CNN) model developed by Google, known for its innovative architecture that emphasizes computational efficiency and multi-scale feature extraction. It features Inception modules that enable parallel processing, allowing the network to capture complex visual features efficiently. Google Net's design has significantly influenced the development of later deep learning models and remains a foundational architecture in various computer vision tasks. **[63]**

c) MobileNetV2

(Sandler et al., 2018) is a highly efficient and lightweight convolutional neural network architecture that improves upon MobileNetV1 by introducing inverted residual blocks and linear bottlenecks. These architectural innovations significantly reduce computational cost while maintaining strong performance.

MobileNetV2 achieves a balance between accuracy and efficiency, making it ideal for tasks such as image classification, object detection, and particularly semantic segmentation in resource-constrained environments. Its compact and powerful design

allows it to function effectively as a **feature extractor** in segmentation pipelines, enabling on-device semantic segmentation with low latency and high accuracy **[64]**

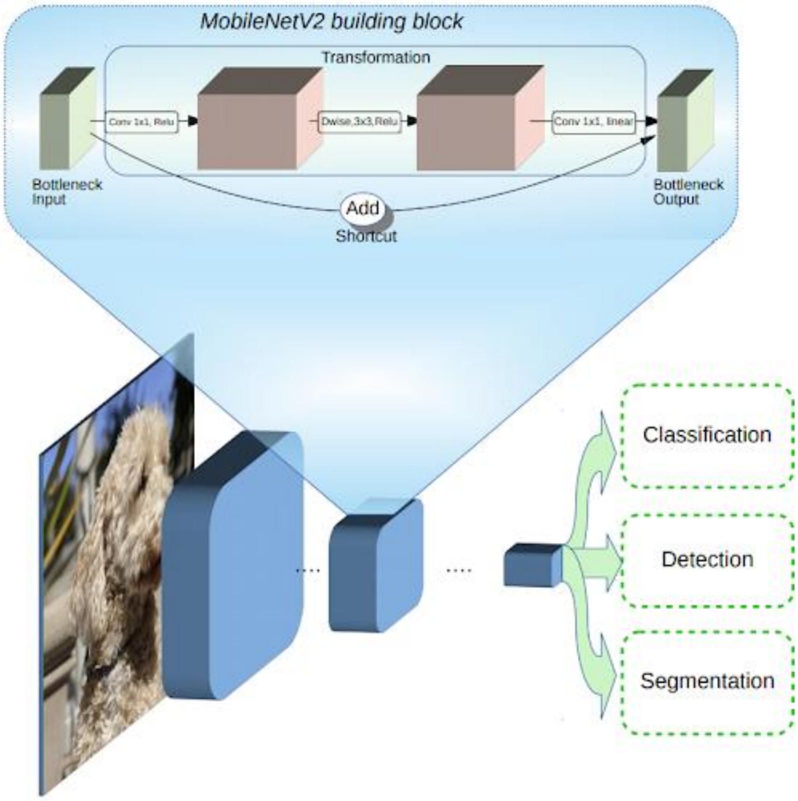


Figure III. 13 : Overview of MobileNetV2

III.6.2 Architectural components of MobileNetV2

a) Depthwise Separable Convolution

A depthwise separable convolution is an efficient alternative to standard convolution, consisting of two main components: **depthwise convolution** and **pointwise convolution. [65]** In depthwise convolution, each filter operates on a single input channel independently, unlike standard convolution where all channels are combined to produce each output feature map. This significantly reduces computational complexity. Following this, **pointwise convolution** —a 1×1 convolution—is applied to combine the outputs from the depthwise step and increase the number of channels. For example, instead of using 256 standard 5×5×3 filters, pointwise convolution uses 256 filters of size 1×1×3 to achieve the same output dimensionality with far fewer computations.

b) Linear Bottleneck

Linear bottlenecks use a linear activation instead of a non-linear one like ReLU. The idea behind bottlenecks is that the important information in the data can be represented in a smaller, low-dimensional space. So, reducing the size of a layer should be enough to capture this key information. However, this idea doesn't work well with ReLU activations, which are common in deep networks.

ReLU can remove too much information when the features are already in a low-dimensional space, making it harder for the network to learn.

c) Inverted Residuals

In the original residual block, the input passes through several bottleneck layers and then an expansion layer, with shortcuts connecting the "thick" layers that have many channels. MobileNetV2 changes this by placing shortcuts directly between the bottleneck layers, which are "thin" layers with fewer channels. This is based on the idea that the bottlenecks hold the important information, while the expansion layers mainly perform non-linear transformations. Some layers (shown as hatched) use linear activation. Also, MobileNetV2 uses ReLU6 as the non-linear activation function because it works better with low-precision computations. **[66]**

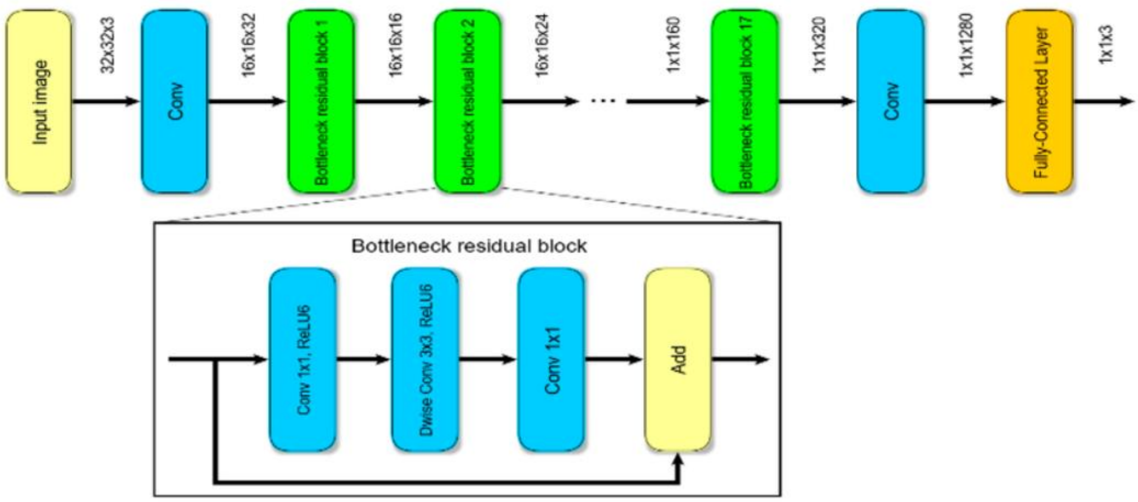


Figure III. 14 : The architecture of the MobileNetv2 network.

III.7 Segmentation techniques in deep learning

Image segmentation has gained significant attention in recent years, several innovative deep learning methods have been proposed and revolutionized the segmentation algorithms improving their efficiency with regard to object detection and recognition of important context in digital images such as landscapes, photos of people, medical images and much more. Image segmentation tasks can be classified into three groups based on the amount and type of information they convey. **[67]**

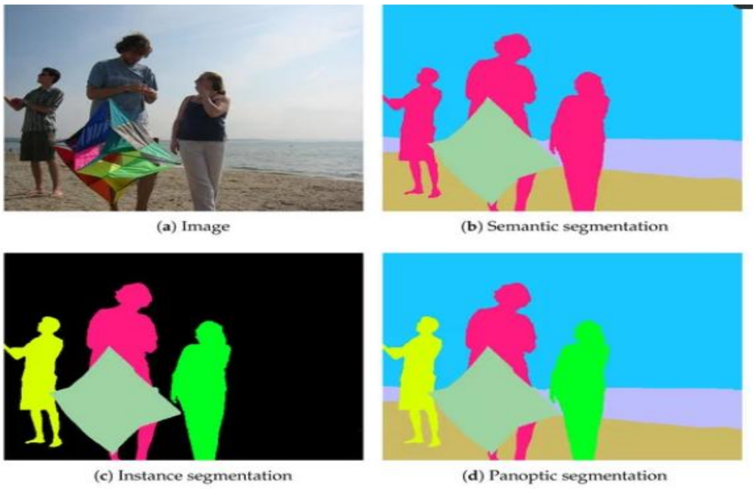


Figure III. 15 : Types of image segmentation based deep learning

III.7.1 Semantic Segmentation image

Semantic segmentation identifies collections of pixels and classifies them according to various characteristics, labeling every single pixel contained in an image by its semantic class allows for detailed image analysis. For example, if an image contains vehicles, semantic segmentation assigns the same label (purple) to all the pixels for all vehicles. Figure (**III.16**). This technique is widely used in fields such as autonomous driving**,** robotics**,** medical imaging**,** and transport facility planning and management**.**

Semantic segmentation models create a segmentation map of an input image. which is, essentially, a reconstruction of the original image, where each pixel has been color-coded by its semantic class to create segmentation masks. A segmentation mask is a portion of the image that has been differentiated from other regions of the image. For example, a segmentation map of a tree in an empty field would likely contain three segmentation masks: one for the tree, one for the ground and one for the sky in the background. **[68]**

Figure III. 16 : Example of Semantic segmentation

III.7.2 Semantic Segmentation models

a) Fully Convolutional networks (FCNs)

A significant advancement in semantic segmentation came from Long et al. (2015), who proposed the fully convolutional network (FCN). Since FCN don't require any fixed-size inputs, they can be applied to images of different dimensions and produce a pixel-wise classification map**.** This is achieved through two main components: **Downsampling (Encoding Path)** and **Upsampling (Decoding Path)**. **[69]**

- **Downsampling (Feature Extraction)**

  In the first half of the model, downsample reduces the spatial resolution of the image while developing complex feature mappings. With each convolution, finer details of the image are captured, leading to efficient discrimination between different classes; however, this process results in the loss of location information.

- **Upsampling (Segmentation Map Reconstruction)**

  To restore the lost spatial information, the downsampling process is followed by upsampling producer which takes multiple lower-resolution images as input and generate a high-resolution segmentation map as output.

b) Convolutional Encoder- Decoder Architecture

Encoder decoder architectures for semantic segmentation became popular with the onset of works like SegNet (by Badrinarayanan et. a.) in 2015 **[70].** As the name implies, the network consists of two main parts, namely encoder and decoder, each comprising a series of convolutional layers. Encoder-decoder models are classified into two categories: general segmentation and medical image segmentation. As shown in (**Figure III.17**)

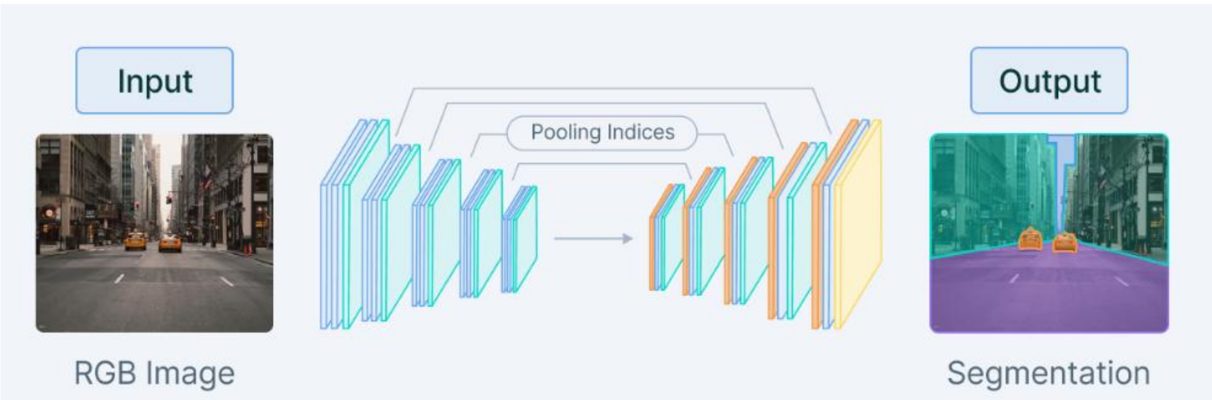


Figure III. 17 : Convolutional encoder – decoder

- **Encoder Network:**

The encoder network is responsible of transforming the input image into an encoded format while preserving the image's most crucial and pertinent details. Encoder network constitute of multiple convolutional blocks, and each block has a number of convolutional layers with a pooling layer to minimize the feature maps' spatial dimensions.

The output from each convolutional block is given to the next block after the pooling layer which reduces the dimensionality of the feature maps. The output from the last convolutional block is a compressed representation of the input image. often referred to as a feature map or an encoding vector. The encoding vector is a higher-level representation of the input image, retaining only the main features and discarding the rest. **[71]**

- **Decoder Network**

The decoder network's goal is to preserve the important properties of the compressed image while reconstructing the input image as precisely as possible. It is responsible for generating the input image from the encoded representation. The decoder network consists of multiple convolutional blocks, each containing several convolutional layers followed by an upsampling layer to restore the spatial dimensions of the feature maps. The upsampling layer, placed after each convolutional block's output, increases the resolution of the feature map. **[71]**

c) SegNet model

SegNet is a CNN architecture proposed by Badrinarayanan et al. (2017) for pixel-wise segmentation applications (semantic segmentation tasks)**,** based on an encoder-decoder architecture. It is designed to take an image as input and produce a pixel-wise label map as output (**Figure III.18**). The encoder captures high-level features by applying a series of convolutional and pooling layers, and a decoder uses the spatial pooling indices generated during the max-pooling in the encoder phase to upsample and produce a segmentation map. This is a form of upsampling that helps preserve the fine-grained details in the output while being memory-efficient **[72].**

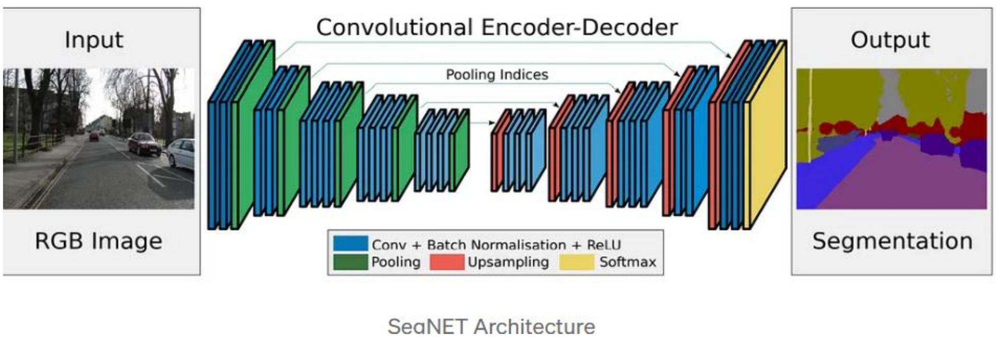


Figure III. 18 : SegNet architecture

d) DeepLabV3

DeepLabV3 architecture is introduced by Chen et al. (2018) for both semantic and instance segmentation tasks. Following encoder-decoder architecture where an encoder network progressively reduces the spatial dimensions of the feature maps until the lowest resolution, a middle network, where several convolutional layers are applied to the output of the encoder network, and a decoder network, where the spatial dimensions of the feature maps increase to the original resolution of the input image **[73].**

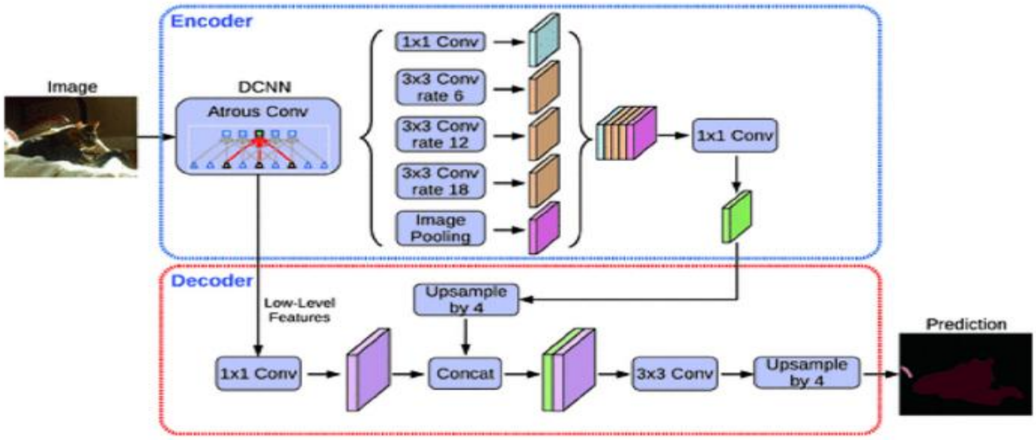


Figure III. 19 : DeepLabV3+ architecture

e) U-Net model

The U-Net is the convolutional neural network proposed by Ronneberger et al. (2015) that consistently achieves better results for biomedical image segmentation**.** It is an adaptation of the original FCN architecture and consists of two parts, an encoder, which stacks convolutional layers that consistently downsampling the image to extract information from it, while the decoder rebuilds the image features using the process of deconvolution. U-Net is two-dimensional network architecture whose structure is shown in Figure (**III.20**)

Skip-connections: An important innovation introduced to FCNs by U-Net is known as skip-connections, used link the output of one convolutional layer to a non-adjacent layer. This helps reduce data loss during downsampling and improves output resolution. Each convolutional layer is upsampled and merged with features from other layers until the final output accurately represents the image **[73] .**

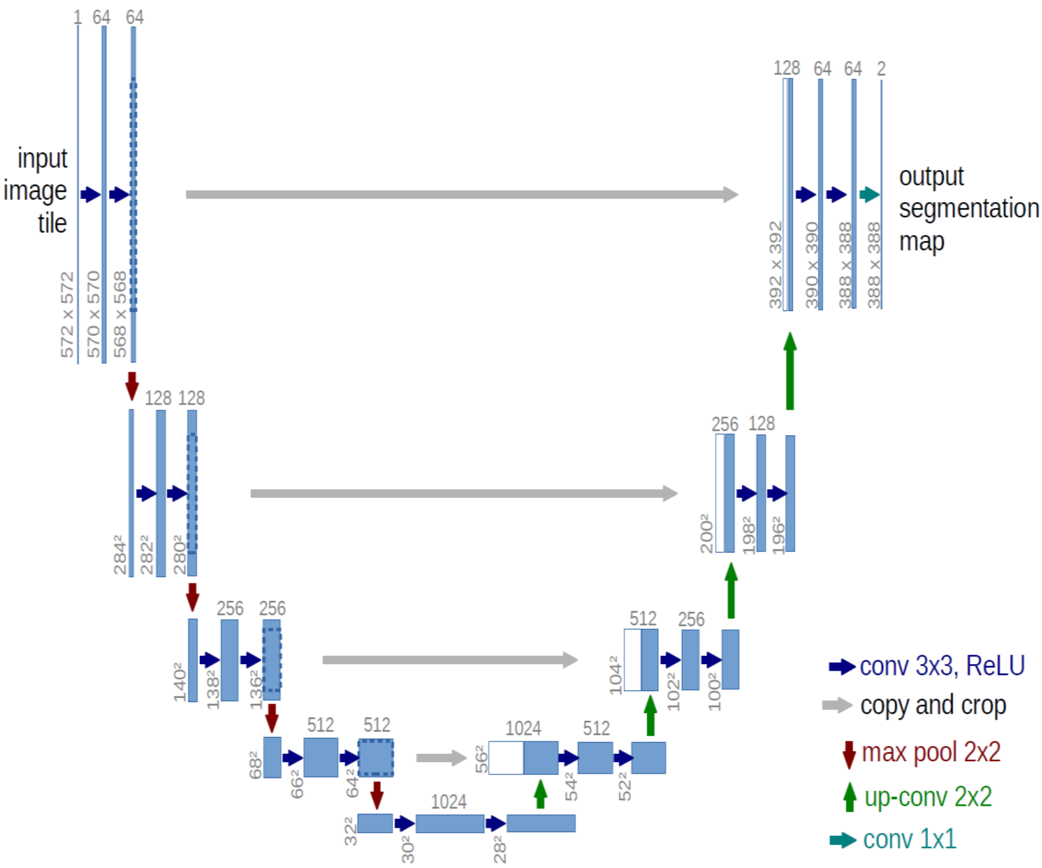


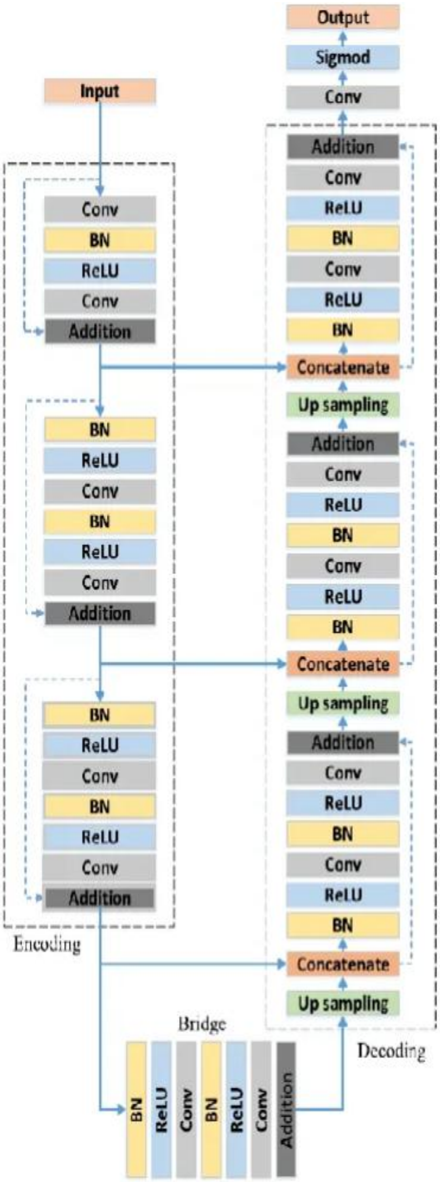
Figure III. 20 : U-Net architecture

f) Deep residual U-Net

ResUNet is an improved version of the original U-Net that combines the strengths of **U-Net** and residual networks (ResNets). It's mainly used for **semantic segmentation**, where the goal is to label each pixel of an image.

In ResUNet, instead of using regular layers, we use residual blocks**,** which are made of stacked layers that include **convolutions, ReLU activation, and batch normalization.** The key difference is that these blocks have **skip connections** that let information pass through directly. This helps the network **learn faster and more efficiently**, because it reduces the chances of losing important details during training, the structure is shown in figure (**III.21**). **[74]**

Each part of the ResUNet (encoder, bottleneck, and decoder) uses these residual units. A typical residual unit includes: an **identity shortcut** (skip connection) and two **3×3 convolutional blocks**, each followed by **batch normalization** and **ReLU activation. [75]**



	Unit level	Conv layer	Filter	Stride	Output size
Input					$224 \times 224 \times 3$
Encoding	Level 1	Conv 1	$3 \times 3/64$	1	$224 \times 224 \times 64$
		Conv 2	$3 \times 3/64$	1	$224 \times 224 \times 64$
	Level 2	Conv 3	$3 \times 3/128$	2	$112 \times 112 \times 128$
		Conv 4	$3 \times 3/128$	1	$112 \times 112 \times 128$
	Level 3	Conv 5	$3 \times 3/256$	2	$56 \times 56 \times 256$
		Conv 6	$3 \times 3/256$	1	$56 \times 56 \times 256$
Bridge	Level 4	Conv 7	$3 \times 3/512$	2	$28 \times 28 \times 512$
		Conv 8	$3 \times 3/512$	1	$28 \times 28 \times 512$
Decoding	Level 5	Conv 9	$3 \times 3/256$	1	$56 \times 56 \times 256$
		Conv 10	$3 \times 3/256$	1	$56 \times 56 \times 256$
	Level 6	Conv 11	$3 \times 3/128$	1	$112 \times 112 \times 128$
		Conv 12	$3 \times 3/128$	1	$112 \times 112 \times 128$
	Level 7	Conv 13	$3 \times 3/64$	1	$224 \times 224 \times 64$
		Conv 14	$3 \times 3/64$	1	$224 \times 224 \times 64$
Output		Conv 15	1×1	1	$224 \times 224 \times 1$

Figure III. 21 : ResUNet architecture

III.7.3 Instance Segmentation image

Instance segmentation systems detect objects in an image with greater detail by generating a pixel-by-pixel “segmentation mask” of the precise shape and area of each instance. Unlike semantic segmentation models, which make no distinction between things-classes of countable entities with distinct shapes but the instance segmentation models focus exclusively on detecting and generating segmentation masks for individual things. An instance segmentation model must be able to delineate each different object instance, even for occluded instances of the same class of object **[76].**

Mask R-CNN: Is deep learning model for instance segmentation and object detection, developed by **Facebook AI Research** in 2017, Mask R-CNN builds on **Faster R-CNN** by adding an additional branch for predicting object masks alongside the existing bounding box detection branch (He et al., 2017). It can precisely identify and outline each object in an image, making it highly effective for complex image analysis tasks. The architecture consists of three main components: **[77]**

- A **backbone network**, typically a **CNN**, that extracts features from input images and is shared by both branches.
- **Region Proposal Network (RPN)** that generates region proposals based on the feature maps from the backbone.
- Two **parallel branches**: one for **bounding box detection** (predicting class labels and box coordinates) and another for **mask prediction** (generating a binary mask for each object within the bounding box).

III.7.4 Panoptic segmentation image

Panoptic segmentation merges the capabilities of **semantic** and **instance segmentation** by assigning both a **semantic label** and an **instance ID** to each pixel in an image. It labels each pixel as either a **"thing"** (countable objects like cars, people, or animals) or **"stuff"** (amorphous regions such as grass, sky, or road). This approach provides a complete understanding of the visual scene, allowing systems to interpret the semantics of different regions while also differentiating between multiple instances of the same class. **[78]**

a) **EfficientPS:** Efficient Panoptic Segmentation was introduced by researchers to overcome the limitations of older CNN approaches. This new approach combines both semantic and instance segmentation into a single powerful network. Technically we can say Efficient PS is an end-to-end network architecture that performs both semantic and instance segmentation simultaneously. This advanced panoptic segmentation technique performs its operations in two stages: **[78]**

- **Stage 1:** EfficientPS starts its operation using a backbone network. This backbone network of EfficientPS extracts meaningful features from the input image and sends it to the panoptic segmentation head for final segmentation. Some of the popular backbone networks used in this stage are ResNet, EfficientNet, and ResNeXt backbones.
- **Stage 2:** The meaningful features extracted from the EfficientPS backbone network are fed into another architecture called Panoptic Segmentation Head. This head uses the information from the backbone to perform two tasks at once: recognize objects (**instance segmentation**) and label background areas (**semantic segmentation**) to yield a combined final output.

III.7.5 Comparison of segmentation Models

Methods	Positive Points	Positive Points
U-Net	Focuses on relevant features. Effective in complex medical imaging tasks.	More complex architecture Increased computational and memory requirements.
Mask R-CNN	Combines detection and segmentation. High accuracy for instance segmentation. Handles occlusions well.	Computationally expensive. Requires extensive training time and resources.
Fully Convolutional Networks (FCN)	Improved gradient flow and feature reuse. High performance on various segmentation tasks.	May produce coarse segmentation boundaries. Requires large, annotated datasets.
DeepLabV3+	Handles multi-scale context. Refines segmentation boundaries. High performance on challenging datasets.	Computationally intensive. Requires significant computational resources for training and inference.
SegNet	Efficient memory usage. Suitable for real-time applications.	May struggle with fine details in segmentation. Requires large amounts of labeled data for training.
DenseNet	Improved gradient flow and feature reuse. High performance on various segmentation tasks.	High memory usage. May be prone to overfitting if not properly regularized.

Table III. 2 : Summary of learning-based image segmentation methods **[79]**

III.8 Conclusion

This chapter provides a comprehensive overview of the fundamental concepts of Artificial Intelligence (AI), Machine Learning (ML), and Neural Networks, leading to a deeper understanding of Deep Learning and its role in computer vision. We also explored various segmentation techniques based on deep learning, including semantic, instance, and panoptic segmentation, along with the architectures that power them. Additionally, the concept of transfer learning was introduced, highlighting how pretrained models can be leveraged to improve performance and efficiency, especially in resource-constrained environments.

As AI and deep learning continue to evolve, segmentation models are becoming more precise and efficient, driving advancements in autonomous driving, industrial automation, and medical imaging. In the following chapters, we will focus on medical imaging, exploring its significance and applications in Breast Cancer.

Chapter IV : Experiments and Realization

IV.1 Introduction

This chapter outlines the experimental setup and implementation of deep learning model for breast cancer segmentation in mammographic images. It describes the dataset used, preprocessing techniques, network architectures of our proposed model, comparison between existing and our model results, training strategies, and evaluation metrics such as Dice coefficient and IoU. As part of this work, a mobile application was developed to facilitate real-time model testing and visualization, demonstrating the practical applicability of the proposed system in clinical scenarios.

IV.2 Tools and Libraries

IV.2.1 Google Colaboratory

Google Colab is a cloud-based Jupyter notebook environment that requires no setup and provides free access to computing resources, including GPUs and TPUs. It is particularly well-suited for machine learning, data science, and educational purposes. Users can write and execute Python code directly in the browser, making it convenient for collaborative research and prototyping. Colab supports integration with Google Drive, allowing easy storage and access to datasets, models, and results. Due to its support for popular machine learning libraries like TensorFlow, Keras, and PyTorch, Colab is widely used in both academic and industrial research for tasks such as image segmentation. **[80]**

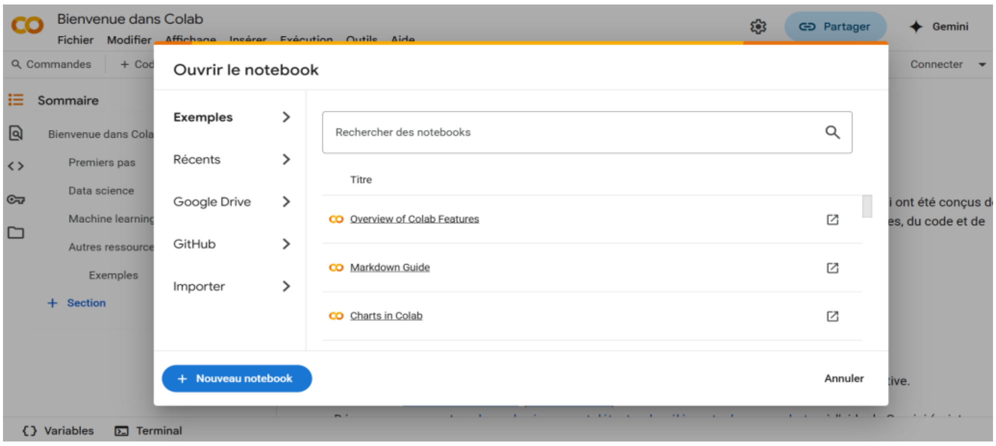


Figure IV. 1 : Google Colaboratory

IV.2.2 Tensor Flow

TensorFlow is an open-source machine learning framework developed by Google, designed for building and training neural networks and other ML models. It provides flexible tools for large-scale numerical computation, supporting CPUs, GPUs, and TPUs for accelerated performance. Widely used in deep learning, TensorFlow enables applications like computer vision, natural language processing, and recommendation systems. Its high-level APIs (like Keras) simplify model development while maintaining scalability. **[81]**

IV.2.3 Python

Python is a powerful and easy-to-learn programming language, offering efficient high-level data structures and a simple approach to object-oriented programming. Thanks to its clear syntax, dynamic typing, and direct interpretation, it is perfectly suited for scripting and rapid application development across various platforms. **[82]**

IV.2.4 Keras

Keras is a high-level neural networks API, written in Python and designed for fast experimentation with deep learning. Originally developed independently, it is now an official part of TensorFlow (as tf.keras), providing a user-friendly interface for building and training models. Keras simplifies complex tasks like defining layers, optimizers, and loss functions while maintaining compatibility with TensorFlow's backend. It is widely used for prototyping due to its modularity and ease of use. **[83]**

IV.2.5 OpenCV

OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. **[84]**

IV.2.6 Flutter

Flutter is an open-source, cross-platform tool created by Google. It is used to develop interactive, reactive, and adaptive frontend applications (desktop, web, and mobile) that can run on several operating systems using one codebase.

IV.2.7 Flask Api

A Flask API is a web application built using the Flask framework, which allows you to create endpoints that can be accessed over HTTP for various operations such as retrieving data, submitting data, and performing other interactions. Flask is a lightweight and flexible web framework for Python that makes it easy to create web applications, including APIs. **[85]**

IV.3 Dataset

The experiments were conducted using the Digital Mammography Dataset for Breast Cancer Diagnosis Research (**DMID**) **[86**], introduced in India in 2023. It consists of 510 mammographic images, each paired with a Region of Interest (ROI) mask highlighting abnormal areas such as breast masses. The dataset was specifically designed to support segmentation research, providing high-quality, pixel-level annotations that are often limited in existing mammography databases. These detailed annotations are essential for training and evaluating deep learning models aimed at accurate tumor boundary detection, which plays a critical role in breast cancer diagnosis. **[87]** The data is divided into 80% training ,10% validation and 10% for testing, we applied data augmentation techniques on training sets to increases the diversity of the training samples without altering the underlying patterns, helping the model to become more robust and effective when applied to new unseen mammographic images.

IV. 3.1 Preprocessing and data augmentation

Image preprocessing is a crucial step in any medical image segmentation especially when working with mammograms to improve image quality, reduce noise, standardize inputs. The original dataset consists of grayscale mammographic images along with their corresponding binary masks. The following steps were performed :

a) Image Resizing and Format conversion

All images and masks were resized to a uniform resolution of **256×256 pixels**. This resizing standardizes input dimensions for the neural network and reduces computational load. Additionally :

- **Images** were converted from single-channel grayscale to **3-channel RGB**, as the pre-trained backbone (MobileNetV2) expects 3-channel input.
- **Masks** were binarized using a threshold of 127 to ensure clear foreground-background separation. Pixels >127 were set to 255 (foreground indicating tumor region), and others to 0 (background). Masks were not normalized, as they represent class labels rather than continuous intensity values.

b) Normalization

Input mammographic images were normalized to the [0, 1] intensity range to improve numerical stability during training. Segmentation masks were kept in their original binary format (0 for background, 255 for foreground) and were not normalized to preserve label integrity.

c) Data Augmentation

To enhance model generalization and reduce overfitting, each image in the training set was augmented three times, generating three synthetic variants per original sample. Augmentation techniques from the Albumentations library were consistently applied to both the images and their corresponding masks to preserve spatial alignment. These augmentations include horizontal flipping, rotation, and shift-scale-rotate, which help the model generalize to various orientations and sizes of lesions, as well as contrast enhancement techniques like CLAHE to improve the visibility of features such as masses.

The results obtained after applying these augmentation techniques are illustrated in (**Figure IV.2**) The following transformations were used:

- **Shift, Scale, Rotate (p=0.5):** Applies random affine transformations within small limits to simulate different spatial perspectives.
- **Random Brightness & Contrast (p=0.3):** Slightly alters brightness and contrast to mimic varying imaging conditions.
- **Gaussian Blur (p=0.05):** Adds mild blur to simulate low-resolution or noisy scans, improving robustness to noise.
- **Horizontal Flip (p=0.5):** Flips the image and its mask horizontally with a 50% probability.
- **CLAHE (p=0.2):** Applies contrast-limited adaptive histogram equalization to enhance local contrast and highlight features in dense tissue.
- **Random 90° Rotation (p=0.5):** Rotates the image randomly by 90°, 180°, or 270°, adding orientation variability.

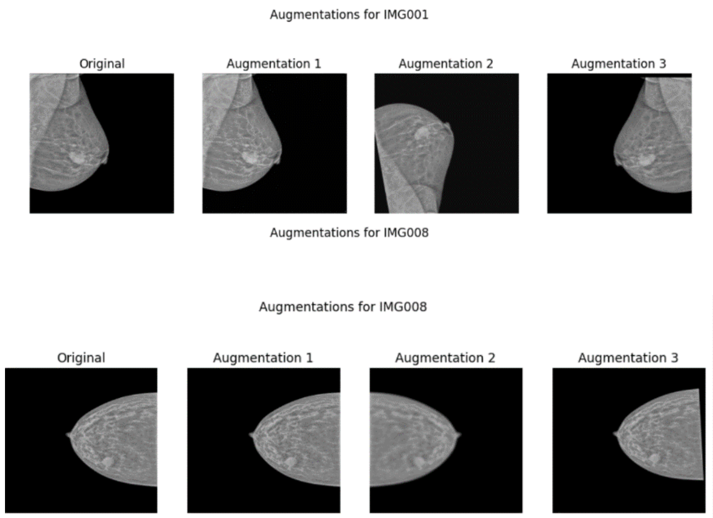


Figure IV. 2 : preprocessing and data augmentation application resukts

IV.4 Proposed Model

The proposed Model architecture combines a lightweight encoder '**MobileNetV2**' with residual decoding blocks '**ResUnet**', Skip connections between encoder and decoder at multiple resolutions, enabling the recovery of spatial details. The bridge layer captures abstract semantic features, while residual blocks enhance feature refinement during upsampling. This design balances efficiency and accuracy, making it well-suited for the segmentation of breast masses in mammograms. A global overview of the proposed architecture is illustrated below in the **(Figure IV.3)** to provide a better understanding.

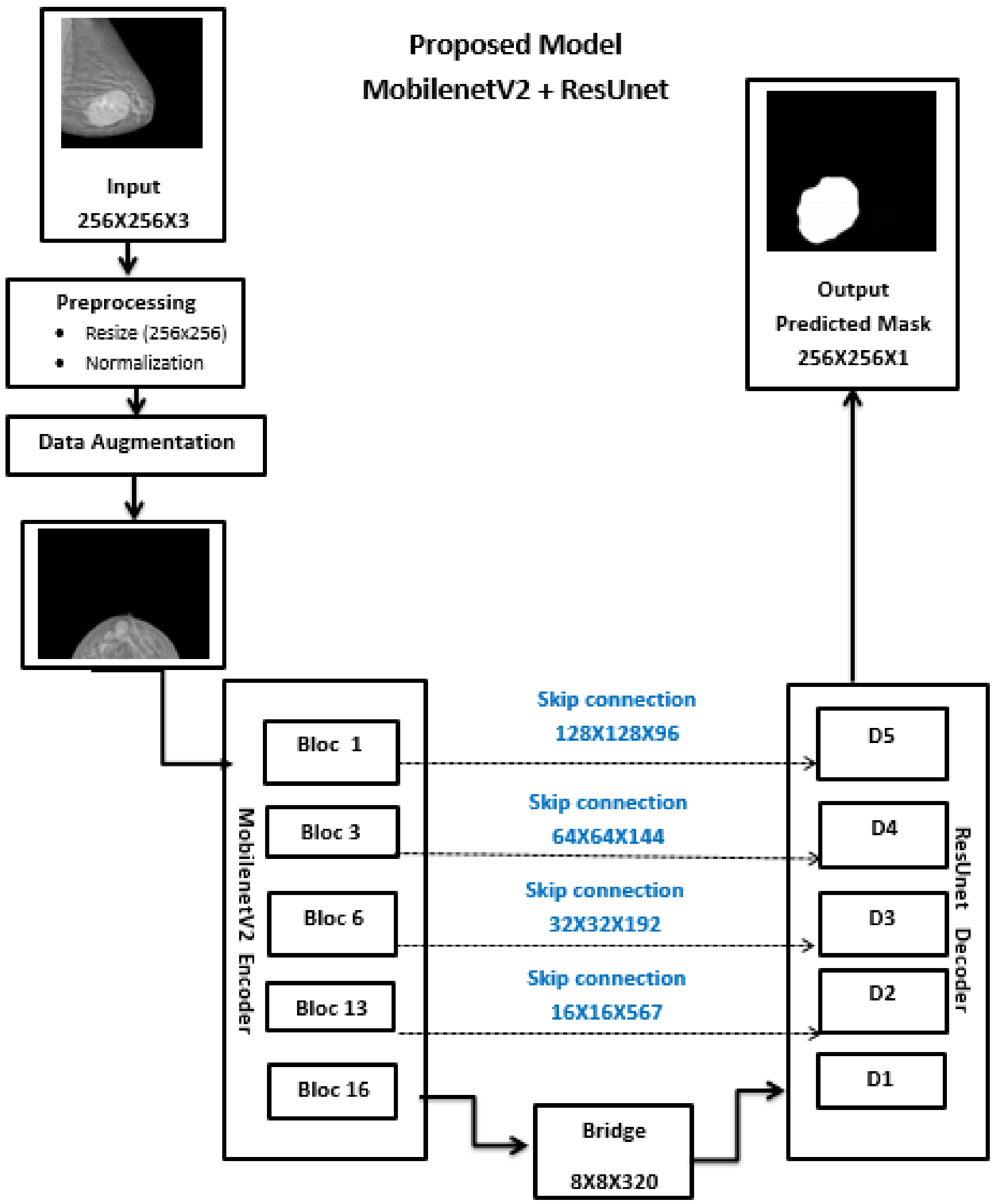


Figure IV. 3 : Proposed model Architecture

- ## Encoder

The encoder utilizes MobileNetV2 a lightweight convolutional neural network initialized with ImageNet weights to extract hierarchical features from input images of size 256×256×3. The top classification layers of MobileNetV2 are excluded (**include_top=False**), preserving only convolutional layers for feature extraction while progressively reducing its spatial resolution through depthwise convolutions and bottleneck layers.

- **Skip Connections**

To recover spatial information lost during downsampling, skip connections are integrated between intermediate layers of the MobileNetV2 encoder and the decoder. These connections transfer multi-scale feature maps at resolutions of 128×128, 64×64, 32×32, and 16×16.These are concatenated with decoder layers at matching resolutions to recover fine details which were lost during downsample like edges, shapes and object boundaries.

- **Bottleneck**

The bottleneck (bridge) of the network is defined by the deepest encoder output layer The bottleneck layer, operating at a resolution of 8×8×320, captures the most abstract and high-level semantic features such as the presence of a mass or tissue texture before the upsampling process begins. This deep representation enables the decoder to reconstruct fine-grained segmentation maps by combining both semantic context and spatial precision.

- **Residual Decoder**

The **decoder** is the part of the model that is responsible for reconstructing the segmented image from the compressed features produced by the encoder (bridge) and gradually reconstructs them back into a full-resolution segmentation map. To make this process more effective, the decoder uses residual blocks after each upsampling stage.

- **Output Layer**

The final decoder layer upsamples to the original image size, followed by a 1×1 convolutional layer that reduces the channels to **num_classes** (**1 for binary segmentation**) with a sigmoid activation function to output a probability map for binary segmentation of masses values between (0 and 1).

IV. 5 Results and discussion

This section presents the experimental results obtained from training and evaluating different segmentation models including our proposed model on the mammography dataset. It is divided into two main parts. The first part reports and analyzes the performance of our implemented architectures, including **ResUNet, MobileNetV2-UNet, and MobileNetV2-ResUNet**. The second part provides a comparative evaluation of our best-performing model against existing methods reported in the literature using the same dataset.

IV. 5. 1 Performance metrics and loss function

The performance of the models was evaluated using the following metrics which are widely used in medical image segmentation:

- **Dice Similarity Coefficient (DSC)**

The Dice coefficient is one of the most widely used evaluation metrics for semantic segmentation. It measures the similarity between the predicted segmentation masks and the manually annotated ground truth masks. The Dice coefficient is calculated as twice the intersection of the predicted and ground truth masks, divided by the total number of pixels in both masks. The formula of the Dice coefficient is as follows **[88]:**

$$DICE = \frac{2|P \cap G|}{|P| + |G|},$$

P = predicted mask

G = ground truth mask

P∩G = intersection between them

- **Intersection over union (IoU)**

Intersection over Union (IoU) is a widely used evaluation metric for image segmentation models. It measures the overlap between the predicted segmentation mask and the ground truth mask. IoU is an important metric for evaluating segmentation models because it measures how well the model can separate objects from their background in an image. The formula of the Dice coefficient is as follows **[89]**:

$$\mathrm{IoU} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}$$

True Positive (TP): Pixels correctly predicted as part of the object.

False Positive (FP): Pixels incorrectly predicted as part of the object (but belong to the background).

False Negative (FN): Pixels that are part of the object but were missed by the prediction.

- **Tversky loss function**

The Tversky loss is a loss function used in image segmentation, particularly in medical imaging. It measures the dissimilarity between predicted and ground truth binary masks, allowing better handling of class imbalance. It generalizes the Dice and Jaccard (IoU) indices by allowing control over the penalties for false positives and false negatives**. [90]**

The Tversky loss is defined by the following formula:

$$TI = \frac{TP}{TP + \alpha FN + \beta FP}$$

TP: Intersection between predicted and ground truth masks

FN: False negative pixels (pixels incorrectly classified as negative in the prediction).

FP: False positive pixels (pixels incorrectly classified as positive in the prediction).

α , β: weights parameters that control the importance of FP(α) and FN(β)

IV. 5.2 Results obtained for proposed model

All three architectures were trained and tested under identical experimental conditions to ensure fair comparison.

a) **Two-Phase Training Strategy**: The training process employed a systematic two-phase approach.

- **Encoder Freezing:** The initial phase involved training for 20 epochs with frozen weights of the pretrained MobileNetV2 encoder to preserve the pre-trained features, and avoid overfitting. Only the decoder and residual blocks are trained.
- **Fine-Tuning:** In the second phase, all layers including the encoder are unfrozen and trained with a reduced learning rate to fine-tune the entire model, allowing adaptation to domain-specific features of mammography images.

b) **Optimizer and Learning Rate**

The Adam optimizer was consistently used across both training phases. The initial training phase utilized a learning rate of 1×10^{-4}, which was reduced to 1×10^{-5} during the fine-tuning phase to enable more refined parameter updates.

c) **Loss Function and Metrics**

The Tversky loss function served as the primary optimization criterion throughout training, specifically designed for handling class imbalance in segmentation tasks. Model performance was evaluated using Dice coefficient and Intersection over Union (IoU) coefficient as key metrics.

d) **Training Callbacks**: Three essential callbacks regulated the training process:

- **ModelCheckpoint** automatically saved the best-performing model based on validation Dice coefficient.
- **EarlyStopping** monitored validation Dice coefficient with a patience of 10 epochs to prevent overfitting while restoring the best weights.
- **ReduceLROnPlateau** dynamically reduced the learning rate by a factor of 0.5 when validation loss plateaued for 5 consecutive epochs, with a minimum threshold of 1×10^{-6}.

IV. 5. 3 Dice Coeff curves of our Model

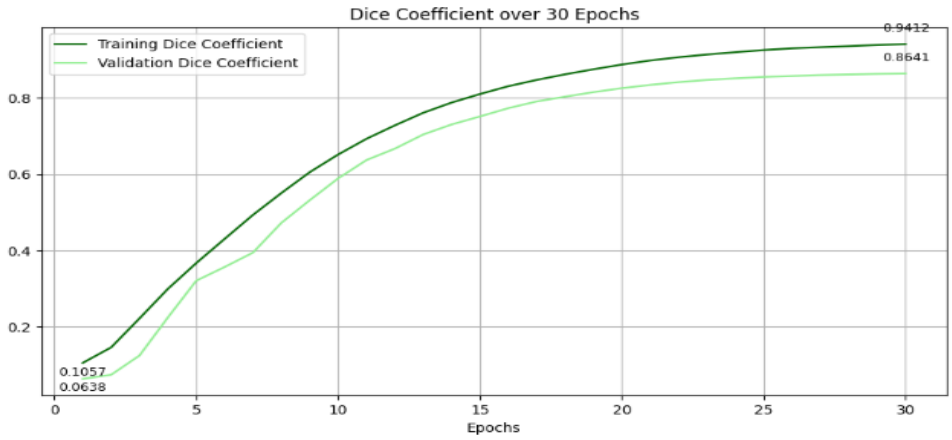


Figure IV. 4 : Trainig and validation DSC

- **Detailed Epoch Analysis:**

Dice coefficient measures overlap between predicted masks and ground truth ()

Training Dice rises from ~0.1057 to **0.9412**, and validation Dice improves from ~0.0638 to **0. 8641.**This is a strong indicator of the model's capability to segment regions accurately.

IV. 5. 4 IOU curves of our Model

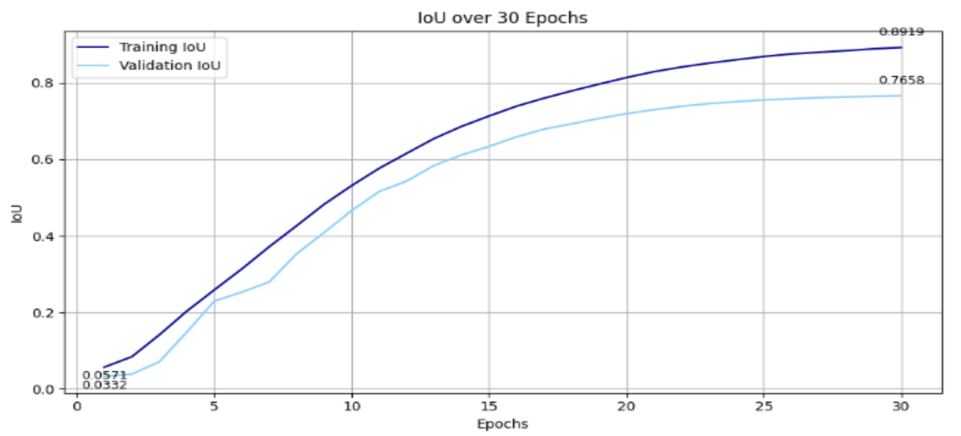


Figure IV. 5 : Trainig and validation IOU

- **Detailed Epoch Analysis:**

IoU measures the ratio of the intersection to the union of predicted and actual masks.

Training IoU improves from ~0.0571 to **0.8919**, while validation IoU reaches **0. 7658.This** confirms that the model learns to localize the masses well, with high segmentation accuracy.

IV.5. 5 Loss curves of our Model

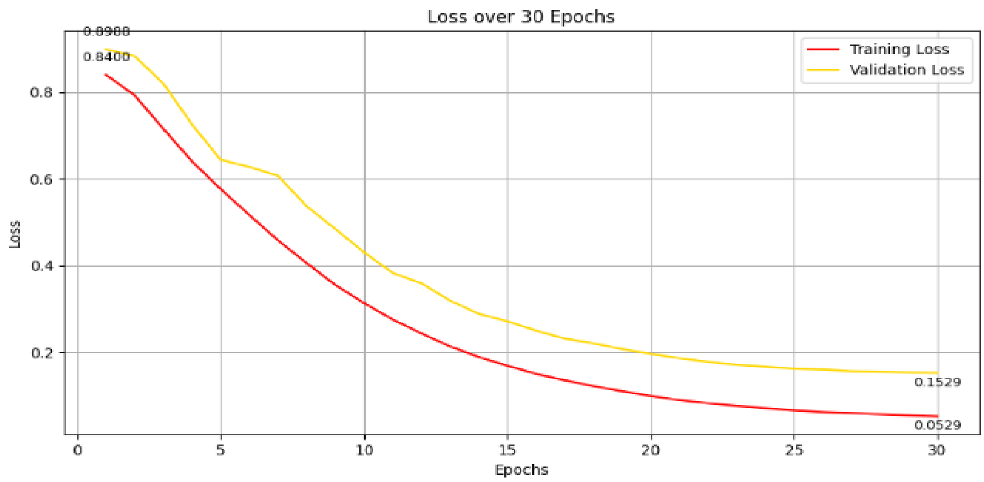


Figure IV. 6 : Trainig and validation DSC

- **Detailed Epoch Analysis :**

Both losses decrease consistently over time, which indicates that the model is learning effectively.

The **training loss** drops from ~0.84 to **0.0529**, while **validation loss** drops from ~0.90 to **0.1529**.A small gap between the two curves at the end suggests **good generalization**, with no overfitting.

IV. 5.6 Comparison of proposed model performance

The results are summarized in the following table (**Table IV.1**). The proposed **MobileNetV2-ResUNet** model outperformed the other two models in both loss reduction and segmentation accuracy. The integration of a lightweight encoder (MobileNetV2) with a residual decoder helped achieve better generalization and boundary precision in mass segmentation.

Model	Loss	Dice Coefficient	IOU
ResUNet	0.2431	0.7667	0.5856
MobileNetV2-UNet	0.3650	0.5823	0.3805
MobileNetV2-ResUNet	**0.1439**	**0.8704**	**0.7741**

Table IV. 1 : Summary of model performance comparison

To further assess the performance of the proposed MobileNetV2-ResUNet model, we compare it with existing models previously evaluated on the same dataset (DMID).

The authors of the referenced article tested U-Net and Attention U-Net architectures and reported their segmentation performance using Dice Similarity Coefficient (DSC) **[87].** In contrast, our proposed approach outperformed these models significantly, achieving a Dice coefficient of **0.8704**, which indicates improved segmentation accuracy, particularly in detecting and delineating mass regions. The comparison highlights the effectiveness of integrating a lightweight encoder (**MobileNetV2**) with a residual decoder.

Model	Dice Coefficient (Val)
UNet	0. 6076
Attention-UNet	0. 6400
MobileNetV2-ResUNet	**0.8704**

Table IV. 2 : Our Proposed Approaches Vs State of the Art

IV. 6 Visualization of the results

To qualitatively evaluate the performance of the proposed model, a set of sample predictions was visualized (**Figure IV. 7**.) This involved displaying the original mammography images alongside their corresponding ground truth masks and the predicted segmentation masks produced by the model. The results demonstrate that the predicted masks closely match the annotated ground truths, confirming the effectiveness of the proposed MobileNetV2-ResUNet architecture in capturing the relevant regions of interest.

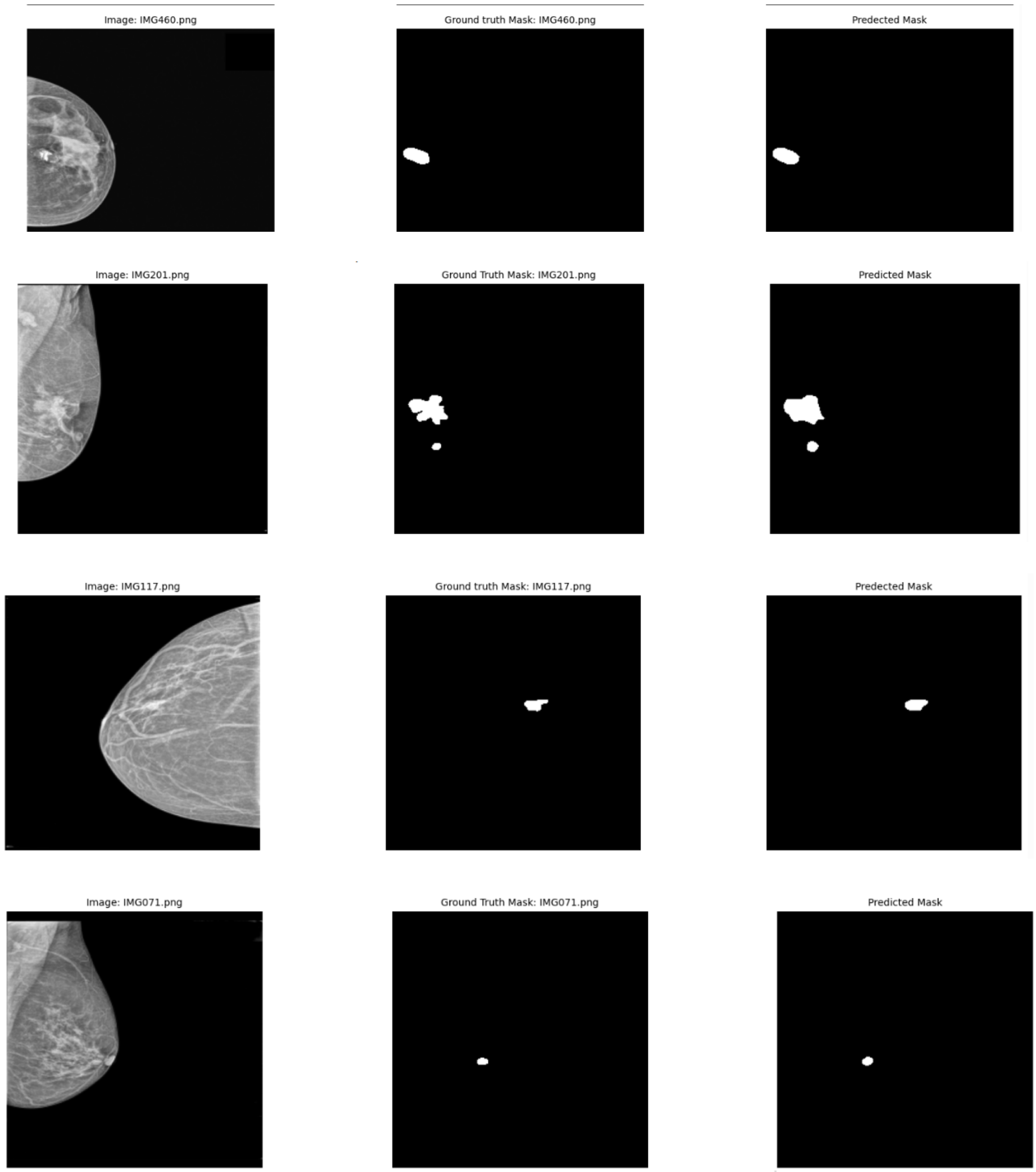


Figure IV. 7 : Segmentation results of breast abnormalities

IV.7 Mobile Application Development Testing

This mobile application enables automatic breast mass segmentation from mammographic images using a deep learning model deployed via a Flask API. It is built using **Flutter** for the front-end and **Flask** as a lightweight server-side framework hosting a trained neural network model.

IV.7.1 System Architecture

The mobile segmentation pipeline involves the following components:

- **User Interface (Flutter App)**: A cross-platform mobile app for selecting and uploading mammogram images.
- **Flask API Server**: Hosts our deep learning segmentation model and processes incoming image requests.
- **Segmentation Model**: A trained model that segments suspicious masses (potential tumors) from mammographic images.

IV.7.2 Workflow Description

A. Flask API and Segmentation (Server Side)

1. **Preprocessing**
   - Converts the uploaded image to RGB.
   - Resizes it to the expected input size for the model (256×256).
   - Normalizes pixel values to [0, 1].
2. **Model Inference**
   - A pre-trained MobileNetV2-based U-Net model predicts a segmentation mask.
   - The output mask is thresholded (values $> 0.5 \rightarrow 1$, else 0) to create a binary map of the suspected mass.
3. **Post-processing**
   - A red overlay is generated where the mask is active.
   - The red mask is blended over the original image using alpha compositing (transparency factor 0.4).
   - The final image is encoded and sent back to the client

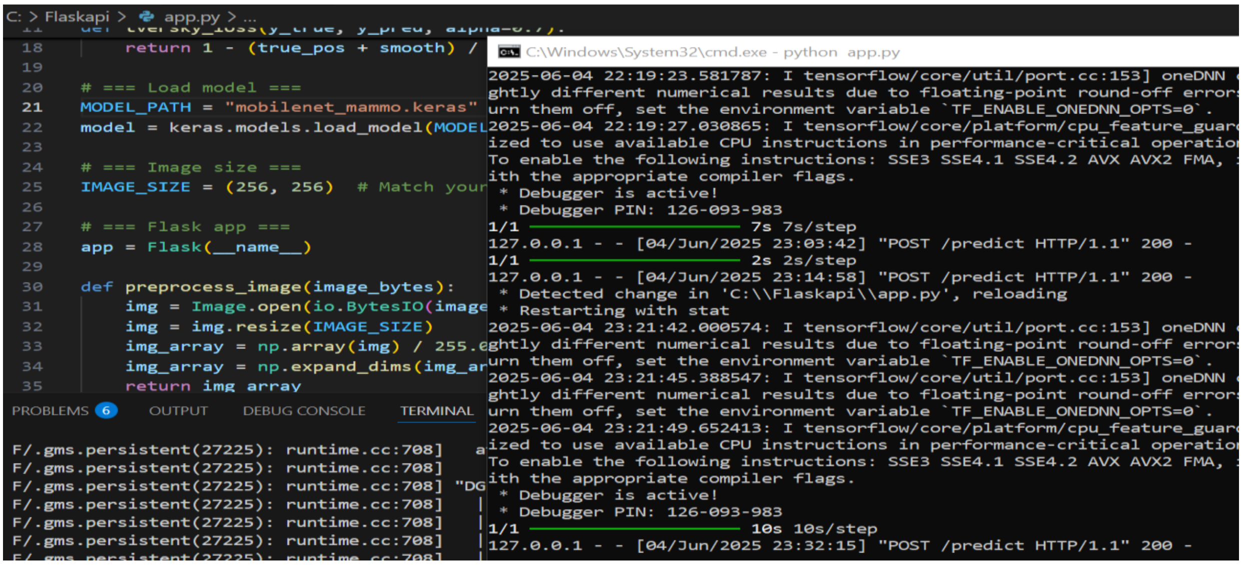


Figure IV. 8 : Activating the server to receive the image request

B. Image Upload and Prediction (Client Side)

1. Image Selection

The user selects a mammogram from the device's gallery using Flutter’s image_picker.

2. Uploading to Server

The app sends the image to a Flask API endpoint /predict using an HTTP POST request.

3. Receiving Result

The Flask server returns the original image with the segmented tumor overlaid in red.

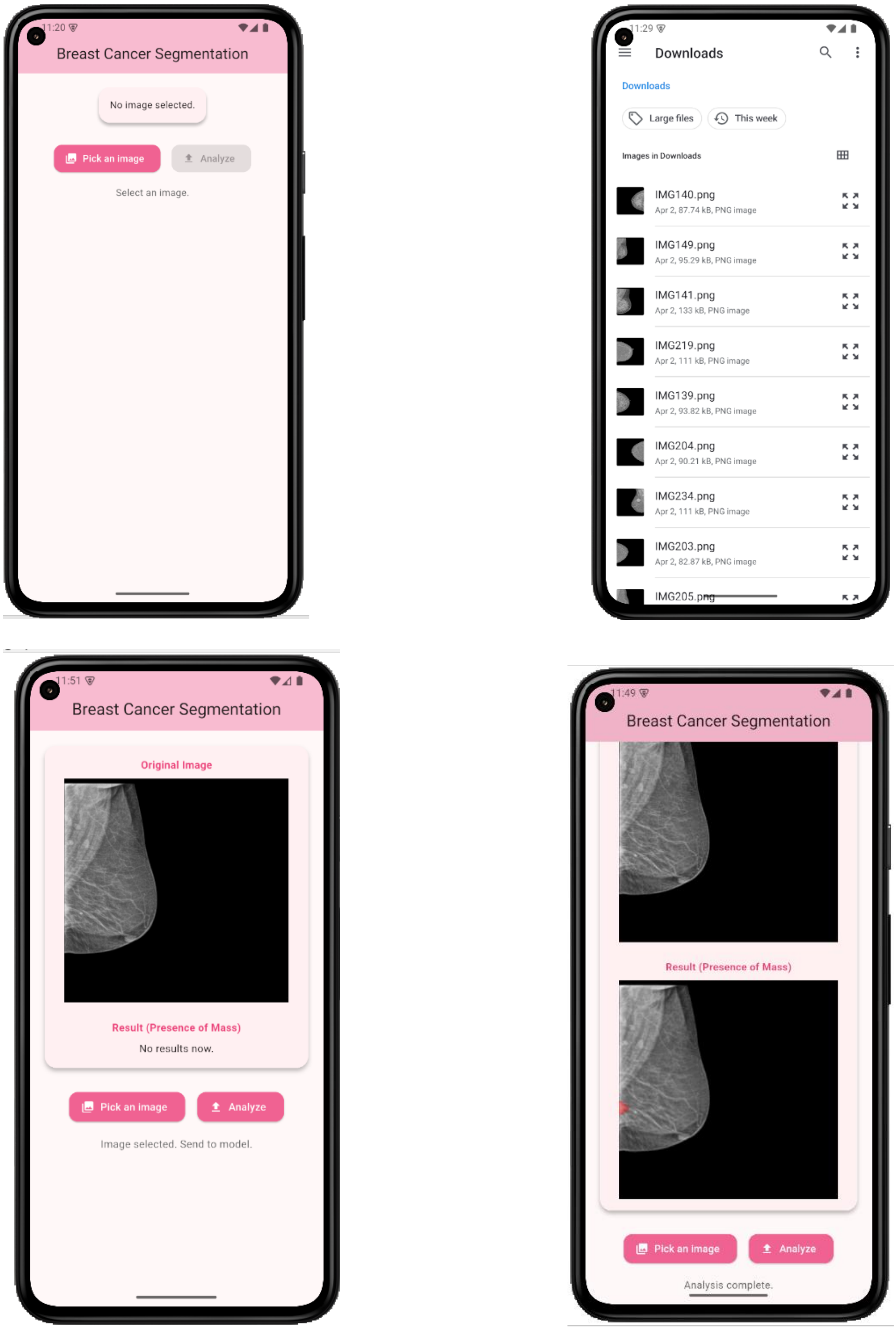


Figure IV. 9: Model testing on the mobile app for small masses

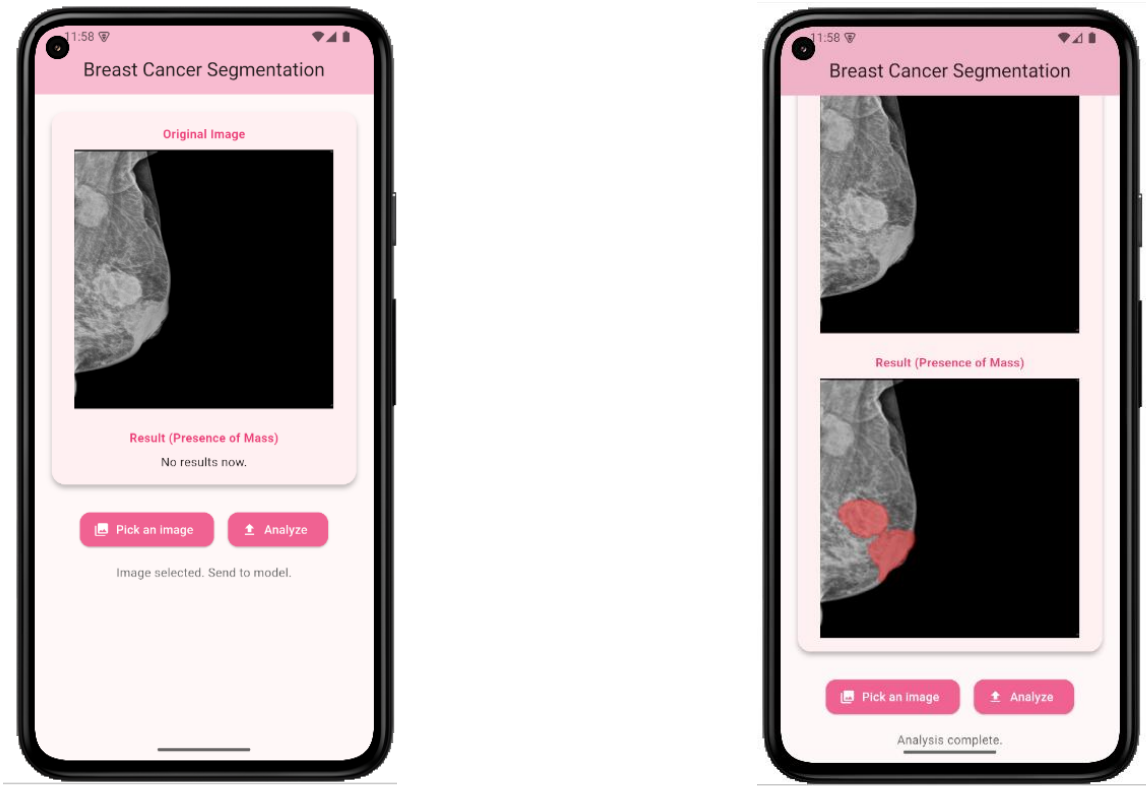


Figure IV. 10 : Model testing on the mobile app big masses

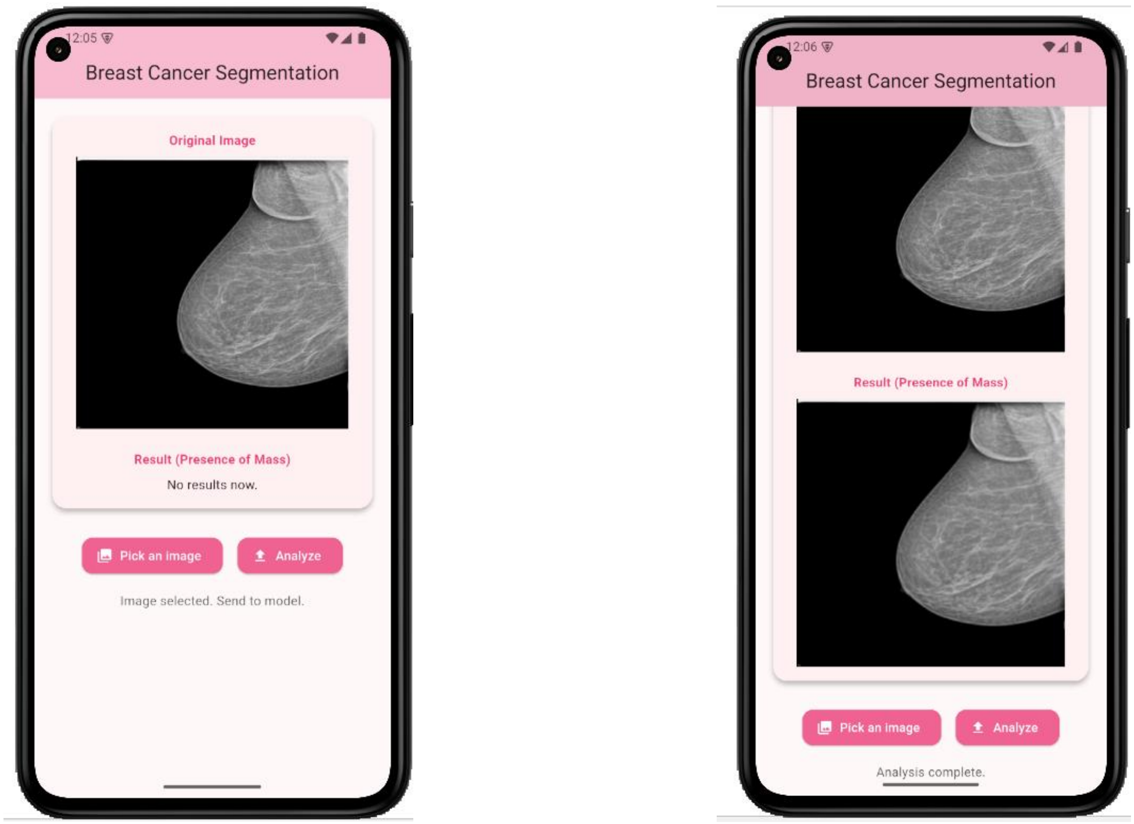


Figure IV. 11 : Model testing on the mobile app for normal breast

IV.8 Conclusion

In this chapter, we presented the experimental setup and detailed implementation of our proposed breast segmentation model. We thoroughly described the preprocessing pipeline applied to the DMID dataset, including image resizing, 3 channel conversion, and normalization to prepare the data for training. Our model architecture combined a pretrained MobileNetV2 encoder with a ResUNet decoder, leveraging transfer learning to effectively extract relevant features for segmentation. We utilized the Tversky loss function and segmentation-specific metrics to accurately evaluate model performance. To enhance generalization and reduce overfitting, we applied data augmentation techniques during training. Additionally, fine-tuning strategies were employed to optimize the pretrained encoder weights for our specific task.

Beyond the model development, we implemented a practical mobile application using Flutter, supported by a Flask API backend. This allowed real-time testing of the segmentation model on user-selected images, providing visual feedback by overlaying predicted masks on original mammograms.

Overall, this chapter validated the technical feasibility of our approach and established a complete end-to-end pipeline, from data preprocessing to user interaction.

General Conclusion

This thesis focuses on advancing the semantic segmentation of mammographic images using deep learning techniques, with the objective of improving the identification of breast masses. It begins by outlining the challenges associated with medical image analysis, emphasizing the critical role of accurate segmentation in the early detection and diagnosis of breast cancer.

The study evaluates two state-of-the-art architectures independently, assessing their performance on a curated dataset of mammograms. While both models demonstrated promising results, they exhibited limitations in segmentation accuracy, reflected by relatively low Dice coefficients and higher loss values.

To overcome these limitations, a hybrid model was proposed, integrating MobileNetV2 as the encoder within the ResUNet architecture. This approach combines the lightweight efficiency of MobileNetV2 with the strong decoding capabilities of ResUNet. Experimental results show that this hybrid architecture significantly improves segmentation performance, achieving higher Dice scores and lower loss compared to the individual models.

These findings underscore the potential of hybrid deep learning architectures in medical image segmentation and pave the way for future research into more efficient and accurate solutions tailored to mammographic analysis.

Key Contributions

• **Evaluation of Existing Models**
A comparative analysis was conducted on the performance of MobileNetV2 with U-Net, ResUNet, and other state-of-the-art models applied to breast mass segmentation in mammographic images.

• **Proposed Hybrid Model**
We proposed a hybrid architecture combining MobileNetV2 as the encoder with a ResUNet-based decoder, specifically optimized for segmentation tasks.

• **Performance Improvement**
The hybrid model outperformed individual models, achieving a higher Dice coefficient and lower loss, thus demonstrating improved segmentation accuracy.

• **Application to a Real Dataset**
The proposed approaches were validated on a real medical dataset, confirming their relevance for clinical image analysis.

• **Mobile Deployment**
The final model was successfully integrated and tested in a mobile application, demonstrating its practical applicability in a real-world healthcare context.

• **Reproducible Pipeline**
A complete and reproducible pipeline was implemented, including preprocessing, model training, evaluation, and deployment.

Implications and Future Work

This work underscores the importance of artificial intelligence in improving the segmentation of breast masses, contributing to earlier detection of breast cancer. Such approaches can be integrated into clinical workflows to automate certain tasks and enhance diagnostic accuracy, thereby supporting healthcare professionals and reducing their workload.

Beyond the technical contributions, this project highlights the real potential of AI to enhance healthcare quality. In Algeria, establishing local medical imaging datasets through the collection and annotation of mammographic data would be a valuable step toward developing models better adapted to the national context.

Future work may focus on continuously improving existing models, exploring additional imaging modalities (such as MRI and ultrasound), and conducting large-scale clinical validation. The ongoing development of a dedicated mobile application for physicians is also a key step, aiming to deliver a complete, automated, and accessible system to support breast cancer diagnosis. This work therefore lays the foundation for practical AI applications in medicine, contributing to more accurate, timely, and accessible healthcare.

Bibliography

[1] A review of biomechanically informed breast image registration - Scientific Figure on ResearchGate. Available from: https://www.researchgate.net/figure/Adult-female-breast-anatomy-illustration_fig1_289525402 [accessed 1 May 2025]

[2] Hoog-Labouret, N., & Merlet, F. (2013). Conséquences des traitements du cancer et préservation de la fertilité : état des connaissances et propositions. Rapport de l'Institut national du cancer et de l'Agence de la biomédecine. Oncologie, 15, 263–270.

[3] Société canadienne du cancer. (2023). Qu'est-ce que le cancer du sein ? https://cancer.ca/fr/cancer-information/cancer-types/breast/what-is-breast-cancer last accessed 2025/04/15

[4] Antoniou, A., Pharoah, P. D., Narod, S., Risch, H. A., Eyfjord, J. E., Hopper, J. L., ... & Easton, D. F. (2003). Average risks of breast and ovarian cancer associated with BRCA1 or BRCA2 mutations. The American Journal of Human Genetics, 72(5), 1117–1130. https://doi.org/10.1086/375033 last accessed 2025/04/15

[5] World Cancer Research Fund / American Institute for Cancer Research. (2018). Diet, Nutrition, Physical Activity and Breast Cancer – Continuous Update Project Expert Report. https://www.wcrf.org/ ;https://www.komen.org/breast-cancer/risk-factor/smoking/ last accessed 2025/04/16

[6] American Cancer Society. Types of Breast Cancer.

https://www.cancer.org/cancer/types/breast-cancer/about/types-of-breast-cancer.html

last accessed 2025/04/16

[7] Mayo Clinic. Lobular Carcinoma in Situ – Symptoms and Causes.

https://www.mayoclinic.org/diseases-conditions/lobular-carcinoma-in-situ/symptoms-causes/syc-20374529 last accessed 2025/04/16

[8]Premier Health Center. Mammography.

https://www.premierhealthcenter.com/mammography last accessed 2025/04/16

[9] American Cancer Society. Breast Ultrasound. https://www.cancer.org/cancer/types/breast-cancer/screening-tests-and-early-detection/breast-ultrasound.html and https://radiologyassistant.nl/breast/ultrasound/ultrasound-of-the-breast last accessed 2025/04/18

Bibliography

[10] Breast magnetic resonance imaging (MRI) is https://www.mayoclinic.org/tests-procedures/breast-mri/about/pac-20384809#:~:text=A%20breast%20MRI%20might%20be,high%20risk%20of%20breast%20cancer. last accessed 2025/04/18

[11] Mayo Clinic. Breast Biopsy – About the Procedure. https://www.mayoclinic.org/tests-procedures/breast-biopsy/about/pac-20384812 last accessed 2025/04/18

[12] American College of Surgeons. Breast Cancer Staging. https://www.facs.org/for-patients/the-day-of-your-surgery/breast-cancer-surgery/breast-cancer-types/breast-cancer-staging/ last accessed 2025/04/25

[13] American Cancer Society. Stages of Breast Cancer.

https://www.cancer.org/cancer/types/breast-cancer/understanding-a-breast-cancer-diagnosis/stages-of-breast-cancer.html last accessed 2025/04/25

[14] BreastCancer.org. Stages of Breast Cancer. https://www.breastcancer.org/pathology-report/breast-cancer-stages last accessed 2025/04/25

[15] American Cancer Society. Breast Cancer Surgery. https://www.cancer.org/cancer/breast-cancer/treatment/surgery-for-breast-cancer.html last accessed 2025/04/25

[16] BreastCancer.org. Radiation Therapy. https://www.breastcancer.org/treatment-side-effects#section-radiation-therapy-side-effects last accessed 2025/04/27

[17] BreastCancer.org. Chemotherapy Side Effects. https://www.breastcancer.org/treatment-side-effects#section-chemotherapy-side-effects last accessed 2025/04/27

[18] BreastCancer.org. Immunotherapy and Targeted Therapy Side Effects. https://www.breastcancer.org/treatment-side-effects#section-immunotherapy-and-targeted-therapy-side-effects last accessed 2025/04/27

[19] Master: Systèmes des Télécommunications Traitement d'images par Mohammed Khamadja et Said Benierbah Available from

https://fac.umc.edu.dz/fstech/cours/Electronique/Master%20ST%C3%A9l%C3%A9com/CoursImageProcessing1.pdf last accessed 2025/03/01

[20] Analysis of Image Enhancement Algorithms. Scientific Figure on ResearchGate. Available from:

https://www.researchgate.net/figure/Wheat-grains-image-taken-in-Australian-University-Canberra-Left-one-is-originally_fig2_319985220 [Accessed February 11, 2025].

Bibliography

[21] Thakur, A., & Gurusamy, M. (2023). Image processing. In Computers and Electronics in Agriculture. Available from: https://www.sciencedirect.com/topics/computer-science/image-processing [Accessed February 8, 2025].

[22] GeeksforGeeks. Digital image processing basics. Available from:

https://www.geeksforgeeks.org/digital-image-processing-basics/. last accessed 2025/02/18

[23] DataScientest. Image processing: Fundamental principles and practical uses. Available from: https://datascientest.com/en/image-processing-fundamental-principles-and-practical-uses . last accessed 2025/02/18

[24] ResearchGate. RGB color images. Available from: www.researchgate.net. last accessed 2025/02/18

[25] Emerging Trends in Image Processing, Computer Vision and Pattern Recognition, 2015 last accessed 2025/02/27

[26] IBM. Image segmentation. Available from: https://www.ibm.com/fr-fr/topics/image-segmentation#:~:text=La%20segmentation%20d'image%20repr%C3%A9sente,classe%20%C3%A0%20une%20image%20enti%C3%A8re.. last accessed 2025/02/18

[27] Omarzai, F. Image segmentation in depth. Medium. Available from: https://medium.com/@fraidoonomarzai99/image-segmentation-in-depth-8ec11fb56371. last accessed 2025/02/18

[28] Classification of Image Segmentation Techniques. Available from: https://pooja-bagad18.medium.com/classification-of-image-segmentation-techniques-cc6a031b75fc. last accessed 2025/02/18

[29] Patil, D. D., & Deore, S. G. Medical image segmentation: A review. S.S.G.B.C.O.E.T. Bhusawal, Dept. of Computer Engg, Jalgaon, Maharashtra, India. last accessed 2025/02/18

[30] Senthilkumaran, N., & Vaithegi, S. (2016). Image segmentation by using thresholding techniques for medical images. Computer Science and Engineering International Journal (CSEIJ), 6(1), 1–13. last accessed 2025/02/18

[31] Thresholding in Image Segmentation. Available from:

https://www.v7labs.com/blog/image-segmentation-guide#thresholding. last accessed

2025/02/18

[32] Cloudinary. Edge-based segmentation. Available from:

https://cloudinary.com/glossary/edge-based-segmentation. last accessed 2025/02/18

[33] Senthilkumaran, N., & Rajesh, R. (2009). Edge detection techniques for image segmentation – A survey of soft computing approaches. International Journal of Recent Trends in Engineering, 1(2), 250. last accessed 2025/02/18

[34] Verma, S. B. Digital Image Processing (MTCS-202). M.Tech (Computer Science), Faculty of Computer Science. last accessed 2025/02/18

[35] TutorialsPoint. Laplacian operator in digital image processing. Available from: https://www.tutorialspoint.com/dip/laplacian_operator.html. last accessed 2025/02/20

[36] GeeksforGeeks. Region and edge-based segmentation. Available from: https://www.geeksforgeeks.org/region-and-edge-based-segmentaion/. last accessed

2025/02/22

[37] Cloudinary. Region-based segmentation. Available from:

https://cloudinary.com/glossary/region-based-segmentation. last accessed 2025/02/22

[38] Bagad, P. Classification of image segmentation techniques. Medium. Available from: https://pooja-bagad18.medium.com/classification-of-image-segmentation-techniques-cc6a031b75fc last accessed 2025/02/24 last accessed 2025/02/23

[39] Encord. Image segmentation for computer vision and **Split and merge segmentation** Best practice guide. Available from: https://encord.com/blog/image-segmentation-for-computer-vision-best-practice-guide/. last accessed 2025/02/25

[40] Image Segmentation Techniques. Scientific Figure on ResearchGate. Available from: https://www.researchgate.net/figure/Example-of-region-splitting-and-merging_fig2_236655552 [Accessed February 14, 2025].

[41] TutorialsPoint. Image segmentation by clustering. Available from: https://www.tutorialspoint.com/image-segmentation-by-clustering. last accessed 2025/02/25.

[42] Aici, Z., & Bentarzi, H. (2022). Different clustering methods applied to image processing in electrical power system field. Algerian Journal of Signals and Systems (AJSS), 7(4), 178. ISSN: 2543-3792, EISSN: 2676-1548. last accessed 2025/02/25.

[43] GeeksforGeeks. Image segmentation with the watershed algorithm in OpenCV-Python. Available from: https://www.geeksforgeeks.org/image-segmentation-with-watershed-algorithm-opencv-python [Accessed February 14, 2025].

[44] Baeldung. Graph-based segmentation. Available from:

https://www.baeldung.com/cs/graph-based-segmentation [Accessed February 26, 2025].

[45] Fiveable Library. Graph-based segmentation study guide. Available from: https://library.fiveable.me/computer-vision-and-image-processing/unit-4/graph-based-segmentation/study-guide/odfdyU1uM5AvQQba [Accessed March 1, 2025].

[46] Issues of Artificial Intelligence Application in Digital Marketing - Scientific Figure on ResearchGate. Available from: https://www.researchgate.net/figure/Relations-between-artificial-intelligence-machine-learning-neural-network-and-deep_fig2_375110440

[47] vlinkinfo. (2024). The Future of AI: How AI Is Changing the World in 2024. https://vlinkinfo.com/blog/future-of-ai-how-ai-is-changing-the-world/ [Accessed: Feb. 4, 2025].

[48] Chaudhari. T. (2024). Artificial intelligence, its types and application in various fields. https://www.researchgate.net/publication/386098684_Artificial_intelligence_its_types_and_application_in_various_fields [Accessed: Feb. 4, 2025].

[49] Srivastava R. (2024). Applications of Artificial Intelligence in Medicine. Explor Res Hypothesis Med.;9(2):138-146. doi: 10.14218/ERHM.2023.00048.

https://www.xiahepublishing.com/2472-0712/ERHM-2023-00048

[50] Wiederhold, G & McCarthy, J. (1992). Arthur Samuel: Pioneer in Machine Learning. IBM Journal of Research and Development. 36. 329 - 331. 10.1147/rd.363.0329.

https://www.researchgate.net/publication/224103556_Arthur_Samuel_Pioneer_in_Machine_Learning

[51] Different Types of Machine Learning Algorithms https://ai.plainenglish.io/different-types-of-machine-learning-algorithms-28974016e108

[52] IBM. Machine learning types. https://www.ibm.com/think/topics/machine-learning-types [Accessed: Feb. 6, 2025].

[53] Types of Machine Learning Alam, Azmir. (2023). What is Machine Learning? 10.5281/zenodo.8231580.
https://www.researchgate.net/publication/373015635_What_is_Machine_Learning

[54] Modeling Course Achievements of Elementary Education Teacher Candidates with Artificial Neural Networks - Scientific Figure on ResearchGate. Available from: https://www.researchgate.net/figure/Similarity-between-biological-and-artificial-neural-networks-Arbib-2003a-Haykin_fig2_326417061 [Accessed: Feb. 13, 2025].

[55] Mahdi, A. B. Artificial neural network. Biomedical Engineering, Thi-Qar University .https://eng.utq.edu.iq/wp-content/uploads/sites/4/2019/09/neural-network-lect.-1-ali-basim-biomedical-eng.pdf [Accessed: Feb. 13, 2025].

[56] Medium. Everything you need to know about neural network, https://medium.com/ravenprotocol/everything-you-need-to-know-about-neural-networks-6fcc7a15cb4 [Accessed: Feb. 13, 2025].

[57] Medium. Neural networks learning process explained,

https://medium.com/@thecybermarty/neural-networks-learning-process-explained-a555d29bf214 [Accessed: Mar. 2, 2025].

[58] TowardsdataScience. Activation Functions in Neural Networks: How to Choose the Right One. https://towardsdatascience.com/activation-functions-in-neural-networks-how-to-choose-the-right-one-cb20414c04e5/ [Accessed: Feb.16, 2025].

[59] Activation Functions in Neural Networks: How to Choose the Right One

https://towardsdatascience.com/activation-functions-in-neural-networks-how-to-choose-the-right-one-cb20414c04e5/

[60] Viso.ai. Deep Neural Network: The 3 Popular Types . https://viso.ai/deep-learning/deep-neural-network-three-popular-types/ [Accessed: Feb.16, 2025].

[61] Purwono, P., Ma'arif, A., Rahmaniar, W., Imam, H., Fathurrahman, H. I. K., Frisky, A., & Haq, Q. M. U. (2023). Understanding of Convolutional Neural Network (CNN): A Review. International Journal of Robotics and Control Systems. 2. 739-748. 10.31763/ijrcs. v2i4.888. https://www.researchgate.net/publication/367157330_Understanding_of_Convolutional_Neural_Network_CNN_A_Review

[62] Adedigba, A. P., Adeshina, S. A., & Aibinu, A. M. (2022). Performance Evaluation of Deep Learning Models on Mammogram Classification Using Small Dataset. Bioengineering, 9(4), 161. https://doi.org/10.3390/bioengineering9040161

[63] Medium. Pre-trained CNN architectures designs, performance analysis and comparison,https://medium.com/@daniyalmasoodai/pre-train-cnn-architectures-designs-performance-analysis-and-comparison-802228a5ce92 [Accessed: May. 15, 2025].

[64] Sandler, M., & Howard, A. (2018, April 3). MobileNetV2: The Next Generation of On-Device Computer Vision Networks. Google Research Blog. Retrieved

from.https://ai.googleblog.com/2018/04/mobilenetv2-next-generation-of-on.html [Accessed: May. 15, 2025].

[65] Hackmd.io. MobileNet-V2: Summary and Implementation.

https://hackmd.io/@machine-learning/ryaDuxe5L [Accessed: May. 15, 2025].

[66] Seidaliyeva, Ulzhalgas & Akhmetov, Daryn & Ilipbayeva, Lyazzat & Matson, Eric. (2020). Real-Time and Accurate Drone Detection in a Video with a Static Background. Sensors. 20. 3856. 10.3390/s20143856.

[67] Types of image segmentation Jung, S., Heo, H., Park, S., Jung, S.-U., & Lee, K. (2022). Benchmarking Deep Learning Models for Instance Segmentation. Applied Sciences, 12(17), 8856. https://doi.org/10.3390/app12178856

[68] Analytics Vidhya. Role of Fully Convolutional Networks in Semantic Segmentation. https://www.analyticsvidhya.com/blog/2024/07/fully-convolutional-networks-for-semantic-segmentation [Accessed: Feb. 16, 2025].

[69] Great Learning. Fully Convolutional Network (Semantic Segmentation).

https://www.mygreatlearning.com/blog/fcn-fully-convolutional-network-semantic-segmentation/ [Accessed: Feb. 16, 2025].

[70] V7labs. Convolutional Encoder- Decoder Architecture.

https://www.v7labs.com/blog/image-segmentation-guide [Accessed: Feb. 16, 2025].

[71] LinkedIn. What is Convolutional Encoder-Decoder Network

https://www.linkedin.com/pulse/what-convolutional-encoder-decoder-network-aionlinecourse-owlvc [Accessed: Feb. 16, 2025].

[72] Medium. SegNet. https://medium.com/@saba99/segnet-a139ce77b570 [Accessed: Feb. 16, 2025].

[73] Hashemi Beni, L., Gebrehiwot, A & Karimoddini, A., Shahbazi, A., &Dorbu, F. (2022). Deep Convolutional Neural Networks for Weeds and Crops Discrimination From UAS Imagery. Frontiers in Remote Sensing. 3. 10.3389/frsen.2022.755939.

https://www.researchgate.net/publication/358532845_Deep_Convolutional_Neural_Networks_for_Weeds_and_Crops_Discrimination_From_UAS_Imagery

[74] Ashraf, H., Waris, A., Ghafoor, M. F., Gilani, S. O., & Niazi, I. K. (2022). Melanoma segmentation using deep learning with test-time augmentations and conditional random fields. Scientific Reports, 12(1), 3948. https://doi.org/10.1038/s41598-022-07885-y]

[75] ResUNet, by Beihang University, 2018 IEEE Geoscience and Remote Sensing Letters, https://sh-tsang.medium.com/brief-review-resunet-road-extraction-by-deep-residual-u-net-f917b13ff6f1

[76] IBM. What is Instance Segmentation. https://www.ibm.com/think/topics/instance-segmentation [Accessed: Feb. 26, 2025].

[77] Sapkota, R., Ahmed, D & Karkee, M. (2024). Comparing YOLOv8 and Mask R-CNN for instance segmentation in complex orchard environments, Artificial Intelligence in Agriculture, Volume 13,2024, Pages 84-99, ISSN 2589-7217,

https://doi.org/10.1016/j.aiia.2024.07.001.
https://www.sciencedirect.com/science/article/pii/S258972172400028X

[78] Viso.ai. Panoptic Segmentation: A Basic to Advanced Guide (2025).https://viso.ai/deep-learning/panoptic segmentation/#:~:text=In%20panoptic%20segmentation%2C%20objects%20in,%2C%20animals%2C%20furniture%2C%20etc. [Accessed: Mar.3, 2025].

[79] Saini, V., Khurana, M., & Challa, R. K. (2025). A hybrid model for the segmentation of mammogram images using Otsu thresholding, morphology and U-Net. Biomedical and Pharmacology Journal, 18(1). https://biomedpharmajournal.org/vol18no1/a-hybrid-model-for-the-segmentation-of-mammogram-images-using-otsu-thresholding-morphology-and-u-net/

[80] Google Colab Research google.

https://research.google.com/colaboratory/faq.html#whatscolaboratory , last accessed

27/05/2025

[81] TensorFlow. Introduction to TensorFlow TensorFlow. https://www.tensorflow.org/learn , last accessed 27/05/2025

[82] The Python Tutorial - Python 3.7.4 documentation Python.org.

https://docs.python.org/3/tutorial/index.html , last accessed 27/05/2025

[83] Keras documentation: About Keras Keras. https://keras.io/about , last accessed

27/05/2025

[84] openCV : https://opencv.org/about/ , last accessed 27/05/2025

[85] Kumawat, S. (2024, May 17). *Flask API for data scientist*. Medium.

https://medium.com/@shrawan662000/flask-api-for-data-scientist-fcbe26da641e

[86]: Digital mammography dataset for breast cancer diagnosis research (**DMID**), https://www.kaggle.com/datasets/orvile/dmid-breast-cancer-mammography-dataset

[87] Oza, P., Oza, U., Oza, R., Sharma, P., Patel, S., Kumar, P., & Gohel, B. (2023). Digital mammography dataset for breast cancer diagnosis research (DMID) with breast mass segmentation analysis. **Healthcare Informatics Research**, Published online December 21, 2023. https://doi.org/ https://pmc.ncbi.nlm.nih.gov/articles/PMC10874363/pdf/13534_2023_Article_339.pdf

[88] Shim, J.-H., Kim, W., Kim, K., Yee, G., Kim, Y. J., & Jeong, T. (2022). Evaluation of U-Net models in automated cervical spine and cranial bone segmentation using X-ray images for traumatic atlanto-occipital dislocation diagnosis. Scientific Reports, 12, Article 23863. https://doi.org/10.1038/s41598-022-23863-w

Bibliography

[89] GeeksforGeeks. (2023, April 18). Calculation Intersection Over Union (IoU) for evaluating an image segmentation model using Java.

https://www.geeksforgeeks.org/calculation-intersection-over-union-iou-for-evaluating-an-image-segmentation-model-using-java/ , last accessed 27/05/2025

[90] Hesaraki, S. (2023, October 27). *Tversky loss*. Medium.

https://medium.com/@saba99/tversky-loss-902f5f8cc35f , last accessed 27/05/2025