NOVA SCHOOL OF SCIENCE & TECHNOLOGY

DEPARTMENT OF PHYSICS

INÊS PIMPISTA METELO

BSc in Biomedical Engineering

RISK PREDICTION ANALYSIS FOR POST-SURGICAL COMPLICATIONS IN CARDIO-THORACIC SURGERY

MASTER IN BIOMEDICAL ENGINEERING

NOVA University Lisbon

June, 2025

NOVA SCHOOL OF SCIENCE & TECHNOLOGY

DEPARTMENT OF PHYSICS

RISK PREDICTION ANALYSIS FOR POST-SURGICAL COMPLICATIONS IN CARDIO-THORACIC SURGERY

INÊS PIMPISTA METELO

BSc in Biomedical Engineering

Adviser: Federico Guede Fernández
Head of Data Science, Value for Health CoLAB

Co-adviser: Ricardo Nuno Vigário
Associate Professor, NOVA University Lisbon

Examination Committee

Chair: Susana Isabel dos Santos Silva Sério Venceslau
Auxiliar Professor, FCT-NOVA

Rapporteur: Ana Teresa Martins Videira Gabriel
Auxiliar Professor, FCT-NOVA

Adviser: Federico Guede Fernández
Head of Data Science, Value for Health CoLAB

MASTER IN BIOMEDICAL ENGINEERING

NOVA University Lisbon
June, 2025

Risk Prediction Analysis for Post-Surgical Complications in Cardio-Thoracic Surgery

This document was created with the (pdf/Xe/Lua)LaTeX processor and the NOVAthesis template (v7.1.18) [1].

Acknowledgements

This dissertation marks the end of an intense five-year journey through my academic life. These years have been filled with challenges and countless learning moments, both intellectually and personally. Looking back, I am immensely grateful for everything this path has taught me and for the people who made it all possible.

First, I would like to express my deepest gratitude to my advisor, Federico, for his invaluable guidance, patience, and encouragement throughout this project. Your mentorship and expertise in the field of AI were fundamental in helping me think critically, refine my ideas, and push the boundaries of my work. I would also like to thank my co-supervisor, Professor Ricardo Vigário, whose knowledge and insightful feedback helped strengthen this work and make it more meaningful. Finally, I am sincerely grateful to the Value for Health CoLAB for welcoming me and providing a supportive and collaborative environment in which to carry out this research.

On a personal note, I extend my deepest thanks to the friends who have walked alongside me on this journey. Margarida, Inês M., Joana, Inês I., Ana M., and Catarina, I am truly lucky to have been surrounded by you. To my lifelong friends, whose unwavering support has remained a constant throughout the years, thank you for always being there. To my boyfriend, thank you for your encouragement, understanding, and constant motivation, I am grateful to have you by my side.

Last but not least, I owe my deepest thanks to my family, especially my parents and my brother, for their unconditional love, belief in me, and constant support. Your encouragement was my foundation through every step of this academic journey, and this achievement is as much yours as it is mine.

Abstract

The detection of wound infections in post cardio-thoracic surgery patients is a critical and challenging issue in postoperative care, due to the high risk of Surgical Site Infections (SSIs) associated with the complexity and invasiveness of these procedures. Early identification of infection risks can significantly improve patient outcomes and reduce healthcare costs.

This thesis, part of the CardioFollow.AI project, a digital telemonitoring service to track the recovery of cardio-thoracic surgery patients, aims to develop an AI-based system for automatic infection risk prediction from patient-submitted wound images. Building upon prior work, this study expands the dataset from 34 to 110 patients and extends the monitoring period from 30 to 90 days, thereby providing a more comprehensive basis for model development.

The proposed system integrates a computer vision pipeline centered on YOLOv8 for wound detection and classification. Two classification strategies were evaluated: a binary model distinguishing between infection risk and non-risk, and a multi-class model that simultaneously predicts wound type (chest, leg, or drain) and infection risk. Data augmentation techniques were implemented to address class imbalance and improve model generalization, while the Optuna framework was used for automated hyperparameter optimization. In addition, a temporal analysis module was integrated to visualize wound progression over time.

The object detection model achieved a final mean IoU of 94.7%, Dice Coefficient of 97.3%, and a Precision of 98.0%. The binary classification approach with data augmentation yielded the highest predictive performance with an F1-score of 98.8% and Kappa score of 0.775, while the multi-class model offered additional clinical detail with slightly lower F1-score, 95.7%, and a Kappa score of 0.902. These results demonstrate the feasibility and clinical relevance of using AI-powered image analysis for early SSI detection and postoperative wound monitoring, paving the way for future integration into telehealth platforms.

Keywords: Cardio-thoracic Surgery, Surgical Site Infections, Image Analysis, Computer Vision, Object Detection, YOLOv8

Resumo

A deteção de infeções em feridas cirúrgicas de pacientes submetidos a cirurgia cardio-torácica constitui um desafio nos cuidados pós-operatórios, devido ao elevado risco de infeções do sítio cirúrgico (SSIs), associado à complexidade e invasividade destes procedimentos. A identificação precoce de sinais de infeção pode melhorar significativamente os resultados clínicos dos pacientes e reduzir os custos associados aos cuidados de saúde.

Esta dissertação, desenvolvida no âmbito do projeto CardioFollow.AI, um serviço digital de telemonitorização que acompanha a recuperação de pacientes após cirurgia cardio-torácica, tem como objetivo desenvolver um sistema baseado em Inteligencia Artificial capaz de identificar o risco de infeção a partir de imagens de feridas submetidas pelos próprios pacientes. Dando continuidade ao trabalho anterior, este estudo alarga o conjunto de dados piloto de 34 para 110 doentes e estende o período de monitorização de 30 para 90 dias, proporcionando uma base mais robusta para o desenvolvimento do modelo.

O sistema proposto integra um pipeline de *Computer Vision* centrado no modelo *YOLOv8* para deteção e classificação de feridas. Foram avaliadas duas estratégias de classificação: um modelo binário, que distingue entre a presença e ausência de sinais de risco de infeção; e um modelo multiclasse, que prevê simultaneamente o tipo de ferida (torácica, perna ou dreno) e o risco de infeção. Para lidar com o desequilíbrio entre classes e melhorar a generalização do modelo, foram aplicadas técnicas de aumento de dados, bem como o framework Optuna para uma otimização automática dos hiperparâmetros. Adicionalmente, foi integrado um módulo de análise temporal para visualizar a progressão das feridas ao longo do tempo.

O modelo de deteção obteve uma *IoU* média de 94.7%, um *Dice Coefficient* de 97.3% e uma *Precision* de 98.0%. Na etapa de classificação, a abordagem binária com aumento de dados revelou o melhor desempenho, apresentando um *F1-score* de 98.8% e um *Kappa score* de 0.775. Por sua vez, o modelo multiclasse ofereceu maior detalhe clínico, com um *F1-score* ligeiramente inferior, 95.7%, mas com um *Kappa score* mais elevado, 0.902. Estes resultados demonstram a viabilidade e a relevância clínica da utilização da análise de imagem assistida por inteligência artificial na deteção precoce de SSIs e no acompanhamento

remoto de feridas cirúrgicas, abrindo caminho para a futura integração destes sistemas em plataformas de telemedicina.

Palavras-chave: Cirurgia Cardio-torácica, Infeções do Sitio Cirurgico, Análise de Imagens, *Computer Vision*, Deteção de Objetos, YOLOv8

Contents

List of Figures x

List of Tables xi

Acronyms xii

1 Introduction 1
- 1.1 Context and Motivation . . . 1
- 1.2 Objectives . . . 2
- 1.3 Document Structure . . . 3

2 Theoretical Concepts 4
- 2.1 Cardio-thoracic Surgery . . . 4
- 2.2 Wound Healing . . . 5
- 2.3 Surgical Site Infection . . . 6
- 2.4 Medical Image Analysis . . . 7
- 2.5 Medical Image Segmentation . . . 7
  - 2.5.1 Traditional Methods . . . 8
  - 2.5.2 Deep Learning Methods . . . 9
- 2.6 Artificial intelligence . . . 10
  - 2.6.1 Artificial Neural Networks . . . 10
  - 2.6.2 Convolutional Neural Network . . . 11
- 2.7 Computer Vision . . . 14
  - 2.7.1 Object detection . . . 14
  - 2.7.2 YOLO . . . 15
- 2.8 Evaluation Metrics . . . 16

3 Literature Review 18
- 3.1 Medical Object Detection . . . 18
- 3.2 AI for diagnostic of Chronic Wounds . . . 18

4 Methodology **21**
4.1 Dataset . . . 21
4.2 Annotation and Labeling . . . 23
4.3 Proposed Pipeline . . . 24
4.4 YOLOv8 Implementation . . . 24
4.5 Data Augmentation . . . 27
4.6 Training Process . . . 27
4.6.1 Optuna . . . 28
4.7 Model Evaluation . . . 29
4.7.1 Wound Detection Evaluation . . . 29
4.7.2 Infection Risk Classification Evaluation . . . 29
4.8 Wound Healing Analysis . . . 30

5 Results and Discussion **31**
5.1 Wound Detection Model . . . 31
5.2 Wound Detection and Risk Classification Model . . . 32
5.2.1 Comparison between dataset sizes . . . 35
5.2.2 Comparison between different cross validation strategies . . . 35
5.3 Wound Temporal Analysis . . . 36

6 Conclusion **39**
6.1 Main Conclusions . . . 39
6.2 Limitations and Future Work . . . 40

Bibliography **41**

Annexes

I Hyperparameter Search Results **48**

List of Figures

2.1 Median sternotomy incision 5
2.2 Difference between semantic segmentation, instance segmentation, and object detection 9
2.3 Diagram of CNN architecture 11
2.4 YOLOv1 Architecture 16
2.5 YOLO process for object detection 16

4.1 Wound's types 22
4.2 Example of the evolution of a patient's LW 22
4.3 Example of a ROI delineation on LabelMe 23
4.4 Proposed Pipeline 24
4.5 Model Structure of YOLOv8 25

5.1 Confusion matrices for binary risk classification 34
5.2 Confusion matrices for multi-class risk classification 34
5.3 Infection risk probability trends for patients without infection risk across the follow-up period 37
5.4 Infection risk probability trends for patients with infection risk across the follow-up period 37
5.5 Infection risk probability during the 7 days preceding the first high-risk prediction 38

List of Tables

2.1 Evaluation metrics for AI algorithms 17

4.1 YOLOv8 Model Variants Comparison 26

5.1 Comparison of wound detection model performance on different sizes of dataset 31
5.2 Best hyperparameter configuration found for Wound Detection Model . . . 31
5.3 Classification performance metrics across different experimental configurations 33
5.4 Best hyperparameter configuration found for Wound Detection and Risk Classification Model 33
5.5 Performance metrics of Binary Classification on different sizes of the dataset 35
5.6 Performance metrics across different cross validation strategies on Binary Classification 36

I.1 Results for the Object Detection Model with initial dataset 48
I.2 Results for the Object Detection Model with expanded dataset 48
I.3 Results for Wound Detection and Risk Classification – Binary Classification without Augmentation 49
I.4 Results for Wound Detection and Risk Classification – Binary Classification with Augmentation 49
I.5 Results for Wound Detection and Risk Classification – Multiclass Classification without Augmentation 49
I.6 Results for Wound Detection and Risk Classification – Multiclass Classification with Augmentation 49
I.7 Results for Wound Detection and Risk Classification – Binary Classification without Augmentation for Initial Dataset 50
I.8 Results for Wound Detection and Risk Classification – Binary Classification without Augmentation with Cross Validation by Patients 50

Acronyms

2D Two Dimension *(p. 7)*

AI Artificial Inteligence *(pp. viii, 10, 18–20, 23)*
ANN Artificial Neural Network *(pp. 10, 11, 13)*
AP Average Precision *(pp. 16, 17)*

CABG Coronary Artery Bypass Grafting *(pp. 1, 4)*
CNN Convolutional Neural Network *(pp. 9, 11, 13, 15, 19)*
CT Computed Tomograph *(pp. 7, 18)*
CV Computer Vision *(pp. 7, 14, 18, 23)*
CW Chest Wound *(pp. 21, 22)*

DL Deep Learning *(pp. 7, 8, 10, 11, 14, 18, 19, 24, 28, 39, 40)*
DW Drainage Wound *(pp. 21, 22)*

GPU Graphics Processing Unit *(pp. 14, 27)*

IoU Intersection over Union *(pp. 16, 17, 19, 29, 48)*

kNN k-Nearest Neighbors *(p. 19)*

LW Leg Wound *(pp. 21, 22)*

mAP mean Average Precision *(pp. 16–20, 48)*
ML Machine Learning *(pp. 10, 19)*
MRI Magnetic Resonance Imaging *(pp. 7, 18)*

NMS Non-Maximum Suppression *(pp. 15, 25)*

ROI Region of Interest *(pp. 3, 14, 23, 24)*

SSI Surgical Site Infection *(pp. 1, 6, 7)*

SVM Support Vector Machine *(p. 19)*

1 Introduction

1.1 Context and Motivation

Cardio-thoracic diseases encompass a spectrum of medical conditions that affect the thorax area, including the heart, lungs and associated structures, and stand as the leading cause of death worldwide [2]. Given their complexity, cardio-thoracic surgeries play a critical role in the treatment of these conditions, with open heart surgery, the most common type, estimated to be approximately 2 million procedures worldwide annually [3]. However, the immediate proximity of the surgical site to vital organs represents a challenge in postoperative care, and SSIs pose a significant health burden for patients and healthcare providers.

According to the 2018-2020 European Center for Disease Control and Prevention epidemiological report, the percentage of Surgical Site Infections in Coronary Artery Bypass Grafting (CABG), the most commonly performed cardiac surgery procedure, is 2% and therefore addressing SSIs is important to improve patient outcomes and reduce healthcare costs.

One promising approach to mitigate these challenges lies in the use of digital post-surgical monitoring. These services enable daily patient clinical data checks which can facilitate early interventions. In particular, leveraging AI to analyze patient-submitted wound images offers the potential to detect subtle changes before obvious signs of infection manifest, which can help prevent serious complications and optimize resource allocation by alleviating the burden on healthcare professionals and health systems.

This master's dissertation is part of the CardioFollow.AI project, a research initiative funded by the Portuguese Foundation for Science and Technology (FCT), which focuses on the development of AI-powered algorithms to enhance the post-discharge surveillance of cardiac surgery patients at Hospital de Santa Marta. The project brings together a multidisciplinary team of researchers from Value for Health CoLAB, Fraunhofer-AICOS, Nova Medical School-UNL and Hospital de Santa Marta-CHULC. Its main goals are the achievement of higher health outcomes, the improvement of patient's engagement and the increase of its cost/effectiveness [4].

The service provides a digital health kit equipped with Internet-of-Things devices to eligible patients, which includes a sphygmomanometer, an electric scale, a smartwatch and a smartphone. These allow the daily collection of data by patients enabling them to monitor their blood pressure, weight, physical activity and heart rate, as well as submit daily photographs of their surgical wound and report symptoms. Following discharge, patients are remotely monitored for a 30-day or 90-day period, during which nurses review the incoming data through a dedicated web platform. This platform generates daily reports and data visualization tools to monitor patient's recovery status and alerts when abnormal alterations occur [4].

As patient enrollment has expanded, the volume of collected data has likewise grown, creating the need for automated methods to efficiently identify high-risk cases. Although CardioFollow.AI already employs AI-driven solutions to generate certain risk indicators, the photographic monitoring of surgical wounds remains a daily manual process for the clinical team. This gap, coupled with the substantial risk of developing surgical site infections, which can increase morbidity, mortality, and healthcare costs, highlights the importance of developing an automated image-based risk evaluation. Therefore, implementing such a system would not only improve efficiency for healthcare providers but would also be highly beneficial for patients, as detecting the risk of infection earlier allows timely medical intervention, reduce complications, and improve overall outcomes.

This dissertation builds upon previous work developed within the CardioFollow.AI project, particularly the preceding master's thesis, focused on wound segmentation and infection prediction, which established the clinical relevance and feasibility of AI-based tools in postoperative care [5]. Consequently, the present work focuses on developing and validating an AI-based risk prediction model for wound infection detection. By extracting relevant features from wound images to identify early indicators of infection and analyzing the healing progress over time, this model aims to diminish hospital re-admissions and reduce the workload for clinical professionals. In so doing, it will improve patient safety and further contribute to the CardioFollow.AI mission of leveraging telemonitoring to improve postoperative health outcomes, patient empowerment and overall cost-effectiveness of healthcare.

1.2 Objectives

The primary goal of this dissertation was to automate the analysis of post-operative wound images by developing a risk prediction model. The system was designed to facilitate early detection of concerning wound alterations, thereby preventing further infections and enabling healthcare professionals to intervene promptly. To achieve this overarching goal, the following sub-objectives have been established:

1. Dataset Expansion and Annotation - Increase the pilot dataset size from 34 to 110 patients and extend the post-surgical observation period from 30 to 90 days by manually

label and annotate wound images to create a comprehensive, high-quality dataset for training and validation.

2. Object Detection and Classification Model - Create a model that combines object detection and classification to accurately identify the wound region and predict infection risk.

3. Wound Healing Process Analysis - Investigate wound healing progress by a quantitative assessment of infection risk probability.

The core of this dissertation lies in computer vision to develop a robust detection and classification pipeline, with object detection serving as a critical first step. By isolating only the surgical ROI, the system can concentrate on clinically relevant local signs of infection. Subsequently, the classification model processes extracted features to yield a final prediction of infection risk, based on recognized indicators of wound deterioration or anomalies.

By integrating infection risk prediction models into the existing CardioFollow.AI telemonitoring infrastructure, this dissertation seeks to reduce hospital re-admissions through timely clinical alerts and interventions, enhance patient safety by offering an automated, continuous monitoring tool for postoperative wound assessment, and contribute to the field of medical image analysis patient care optimization, ultimately supporting the broader CardioFollow.AI mission to improve outcomes, enhance patient empowerment, and increase cost-effectiveness.

1.3 Document Structure

This dissertation is divided into six chapters. The first one, the Introduction, establishes the study's relevance and objectives. The Theoretical Concepts Chapter provide an overview of the foundational ideas necessary to understand the research and the Literature Review one summarizes the relevant studies and publications related to it. The Methodology outlines the procedures, analytical approaches, and data-processing techniques, while the Results and Discussion presents the main findings, examines their significance, and ties them back to the study's objectives and existing literature. Finally, the Conclusion highlights the key insights, discusses potential limitations, and suggests avenues for further research, underlining the study's contributions to the field.

The study was conducted in accordance with the Declaration of Helsinki, and approved by the Institutional Review Board (or Ethics Committee) of Centro Hospitalar Lisboa Central (protocol code CA2651 2020/06/08 and CA2011 2024/04/29). Informed consent was obtained from all subjects involved in the study.

The research work described in this dissertation was carried out in accordance with the norms established in the ethics code of Universidade Nova de Lisboa. The work described and the material presented in this dissertation, with the exceptions clearly indicated, constitute original work carried out by the author.

2

Theoretical Concepts

This chapter presents the theoretical concepts necessary to understand this document, including: Cardio-thoracic Surgery, Wound Healing Process, Surgical Site Infections, Deep Learning, Computer Vision and related topics.

2.1 Cardio-thoracic Surgery

Cardio-thoracic surgery occupies a central role on medical surgeries, addressing complex pathologies which affect vital structures within the chest cavity, including the heart, lungs, esophagus, and other thoracic organs. It is subdivided into cardiovascular surgery, focusing on the heart and major blood vessels, and thoracic surgery, addressing conditions affecting the lungs, mediastinum and chest wall. It can be performed with different types of procedures: open surgery or minimally invasive surgery, as endoscopic and robotic [6].

Open heart Surgery, a conventional approach, involves accessing the heart directly through an incision and is commonly performed for coronary revascularization, heart valve repair or replacement, congenital heart defect correction and implantation of medical devices such as pacemakers and defibrillators [7]. CABG stands as the most commonly performed cardiac surgery procedure worldwide [8].

CABG involves rerouting blood flow around obstructed coronary arteries using grafts harvested from alternative vascular sources within the body. The selection of grafts depends on the patient's specific anatomy and the location of arterial blockages. Common grafts utilized in CABG include the left internal mammary artery and saphenous vein grafts sourced from the lower extremities. Additional conduits that may be employed include the right internal mammary artery, the radial artery, and the gastroepiploic artery [9].

Central to the success of open heart surgery is the sternotomy, a surgical technique performed in over 1 million patients per year worldwide [10]. In this procedure, as illustrated in Figure 2.1, a median and vertical incision (12 to 18 cm) between the sternal notch and the tip of the xiphoid process is made. In a CABG surgery, the incision needs to be extended into the upper part of the linea alba [11]. After this step, the patient's sternum

is split until there is enough room for a Finochietto retractor to be inserted to separate the ribs and make the heart visible [12].

Upon completion of the surgery, stainless steel wires are used to close the breastbone, followed by suturing of the skin. The closure is extremely important since the inappropriate sternal approximation can cause postoperative pain and dehiscence, which is a risk factor for wound infection [12].

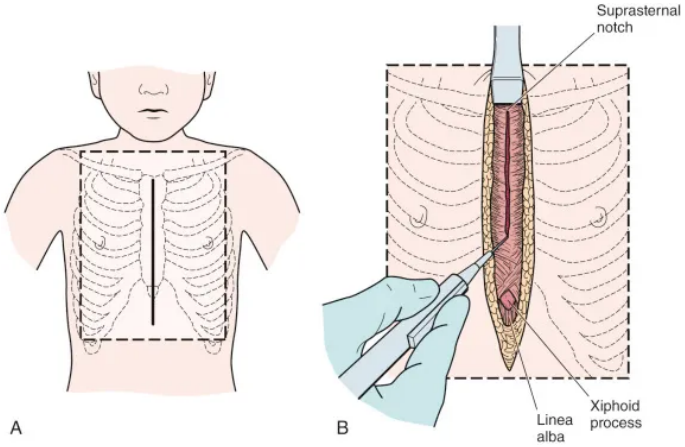


Figure 2.1: Median sternotomy incision [13]

2.2 Wound Healing

Wound healing is a natural physiological process for tissue injury that can be broken down into four distinct phases - hemostasis, inflammation, proliferation and tissue remodelling - reling on a complex interplay of cellular and molecular mechanisms [14].

The hemostasis phase is the body's natural response to injury and takes place immediately after the wound's formation. To prevent exsanguination, vasoconstriction occurs and platelets undergo activation, adhesion and aggregation to form a fibrin clot, establishing a provisional matrix that facilitates the recruitment of immune cells and the release of signaling molecules [15].

Following clot formation, comes the inflammatory phase, where increased blood flow brings phagocytic leukocytes—mainly neutrophils and macrophages—to the wound site. These cells clear debris, engulf pathogens, and secrete various growth factors essential for the next stages. During this phase, the wound typically exhibits the classic signs of inflammation —redness (erythema), heat, edema, pain and functional limitation [15] [16].

The wound starts to rebuild itself in the proliferative phase by the formation of granulation tissue, composed of collagen and extracellular matrix. Angiogenesis ensures an adequate blood supply, while fibroblasts help generate new connective tissue. As the wound contracts, epithelial cells migrate from the wound edges, ultimately re-epithelializing the surface and closing the defect [16].

The final stage of wound healing is remodeling, which occurs once the wound is closed. In this phase, the wound regains its tensile strength as the collagen fibers within the wound remodel and reorganize themselves. During this phase, the wound also devascularizes and returns to its original state of blood supply [16].

Overall, the interplay of vascular responses, immune activity, and cellular proliferation is vital for successful healing. Nonetheless, the natural inflammatory signs—such as mild redness, warmth, and tenderness—can sometimes be mistaken for infection, underscoring the importance of careful observation [16]. In cardio-thoracic surgery, accurately distinguishing between normal healing and the early onset of infection is particularly crucial, as prompt recognition and intervention can avert complications like SSIs and greatly improve patient outcomes.

2.3 Surgical Site Infection

An infection is defined as "the invasion of an organism's body tissues by disease-causing agents and their multiplication, as well as the reaction by the host to these organisms and/or toxins that the organisms produce" and it is a major concern in postoperative care, especially in the context of cardio-thoracic surgeries, where the risks are heightened due to the complexity and invasiveness of the procedures [17].

An infection that occurs in the wound created by an invasive surgical procedure is generally referred to as Surgical Site Infection (SSI). SSI is defined by the Centers of Disease Control and Prevention as the infection occurs within 30 days of the surgical operation at the site of the surgery or within 1 year with prosthetic material implantation [18]. Common symptoms include redness, exudate, necrotic tissues, separation of wound edges and swelling, although systemic signs such as fever, chills, elevated blood pressure, sudden and intense thoracic pain, and loss of conscious can also be present [19]. SSIs can be classified by three categories [20]:

1. Superficial incisional SSI - occurs just in the area of the skin where the incision is; symptoms typically include redness, swelling, heat, pain, and drainage of pus.

2. Deep incisional SSI - occurs beneath the incision area in muscle and the tissues surrounding the muscles; symptoms can include fever, localized pain or tenderness, and the presence of pus or an abscess.

3. Organ or space SSI - occurs in any area of the body other than skin, muscle and surrounding tissue that was involved in the surgery, this includes a body organ or a space between organs; these infections occur within 30 days if no implant is in place, or within one year if an implant is in place; symptoms are often more systemic and can include fever, an increased heart rate, and drainage from a drain placed through the skin into the organ/space.

Several factors contribute to the increased risk of SSIs in cardio-thoracic surgeries, including surgical complexity and duration, use of prosthetic materials, and patient

comorbidities, such as as diabetes, obesity, and cancer. [20]. Early detection of SSIs is fundamental to reduce hospital readmissions and costs and to improve patients outcomes.

2.4 Medical Image Analysis

Medical Image Analysis is the field of study which extracts meaningful information from medical images, often using computational methods, facilitating the diagnosis, monitoring and treatment of various conditions [21]. There are several types of medical images, including X-rays, Computed Tomograph (CT), Magnetic Resonance Imaging (MRI), ultrasound and, more recently, digital photographs of wounds [22]. The analysis of medical images involves several key processes, such as image pre-processing, segmentation, object detection, feature extraction and classification.

Digital wound photographs, a non-invasive technique, have become widely used in the clinical daily routine, allowing qualitative evaluation of symptoms at the site of the lesion. Digital images are a matrix representation of 2D images using a finite number of digital values, known as pixels, each containing information about intensity and color. The color representation can vary, with gray-scale images using a single value representing the intensity of a pixel(0-255), and RGB images, represented by three values, each one representing the amount of colors red (R), green (G), and blue (B) [23].

The use of wound digital photographs is simple and practical, requiring minimal technical expertise and making it particularly well suited for home-based monitoring. By enabling patients or caregivers to capture images in their own environment, this approach supports a more continuous and accessible assessment of wound healing. However, despite its convenience, photographic monitoring still depends on medical interpretation and can be time-consuming for clinicians [24]. Manual analysis also introduces subjectivity and may suffer from variability due to human factors, such as fatigue or inconsistent evaluation criteria, which can be mitigated with the introduction of CV [25].

2.5 Medical Image Segmentation

Image segmentation is the process of partitioning a digital image into non-overlapping regions that share similar properties, such as gray level, color, texture, brightness, and contrast in order to extract meaningful information [26].

This process is essential in digital wound photographs to extract the region of interest, for measuring wound dimensions, monitoring healing progress, and identifying infection signs, which can help the clinicians to make accurate diagnoses and better treatment decisions.

Medical image segmentation can be done manually, semi-automatically, or fully-automatically, based on the need of user interations, and can be conducted either by traditional algorithms or DL algorithms, each having its own advantages and disadvantages depending on the given circumstances. Traditional algorithms are widely adopted,

however, they rely heavily on features being extracted from data by domain specialists which might overlook some less obvious attributes that we need to segment. On the other hand, DL automatically extracts features, but it requires an extensive dataset for better outcomes [27].

2.5.1 Traditional Methods

2.5.1.1 Threshold-based Segmentation

Threshold-based methods are the simplest techniques for image segmentation since they are based on the pixel intensity differences between the background and the object. To compare this difference, each pixel intensity value is compared to a threshold and assigned to a specific class. There are different types [28]:

1. **Single Threshold**: A simple global threshold T is chosen, and each pixel p is classified as belonging to the object if $I(p) \geq T$ or to the background otherwise.

2. **Multiple Thresholds**: For multi-class segmentation, multiple thresholds can partition the intensity histogram into multiple intervals.

3. **Adaptive/Local Thresholding**: Chooses a threshold value for each pixel based on local image characteristics, helping deal with non-uniform lighting for example.

2.5.1.2 Edge or Boundary-based Segmentation

A edge is the boundary between two regions with different properties and is typically identified when an abrupt change of intensity occurs. The main idea in edge-based methods is the computation of local intensity gradients to highlight boundaries, linking them together to form closed object boundaries. This type of segmentation works well on images with high contrast between the object and the background [28].

2.5.1.3 Region-based Segmentation

Region-based segmentation algorithms group pixels into regions based on predefined criteria, such as gray level, texture, color, shape. There are a variety of approaches of region-based segmentation. These methods can be classified into two categories[28]:

1. **Region growing method**: This approach is based on predefined criteria and starts with the selection of one or more seed points in the image. Secondly, the regions grown from these seed pixels to the neighboring pixels and analyze if they meet similarity criteria, if so they are added to the same region. The second process is iterated on, in the same manner as general data clustering algorithms. In noisy images, these technique are better but more computationally expensive.

2. **Split-and-merge method**: This approach segment an image through two sequential operations:region splitting and region merging. Region splitting begins with the whole image as an region and subdivides it into sub-regions recursively until some condition of

homogeneity is not satisfied. Conversely, region merging avoids the excessive subdivision by starting with small regions and merging the regions that have similar characteristics.

2.5.2 Deep Learning Methods

Deep learning has revolutionized medical image analysis, particularly in tasks such as object detection and image segmentation, by leveraging CNNs for sophisticated pattern recognition. In the context of image segmentation, the output typically consists of segmentation masks, representing the specific pixel-by-pixel boundary and shape of each class, typically corresponding to different objects, features or regions in the image. The two predominant segmentation paradigms in deep learning are semantic segmentation and instance segmentation [29], while object detection constitutes a complementary approach.

1. **Semantic Segmentation** - subcategory of image segmentation and is defined as the process of assigning a meaningful label to every pixel in the image. The difference between the normal segmentation and semantic segmentation is that all objects of the same class share a single label, which means there is no distinction between objects belonging to the same class.

2. **Instance Segmentation** - subcategory of image segmentation but not only classifies each pixel into a semantic category but also differentiates between distinct instances of the same category. For example, if there are multiple wounds in a scan, instance segmentation aims to assign each wound a unique label.

3. **Object Detection** – unlike segmentation, object detection does not predict the class of each pixel. Instead, it identifies and localizes objects in an image by drawing bounding boxes around them and assigning class labels, making it computationally efficient and suitable for tasks where precise localization is sufficient, such as detecting wound regions prior to classification or segmentation [30].

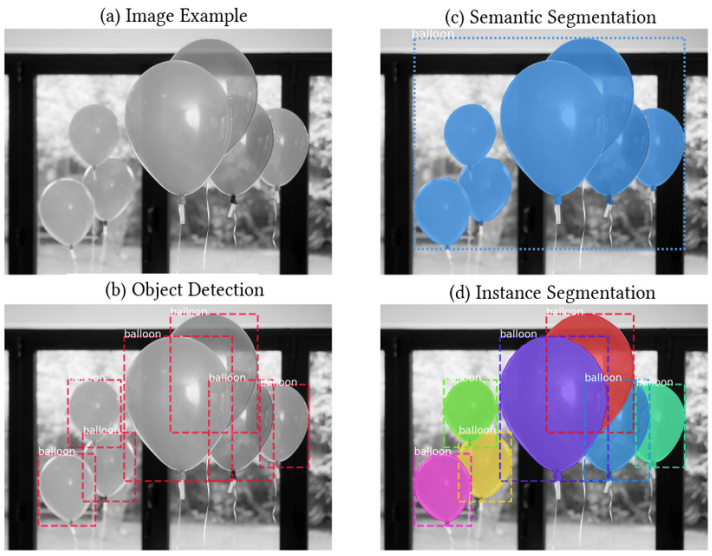


Figure 2.2: Difference between semantic segmentation, instance segmentation, and object detection [31]

2.6 Artificial intelligence

Artificial Inteligence (AI) refers to the computer science field where a system is able to interpret and learn from external data, recognizing patterns, and achieve specific goals, similar to human intelligence [32].

Machine Learning (ML) can be defined as a subset of AI in which computing systems learn from data, identify patterns and use algorithms to execute tasks without being explicitly programmed. In the context of healthcare, ML has shown promising results, enabling computers to process medical data, identify health patterns and make predictions. The fundamental concept behind ML involves several components such as models, training and evaluation [33]. ML models can be divided into two groups: supervised learning and unsupervised learning. Supervised learning uses labeled data to make predictions, while unsupervised learning uses unlabeled data and work independently to identify patterns [33].

ML methods have been developing rapidly but getting the desired performance is still a challenge. To address this, recent advances have focused on automating key stages of the model development process, such as algorithm selection, feature engineering, and hyperparameter optimization. These approaches aim to reduce the complexity and manual effort involved in traditional ML workflows, making model development more accessible and scalable [34].

Deep Learning (DL) is a subset of ML based on the concept of Artificial Neural Network (ANN) to simulate the learning process of human brain's neural structure.

2.6.1 Artificial Neural Networks

ANN correspond to networks of artificially nodes, known as neurons, each of which is a small processing unit that can be mathematically interpreted as a non-linear transfer function controlled by weights, designed to extract higher-level features from data [35]. ANNs are composed of multiple layers, each serving a specific function [36]:

- Input Layer: Receives raw data features and passes them to the next layer.
- Hidden Layers: Perform feature transformation using weighted connections and activation functions. A neural network can contain multiple hidden layers, each refining the data representation.
- Output Layer: Produces the final output or prediction based on the computations performed by previous layers. It can contain one or more nodes depending on the task.

Each neuron receives inputs from the previous layer, computes a weighted sum, and applies an activation function to introduce non-linearity. The output of a neuron is mathematically represented as [35]:

$$y = f\left(\sum_{i=1}^{n} w_i x_i + b\right) \tag{2.1}$$

where y is the output, x_i represents input features, w_i are the corresponding weights, b is the bias term, and $f(\cdot)$ is the activation function that determines the model's non-linearity and learning capability.

ANNs learn by adjusting their weights to minimize a loss function. This optimization is typically achieved using gradient descent, with the most common weight adjustment technique being backpropagation. In backpropagation, gradients are computed using the chain rule to update weights as follows [35]:

$$w_i^{(t+1)} = w_i^{(t)} - \eta \frac{\partial L}{\partial w_i} \tag{2.2}$$

where η is the learning rate and L is the loss function.

By iteratively refining the weights, ANNs improve their ability to recognize patterns, making them powerful tools for tasks such as feature extraction and classification.

2.6.2 Convolutional Neural Network

Convolutional Neural Network (CNN), illustrated on Figure 2.3, is one of the most prominent and employed architecture in DL field. Its key advantage over other models is its ability to automatically identify relevant features without human supervision, which has significantly advanced the field of computer vision, including tasks such as object detection and image segmentation.

CNN consist of three main types of layers: convolutional, pooling and fully connected layers, each described in the next subsections, which work together to automatically extract and learn features from images, progressing from simple features like edges and colors to complex shapes and objects. CNNs are computationally expensive but outperform traditional approaches in terms of both accuracy and efficiency [37].

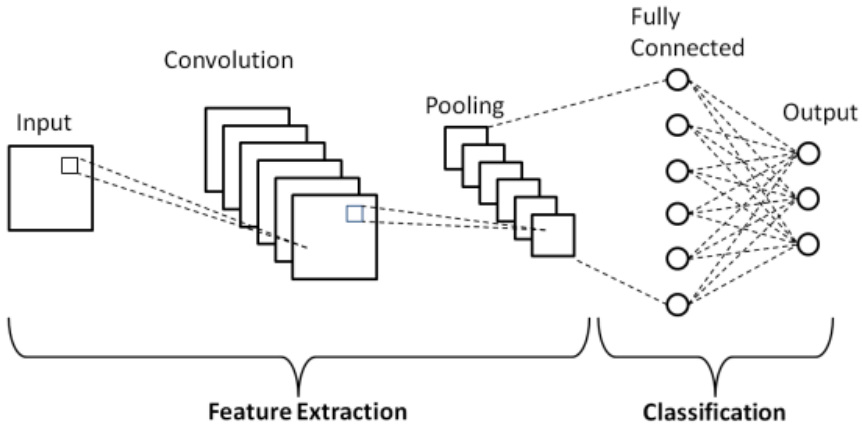


Figure 2.3: Diagram of CNN architecture [38]

2.6.2.1 Convolutional Layer

The convolution layers are the fundamental blocks of CNNs since they are responsible for extracting the features from the input images. These blocks perform a mathematical operation named convolution which consists in the application of filters, known as kernels, through the input image to generate the output feature map [39]. A kernel can be described as a matrix of values or numbers (weights of the kernel), smaller than the input, that moves across the input image matrix. In the beginning of the training process, the weights of a kernel are assigned with random numbers, but over each training step they are adjusted [40]. The convolution operation consists in sliding the filter across the input to generate the output feature map. It is described as the multiplication of the kernel values by the input values at each location and their subsequent sum to obtain a single value. For example, if we use a two-dimensional image I as our input and also a two-dimensional kernel K, the formula of the convolution operation is given by [40]:

$$S(i,j) = (I * K)(i,j) = \sum_m \sum_n I(m,n)K(i-m, j-n). \tag{2.3}$$

where $S(i,j)$ is the output feature map value at position (i,j), $I(m,n)$ is the input image value at position (m,n), and $K(i-m, j-n)$ the kernel value at position $(i-m, j-n)$.

After performing the complete convolution operation, the final feature map is obtained with it dimension depending on several factors, including the kernel size, the stride value, and whether padding is applied to the input [40].

Padding plays a crucial role in determining the spatial dimensions of the feature map. Without padding, the size of the output feature map decreases as the kernel slides over the input image. By applying padding, the borders of the input image are preserved, allowing the convolution operation to capture edge features more effectively while maintaining the spatial size of the feature map [40].

Stride is another parameter that affects the feature map size. It defines how much the filter moves at each step. A stride of one means the kernel shifts one pixel at a time, while a larger stride results in a reduced feature map with fewer computations [40].

The formula to find the output feature map size after the convolution operation is given by [40]:

$$h' = \left\lfloor \frac{h - f + p}{s} + 1 \right\rfloor \tag{2.4}$$

$$w' = \left\lfloor \frac{w - f + p}{s} + 1 \right\rfloor \tag{2.5}$$

Where h' denotes the height of the output feature map, w' denotes the width of the output feature map, h denotes the height of the input image, w denotes the width of the input image, f is the filter size, p denotes the padding of the convolution operation, and s denotes the stride of the convolution operation.

2.6.2.2 Pooling Layer

Pooling layers are the blocks responsible for sub-sample the feature maps after convolution operation, which means shrinking the larger size features map to create smaller ones, preserving the most dominant features. Similarly to the convolutional operation, both the stride and kernel are intrinsically specified. There are several types of polling techniques such as max pooling, min pooling, average pooling, and gated pooling. The output feature map size after pooling operation is given by [40]:

$$h' = \left\lfloor \frac{h - f}{s} \right\rfloor \tag{2.6}$$

$$w' = \left\lfloor \frac{w - f}{s} \right\rfloor \tag{2.7}$$

Where h' and w' represent the height and width of the output feature map, respectively, h and w denote the height and width of the input feature map,f is the size of the pooling region size, and s represents the stride of the pooling operation.

2.6.2.3 Fully Connected Layer

In the final layer of every CNN architecture, each neuron is fully connected to all neurons from its previous layer, forming what is known as the Fully Connected Layer. This block acts as the output of the CNN, essentially functioning as a multilayer perceptron, a type of feedforward ANN [40].

The input to the Fully Connected layer comes from the final pooling or convolutional layer and is represented as a vector. Mathematically, the output of a fully connected layer is computed as follows [40]:

$$y = Wx + b \tag{2.8}$$

where x is the input vector of size d, W is the weight matrix of size $n \times d$, where n is the number of output neurons, b is the bias vector of size n, item y is the output vector of size n.

To introduce non-linearity, an activation function $f(\cdot)$ is applied element-wise to the output [40]:

$$y = f(Wx + b) \tag{2.9}$$

Common activation functions include:

- ReLU (Rectified Linear Unit): $f(x) = \max(0, x)$
- Sigmoid: $f(x) = \frac{1}{1+e^{-x}}$ (used for binary classification)
- Softmax: Used in the final FC layer for multi-class classification, defined as:

$$\text{Softmax}(y_i) = \frac{e^{y_i}}{\sum_{j=1}^{n} e^{y_j}} \tag{2.10}$$

2.7 Computer Vision

Computer Vision (CV), a field of artificial intelligence, aims to replicate human visual perception by enabling machines to process, analyze, and interpret digital images and videos. This field encompasses various tasks, including object detection, image classification, image segmentation, and pattern recognition, allowing automated systems to extract meaningful insights and make data-driven decisions [41].

2.7.1 Object detection

Object detection is a fundamental task in CV, that involves both localization and classification of objects within an image or video by using bounding boxes or segmentation masks to define the ROI. Object detection can be divided into two main tasks [30]:

Localization – Determining the position of an object within the image using bounding boxes or segmentation masks.

Classification – Assigning the detected object to a predefined category.

In the last years, with the advancements in DL and the increase of computing power GPUs, the accuracy and efficiency of object detection algorithms have improved. In the field of medical imaging, DL-based object detection models play a crucial role in identifying and localizing anomalies such as tumors, fractures, and wounds, facilitating early detection, diagnosis, and treatment planning of various diseases [30].

Among deep learning-based object detection approaches, models are generally categorized into one-stage detectors and two-stage detectors, each with specific advantages and trade-offs [30].

1. **One-Stage Detectors** - Predict bounding boxes and class labels in a single forward pass, optimizing detection speed and making them ideal for real-time applications. These models eliminate the need for a separate region proposal step, allowing for faster inference but potentially lower accuracy when detecting small or irregularly shaped objects.

2. **Two-stage detection** - Follow a multi-step process, where potential object regions are first proposed and then classified, followed by bounding box regression. These models achieve higher precision but suffer from longer inference times.

The choice between two-stage and one-stage object detectors depends on the specific application requirement. Although accurate, two-stage detectors require significant computational resources and long inference times, making them less suitable for real-time applications. On the other hand, one-stage detectors, such as YOLO framework, known for its ability to simultaneously predict bounding boxes and class probabilities, has stood out for its remarkable balance of speed and accuracy. Notably, recent YOLO versions have reached accuracy levels comparable to two-stage methods, making it one of the most widely used object detection models in medical imaging.

2.7.2 YOLO

The YOLO (You Only Look Once) architecture, developed by Joseph Redmon et al. and published at CVPR 2016, offered a paradigm shift in efficiency and accuracy by presenting for the first time a real-time end-to-end approach for object detection [42]. YOLO employs a fully CNN to process the entire image in a single forward pass, extracting spatial features and making predictions in a structured manner. The network consists of 24 convolutional layers for feature extraction, 4 max-pooling layers for spatial downsampling, and 2 fully connected layers for final bounding box regression and class probability prediction, as shown in Figure 2.4 [42]. YOLO's detection process, as illustrated in Figure 2.5, follows a structured pipeline to efficiently predict bounding boxes, confidence scores, and class probabilities [43]:

1. Image Grid Division – The input image is split into an $S \times S$ grid.

2. Bounding Box Prediction – Each grid cell predicts B bounding boxes and their confidence scores.

3. Class Probability Prediction – Each grid cell assigns probabilities to C predefined object classes.

4. Confidence Score Calculation – The confidence score for each predicted bounding box is computed as:

$$S_{\text{conf}} = \Pr(\text{Object}) \times \text{IOU}_{\text{truth, pred}} \tag{2.11}$$

   where Pr(Object) represents the probability that an object is present within the grid cell and $\text{IOU}_{\text{truth, pred}}$ (Intersection over Union) measures the overlap between the predicted bounding box and the ground truth bounding box.

5. Class-Specific Confidence Score Calculation – The final confidence score for a bounding box is obtained by multiplying the confidence score with the class probability:

$$CS_{\text{conf}} = \Pr(\text{Object}) \times \text{IOU}_{\text{truth, pred}} \times \Pr(\text{Class}_i|\text{Object}) \tag{2.12}$$

6. Non-Maximum Suppression (NMS) – Applied to remove redundant detections and retain only the most confident bounding boxes.

Over the years, YOLO has undergone massive architectural changes, resulting in substantial improvements in detection accuracy, speed, and robustness. From YOLOv1 to YOLOv11 have been many changes, including architectural design, training strategies, and optimization techniques [43].

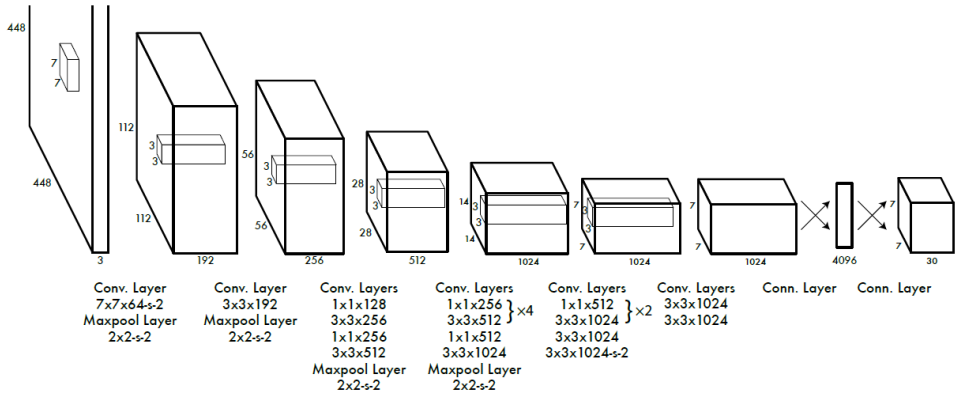


Figure 2.4: YOLOv1 Architecture [42]

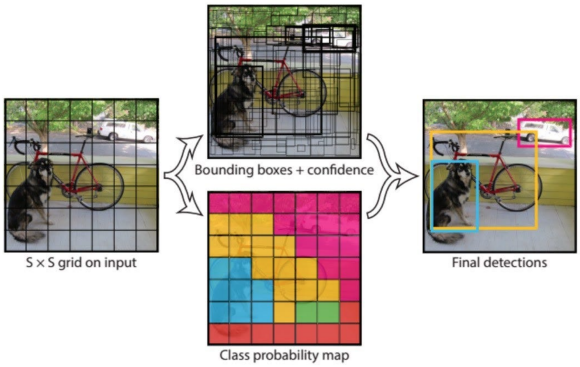


Figure 2.5: YOLO process for object detection [42]

2.8 Evaluation Metrics

To evaluate the performance of AI algorithms, various evaluation metrics are used. These metrics are calculated using the following measures [44]:

- True Positives (TPs) – number of positive results classified correctly.
- False Positives (FPs) – number of negative results classified incorrectly as positive.
- True Negatives (TNs) – number of negative results classified correctly.
- False Negatives (FNs) – number of positive results classified incorrectly as negative.

In this work, standard object detection metrics were employed, including mean Average Precision (mAP), Intersection over Union (IoU), Dice Coefficient, Precision, Recall, and F1-score. These metrics are presented in Table 2.1.

The Average Precision (AP), traditionally called mean Average Precision (mAP), is the commonly used metric for evaluating the performance of object detection models. It is computed by averaging the AP values across all object categories, ensuring a fair evaluation of model performance across multiple classes. The AP for each category is computed by integrating the Precision-Recall curve [43]:

$$AP = \int_0^1 P(R), dR \tag{2.13}$$

Finally, mAP is calculated as the mean of the AP values across all object categories [43]:

$$mAP = \frac{1}{N} \sum_{i=1}^{N} AP_i \tag{2.14}$$

where N is the total number of object categories.

To ensure accurate localization, mAP incorporates the IoU metric. A detection is considered positive if its IoU exceeds a predefined threshold (e.g., IoU ≥ 0.5), that determines how strict the model must be when aligning predicted bounding boxes with ground truth objects. By computing AP for each category and aggregating the results, mAP provides a balanced measure of how well the model performs across different object types and varying degrees of localization accuracy [43].

Table 2.1: Evaluation metrics for AI algorithms [44][45][46]

Metric	**Formula**	**Meaning**
Intersection over Union (IoU)	$\frac{TP}{TP+FP+FN}$	Ratio between the predicted region and the ground-truth region
Dice Coefficient	$\frac{2TP}{2TP+FP+FN}$	Ratio of overlap between the predicted and true positive regions, similar to IoU but weighted for overlap
Precision	$\frac{TP}{TP+FP}$	Proportion of positive predictions that are actually correct
Recall	$\frac{TP}{TP+FN}$	Proportion of actual positives that are correctly identified
F1 Score	$\frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision}+\text{Recall}}$	Harmonic mean of precision and recall

3

Literature Review

In this chapter, relevant studies of medical object detection and AI for the diagnosis of chronic wounds are presented.

3.1 Medical Object Detection

Object detection is a fundamental task in CV, and its applications in medical imaging have gained significant attention in recent years, since the precise localization and detection of anomalies is crucial for early diagnosis and treatment planning. Among the various DL models, the YOLO has been widely used due to its efficiency, speed, and accuracy.

One of the primary applications of YOLO in medical imaging is tumor and lesion detection. Studies have applied YOLO to detect lung nodules on chest radiographs, brain tumors on magnetic resonance imaging, and skin lesions on dermoscopic images [47]. For example, a study comparing YOLOv3 with Faster R-CNN for lung nodule detection found that YOLO achieved a higher inference speed with comparable accuracy, making it more suitable for real-time diagnostic applications [47]. Medical object detection is not limited to disease identification but extends to organ and tissue segmentation, essential for surgical planning and intervention. Studies have used YOLOv4 and YOLOv7 for automated kidney, liver, and lung segmentation in CT and MRI scans, significantly reducing manual annotation efforts [47]. Cascella *et al.*, [48], presented YOLOv8 model to analyze facial expressions indicative of pain, obtaning a mAP of 0.893 and a precision for pain and non-pain of 0.868 and 0.919, respectively. In the context of wound detection, YOLO-based models have been increasingly explored to automatically detect chronic wounds, pressure ulcers, and diabetic foot ulcers (DFUs) [49] [50] [51].

3.2 AI for diagnostic of Chronic Wounds

Chronic wounds pose a significant burden on healthcare systems, affecting millions of individuals worldwide. AI-based methods are transforming the landscape of chronic wound diagnosis by offering automated and objective assessment tools [52].

Early AI-driven wound classification relied on traditional ML techniques, where feature extraction played a crucial role. For example, Suvarna *et al.*, [53], used ML algorithms, Support Vector Machine (SVM) and k-Nearest Neighbors (kNN), to classify the scalding burns into different categories. Color and texture features were extracted from the LAB color space and the classifier with the best results was SVM.

Recent studies have combined deep learning with traditional ML classifiers to predict the likelihood of infection based on wound characteristics. In a previous master's thesis conducted within the scope of the CardioFollow.AI project, Pereira *et al.*, [5], developed a deep learning segmentation model called MobileNet-UNet, capable of identifying the specific area of a wound and classifying it into one of three categories: chest, drain or leg. The segmentation model obtained a mAP of 90.1% and a IoU of 89.9%. Furthermore, to predict the likelihood of wound alterations for each respective category, different algorithms (SVM, kNN, and Random Forest) were proposed. The utilization of distinct classifiers for the final classification yielded greater efficacy compared to employing a single classifier for all different kinds of wounds. The classifier for leg wounds demonstrated superior performance, achieving an 87.6% recall rate and 52.6% precision rate.

Additionally, CNN-based models are widely used in medical imaging tasks due to their ability to extract hierarchical features, making them suitable for wound segmentation. Zahia et al., [54], focused on pressure injury segmentation, utilizing a nine-layer CNN to differentiate between granulation, slough, and necrotic tissues. Their model achieved an overall accuracy of 92.01% and a weighted Dice Similarity Coefficient of 91.38%, demonstrating its effectiveness in distinguishing tissue structures.

Ramachandram et al., [55], introduced a DL approach leveraging an EfficientNetB0-based encoder-decoder architecture to segment four key wound tissue types: epithelial, granulation, slough, and eschar. By employing a fully convolutional network structure, the model effectively captured fine-grained wound details and the segmentation model demonstrated high IoU scores, particularly excelling in the segmentation of slough (F1-score = 73.1%) and eschar (F1-score = 80.2%), though it exhibited lower performance for epithelial tissue (F1-score = 25.3%) due to its indistinct boundaries and variation in appearance.

A significant challenge in wound image classification is the limited availability of annotated data, which makes DL models difficult to train. One-shot learning techniques, as YOLO, offer a potential solution by requiring fewer labeled examples while maintaining high accuracy. For example, Anisuzzaman *et al.*, [49], integrated a lighter version of YOLOv3 into an iOS mobile application for wound detection in video processing scenarios. The YOLOv3 model was trained and evaluated on a proprietary dataset achieving a mAP of 93.9%. Furthermore, Aldughayfiq *et al.*, [50], presented a novel approach that used YOLOv5 to detect and classify pressure ulcers into four stages and non-pressure ulcers, achieving an overall mAP of 76.9%, while Sendilraj *et al.*, [51], used YOLOv5s to localize, classify, and analyze Diabetic foot ulcers with an achieved F1-score of 80% and a mAP of 86.1% for wound localization and a binary accuracy of 79.8% for wound infection. Tusar *et*

al., [56], train YOLOv8 to detect and stage pressure injuries, with YOLOv8s, with AdamW optimizer and hyperparameter tuning, achieving a mAP of 84.16% and a recall of 82.31% .

In summary, while previous studies have shown the effectiveness of AI in wound detection and classification, few address both infection risk and wound evolution over time. Additionally, there is limited research on using these techniques in real-world telemonitoring settings, especially for postoperative patients recovering at home. This thesis proposes an end-to-end system using YOLOv8 to detect surgical wounds, classify infection risk, and track wound progression over time in a telehealth context.

4

Methodology

This chapter outlines the methodology used to develop and implement the prediction model using the established dataset.

4.1 Dataset

The dataset comprises surgical site RGB photographs collected from 110 patients who underwent cardio-thoracic surgery at Hospital de Santa Marta. This work builds upon a previous dataset initially composed of 1,337 images from 34 patients, covering a follow-up period of 30 days. As part of this dissertation, the dataset was significantly expanded both in scale and scope, now including 7,867 additional photographs and extending the observation period to up to 90 days post-surgery.

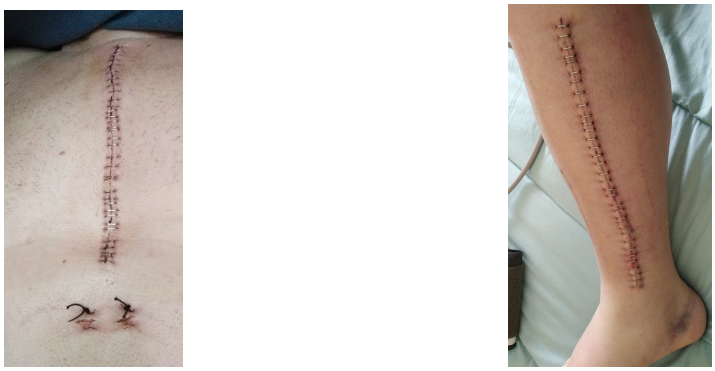
The dataset contains photographs of individuals aged between 24 and 83 years, with the age group of 51 to 65 being the most represented one. Regarding gender distribution, among the 110 patients, 75 are male, while the remaining are female.

The photographs document the evolution of each patient's surgical wound over either a 30- or 90-day follow-up period. For patients under the 30-day protocol, a daily photograph of the surgical site is captured, either by the patient itself or a family member, and submitted through the telemonitoring platform for clinical evaluation. Regarding the newly enrolled 76 patients, the follow-up methodology remains the same during the first month, while photographs are collected on a weekly basis in the subsequent two months.

All images were captured using the front or back camera of a Xiaomi Mi A2 Lite smartphone, resulting in variations in image resolution. Because there was no standardized acquisition protocol, a broad range of conditions was observed, including variations in lighting, patient positioning, orientation, and background. Consequently, to ensure a minimum quality threshold within the dataset, duplicates, extremely low-quality images, and unrelated photographs were excluded. After this cleaning process, the dataset corresponding to the additional 76 patients was reduced from 7,867 to 3,948 images.

The dataset can be categorized by wound type: Chest Wound (CW), Drainage Wound (DW), and Leg Wound (LW). Figure 4.1 illustrates one example of each category: (a) a

midline CW resulting from sternotomy and a DW, and (b) a LW from the saphenous vein graft. After the cleaning process, a total of 2966 CW, 2282 DW and 900 LW incorporated the dataset. The higher total number of annotated wounds compared to the number of images is explained by the fact that some photographs contain more than one wound region, reflecting real clinical scenarios where multiple surgical sites appear in the same image.



(a) (b)

Figure 4.1: Example of Wound's types (a) Chest Wound (above) and Drainage Wound (below) (b) Leg Wound

Alternatively, the dataset can be used to define a classification task related to infection risk, where class 0 corresponds to photographs without alarming signs, and class 1 corresponds to photographs showing visible alarming signs of potential infection. Figure 4.2 shows one example of each: (a) a wound with no signs of infection and (b) a wound with evident inflammatory characteristics.

(a) (b)

Figure 4.2: Example of the evolution of a patient's LW (a) The photograph does not show any alarming alteration (b) The photograph shows the present of redness and swelling - alarming alterations

Despite the relatively large dataset, the alarming photographs are only 123, representing just 2.3% of the dataset, resulting in a highly imbalanced class distribution. Most of the

collected images capture the normal healing progression of similar or identical wounds over time, which means the dataset is clearly unbalanced and may pose significant challenges for AI applications, particularly in training and evaluating algorithms focused on detecting and classifying critical wound anomalies.

4.2 Annotation and Labeling

For image segmentation, a graphical annotation tool for manually labeling was employed. LabelMe is an open-source image annotation tool widely used in CV for tasks such as object detection, segmentation, and classification [57].

In this study, all images of the dataset (5,285 images) were uploaded in the software, since the masks of the initial 1337 images were not available, and each ROI was precisely delineated by drawing simple polygons around the sutured area, Figure 4.3. Once the annotations were finalized, Labelme automatically generates a JSON file for each image, containing the mask coordinates and corresponding wound type classification.

Figure 4.3: Example of a ROI delineation on LabelMe

These annotations were performed manually to guarantee a greater control and accuracy over automated tools, which often struggle with variations in lighting, patient positioning, and other factors common in images of real-world surgical sites. This way, more reliable annotations were generated providing a certain quality and facilitating better model training and predictive performance.

Although an infected wound typically exhibits specific local signs as redness, exudate, necrotic tissue, wound edge separation, and swelling, definitive diagnosis cannot be made solely on the basis of visual inspection and requires expert clinical confirmation. Consequently, for data annotation purposes, a classification based on infection risk was used by examining the presence or absence of these visible indicators. Each wound was labeled either as “Yes” or "No", indicating alarming alterations and potential infection or indicating normal wound healing, respectively. These annotations were done in a CSV file with relevant information about the patients, and subsequently validated by the clinical team of Hospital de Santa Marta.

4.3 Proposed Pipeline

The proposed approach consists of two stages aimed at detecting ROI and classifying them based on wound type and risk infection. The process begins with the cleaned dataset and the pipeline follows two distinct workflows, Figure 4.4.

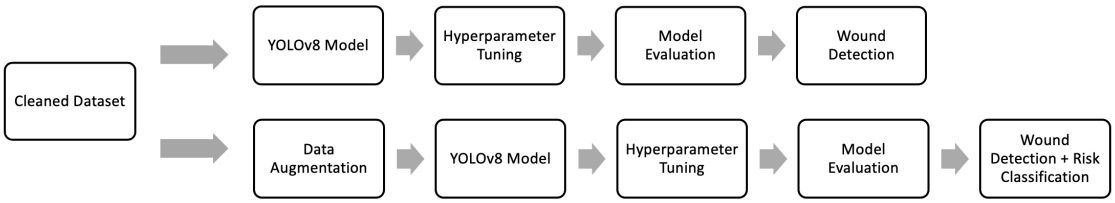


Figure 4.4: Proposed Pipeline

1. Wound Detection: This approach employs an object detection model to localize the wound region and classify it into one of three predefined categories: CW, DW, LW;

2. Wound Detection and Infection Risk Classification: In this second approach, the model is extended to predict the infection risk associated with each detected wound.

In both approaches, the YOLOv8 model was employed to detect wound regions and, when applicable, classify the infection risk. The infection risk classification strategy is evaluated under four experimental conditions, combining two key factors: with and without data augmentation, and binary (risk of infection (1) or not at risk (0)) versus multi-class classification (predicts both wound type and infection risk, resulting in six possible classes:$CW_0, CW_1, DW_0, DW_1, LW_0, LW_1$).

To further enhance detection accuracy and classification reliability, the models underwent hyperparameter optimization, fine-tuning parameters such as learning rate, batch size, weight decay, and the number of training epochs. This optimization ensured that the YOLOv8 model achieves a balance between precision, recall, and computational efficiency, improving its robustness across different experimental conditions. This structured pipeline establishes a scalable DL-based framework for automated wound assessment, facilitating early detection, classification, and infection risk evaluation.

4.4 YOLOv8 Implementation

Given the wide range of object detection models available, the choice of architecture plays a crucial role in balancing performance and computational efficiency. In this study, YOLOv8 was selected over newer versions (YOLOv9, YOLOv10, and YOLOv11) after exploratory testing showed that it provided more stable performance. YOLOv8, released by Ultralytics in 2023, is a state-of-the-art object detection model built upon its predecessors, that introduces architectural enhancements that improve performance across various CV tasks. The YOLOv8 architecture consists of 3 blocks: Backbone, Neck and Head, as shown in figure 4.5 [58].

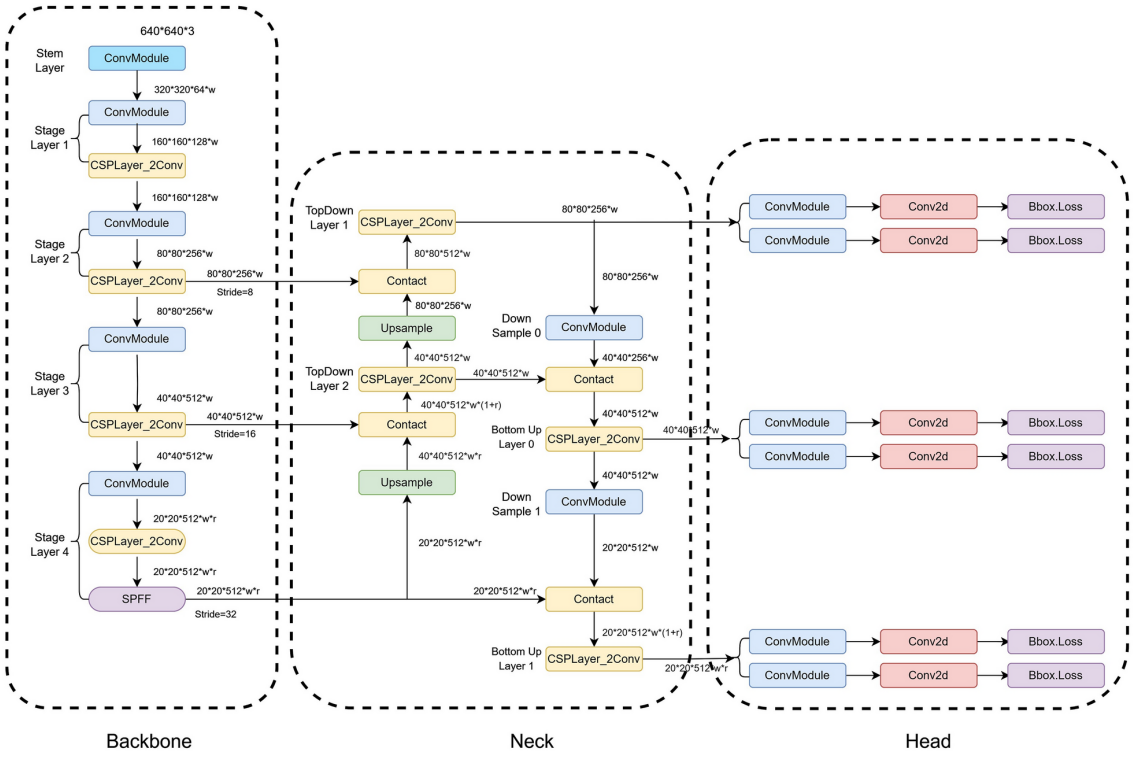


Figure 4.5: Model Structure of YOLOv8 [59]

The backbone block is based on CSPDarknet53 and is responsible for extracting features from the input image. The backbone uses a cross-stage partial architecture to split the feature map into two components, where the first undergoes convolutional operations, and the second is concatenated with the processed output from the first component, enhancing gradient flow and reducing computational cost. The backbone employs a series of 3×3 convolutions interspersed with bottleneck blocks, striking a balance between features and parameter efficiency [60].

The Neck block uses multi-scale feature fusion, Feature Pyramid Network and Path Aggregation Network, to combine features extracted by the backbone This process is critical to detect different sizes and scales of objects [60].

The Head block is responsible for predicting the final classifications. Yolov8 introduces an anchor-free approach, directly predicting an object's mid-point instead of using predefined anchor boxes, which simplifies the prediction process by reducing the number of hyperparameter and speeding up NMS, a pre-processing step that discards incorrect predictions. The head employs a decoupled architecture, separating the tasks of bounding box regression, class prediction, and confidence scoring [60].

To support different deployment needs, YOLOv8 offers several model variants, each with increasing complexity and computational cost. These are summarized in Table 4.1, which presents the number of parameters, the mean average precision (mAP) on the validation set, and the computational cost measured in GFLOPs. The data presented in this table is sourced from the official Ultralytics benchmarks, and was not computed using

this study's dataset.

Table 4.1: YOLOv8 Model Variants Comparison [58]

Variant	**Parameters (M)**	**mAPval**	**GFLOPs**
YOLOv8n	3.5	18.4	10.5
YOLOv8s	11.4	27.7	29.7
YOLOv8m	26.2	33.6	80.6
YOLOv8l	44.1	34.9	167.4
YOLOv8xl	68.7	36.3	260.6

These variants are defined by three key parameters [58]:

- Depth multiple: Determines the number of bottleneck blocks in the C2f module.
- Width multiple: Defines the number of channels in each layer.
- Maximum channels: Sets an upper limit on the number of channels, allowing dynamic adjustment of the network's capacity.

The YOLOv8s variant was selected for this study as it demonstrated the best balance between accuracy and computational efficiency during preliminary comparative testing on a small partition of the dataset. To train the model, annotations were created using the open-source tool LabelMe, which was used to manually define the ground truth regions of interest. These annotations were initially stored in JSON format and later converted to the YOLO polygon-compatible format. The conversion process involved:

1. Extracting masks coordinates and class labels from JSON files.
2. Transforming these annotations into the YOLO format, where each annotation is stored in a TXT file corresponding to the image. Each TXT file contains multiple lines, with each line representing an object as:

$$\text{class } x_1\ y_1\ x_2\ y_2\ \ldots\ x_n\ y_n \tag{4.1}$$

   where class is the object class ID (for example: CW = 0, DW = 1, LW = 2) and $x_1, y_1, x_2, y_2, \ldots, x_n, y_n$ are the normalized coordinates of each vertex of the polygon.
3. Linking the dataset to a YOLO dataset configuration file ('.yaml'), which specified the number of classes, the paths to the training, validation, and test datasets, and the class names and their corresponding IDs.

Since YOLOv8 is primarily designed to handle bounding boxes, a polygon-to-bounding-box transformation was applied when necessary, ensuring compatibility with the model. After training, YOLOv8 predicts a set of vertices defining object boundaries (polygon

predictions) and a class score for each detected object, indicating the confidence in classification. Only detections above a predefined Confidence Thresholding of 0.5 were retained. This threshold was empirically determined through validation experiments to balance sensitivity and specificity, ensuring the exclusion of low-confidence predictions while preserving relevant detections.

4.5 Data Augmentation

Data Augmentation is a technique used to artificially increase the size of the training dataset. In this work, to enhance dataset diversity and address class imbalance in infection risk,a series of augmentation techniques were applied to images and their associated annotations in the training set of the YOLOv8 classification model, specifically targeting high-risk cases.

The augmentations included adjustments to brightness, contrast, hue, and saturation to simulate varying lighting conditions. Rotations and flips were applied to account for arbitrary object orientations, while Gaussian blur was used to simulate low-quality image conditions. The augmentations were applied probabilistically (50% for most transformations), ensuring variability while retaining a representative dataset. Keypoint-based annotations (polygons) were transformed alongside the images, preserving their alignment and validity. These augmentations were applied five times to images labeled with risk of infection using the Albumentations library to improve their representation during training.

4.6 Training Process

To ensure robust model generalization, the dataset was partitioned into three subsets: 60% training set, 20% validation set, and 20% test set, enabling the model to learn meaningful patterns from the training set while leveraging the validation set for hyperparameter tuning and overfitting prevention. The test set was reserved exclusively for final performance evaluation on unseen data, ensuring an objective assessment of the model's generalization capability.

To address class imbalance and ensure reliable evaluation, the dataset was partitioned using stratified sampling, an approach which preserves the original distribution of classes across all subsets, preventing skewed representations that could bias model performance. The training set was further split into five folds using StratifiedKFold cross-validation from Scikit-Learn, ensuring consistent class distribution across all splits.

Model training was conducted on a 64-bit Ubuntu PC with an 8-core 3.4 GHz CPU and a single NVIDIA RTX 3090 GPU, ensuring efficient computational performance. Optuna, an automated hyperparameter optimization framework, was employed to explore the hyperparameter space efficiently. During each iteration of the cross-validation process, four folds were used for training and one for validation, with performance metrics averaged

across all folds to mitigate bias and improve reliability in hyperparameter selection. Once the optimal hyperparameters were determined, the model was re-trained on the full training and validation dataset, incorporating all available labeled data to maximize learning. The final evaluation was performed on the independent test set, which remained entirely unseen throughout training and validation, ensuring an unbiased assessment of the model's performance in wound detection and classification.

4.6.1 Optuna

Hyperparameter optimization, a fundamental process of DL training, directly impacts the model's convergence, stability, and accuracy. Traditional manual tuning methods such as grid search and random search are inefficient and computationally expensive, making them unsuitable for complex DL models like YOLOv8.

Optuna is a framework designed for efficient and automated hyperparameter optimization. It employs Bayesian optimization, early pruning, and adaptive search strategies to effectively explore the hyperparameter space and find the optimal settings with minimal computational cost [61]. In this work, Optuna was employed to maximize mAP (mean Average Precision). This way, for each hyperparameter combination, the model was trained and evaluated and the optimal hyperparameter combination was selected based on performance metrics. The search space was defined as follows [58]:

- Learning Rate (η): Determines the step size of weight updates during backpropagation. A high learning rate may lead to unstable convergence, whereas a low learning rate can result in slow training progress. To capture variations efficiently, the learning rate was optimized on a logarithmic scale:

$$\eta \sim \text{log-uniform}(10^{-5}, 10^{-1}) \tag{4.2}$$

- Batch Size (B): Defines the number of training samples processed per iteration. A smaller batch size provides better generalization at the cost of increased noise in gradient updates, while a larger batch size improves computational efficiency but may lead to overfitting. The batch size was chosen from discrete values:

$$B \in \{8, 16, 32\} \tag{4.3}$$

- Weight Decay (λ): A regularization term that prevents overfitting by penalizing large model weights. It influences the model's ability to generalize across unseen data. The weight decay parameter was optimized within the range:

$$\lambda \sim \text{log-uniform}(10^{-6}, 10^{-3}) \tag{4.4}$$

- Number of Epochs (E): Defines the number of complete training cycles over the dataset. Insufficient epochs can result in underfitting, whereas excessive may lead to overfitting. The number of epochs was optimized over a predefined range:

$$E \in [30, 60] \tag{4.5}$$

4.7 Model Evaluation

Given the dual-stage design of the proposed pipeline, evaluation strategies were tailored separately for the wound detection and risk classification tasks.

4.7.1 Wound Detection Evaluation

For the object detection component, which localizes and classifies wound regions, evaluation focused on spatial overlap and class prediction accuracy. The following standard metrics were employed:

- **Mean IoU** : Quantifies the overlap between predicted and ground truth bounding polygons, averaged across all predictions.
- **Dice Coefficient**: Measures spatial similarity, especially useful in medical image analysis where boundary precision is important.
- **Precision**: Evaluates the proportion of positive predictions that are actually correct.

4.7.2 Infection Risk Classification Evaluation

For infection risk classification, two experimental setups were evaluated: binary classification (risk vs. no risk) and multiclass classification (wound type + risk identification). To ensure consistent evaluation across these tasks, the following metrics were calculated:

- **Precision**: Proportion of true positives among predicted positives, indicating false positive control.
- **Recall**: Proportion of true positives among all actual positives, indicating false negative control.
- **F1-Score**: Harmonic mean of precision and recall, capturing the balance between both.
- **Cohen's Kappa**: Measures agreement between prediction and ground truth labels, adjusted for random chance, and specially relevant for imbalanced class distributions. It is defined as:

$$\kappa = \frac{p_o - p_e}{1 - p_e} \tag{4.6}$$

where p_o is the empirical probability that both annotators assign the same label to a given sample and p_e the probability of agreement by chance, estimated based on the empirical distribution of class labels assigned by each annotator.

F1-score and Cohen's kappa are particularly important when evaluating classifiers on imbalanced datasets, such as those in medical imaging, where the number of infection cases is often much smaller than non-infection cases, offering a more informative and fair assessment of model performance than overall accuracy alone. All metrics were calculated using metrics scikit-learn functions to ensure consistency and reproducibility.

4.8 Wound Healing Analysis

To evaluate wound healing progression and predict the risk of infection over time, a temporal analysis framework was developed, providing both visual and quantitative insights into the wound's evolution following hospital discharge. This analysis was based on confidence scores generated by the YOLOv8-based binary wound classification model which categorized each detected wound region as no risk (class 0) or at risk (class 1) of infection. To place these predictions on a temporal scale, the number of days since hospital discharge was calculated for each image. Discharge dates were retrieved from the CSV file containing patient metadata, while the acquisition dates of wound photographs were extracted from the filenames. A mapping file was subsequently generated, linking each image to its corresponding time point in days since discharge. Then, two different temporal analyses were performed:

1. **Patients Without Infection Risk**: For patients whose images were consistently classified as no risk, all infection probabilities were plotted on the same figure. Each patient was represented by a separate line, showing how their predicted risk remained stable or changed across the follow-up period.

2. **Patients With Infection Risk**: For patients with at least one image classified as at risk, the visualization was restricted to the 7 days preceding the first predicted infection. All such patients were grouped into a single plot, with each line representing a patient's risk trend in the week leading up to the first high-risk prediction.

This dual analysis approach allowed visual differentiation between stable, low-risk cases and those that developed risk over time, allowing the visualization of wound progression.

5

Results and Discussion

This chapter reports and analyzes the results obtained, and is divided into three sub-chapters taking into account the different steps of the study.

5.1 Wound Detection Model

As mention in section 4.3, YOLOv8 was applied to detect the wound region and assign a label to the type of wound. To enhance detection accuracy and classification reliability, the models underwent a hyperparameter optimization, where key parameters such as number of training epochs, batch size, learning rate (lr0) and weight decay were fine-tuned with the help of Optuna. A total of five trials were conducted for both the initial dataset and the expanded dataset (Table I.1 and I.2 of Annex I). The final reported results, as presented in Table 5.1, correspond to the best-performing model configuration obtained after the completion of all trials, while Table 5.2 summarizes the optimal hyperparameters identified for each dataset.

Table 5.1: Comparison of wound detection model performance on different sizes of the dataset

Dataset	**Mean IoU**	**Dice Coefficient**	**Precision**
Initial	93.9%	96.9%	98.6%
Expanded	94.7%	97.3%	98.0%
Previous Results [5]	89.9%	94.6%	90.1%

Table 5.2: Best hyperparameter configuration found for each dataset size

Dataset	**Epochs**	**Batch Size**	**lr0**	**Weight Decay**
Initial	42	16	0.0038	0.00016
Expanded	40	8	0.01	0.0005

The evaluation of the wound detection model was based on key performance metrics, including mean Intersection over Union (IoU), Dice Coefficient, and Precision. The results

presented in Table 5.1 compare the model's performance on two datasets (Initial and Expanded) and also provide a reference to previously reported results.

The results indicate a consistent improvement when transitioning from the Initial dataset to the Expanded dataset:

- Mean IoU increased from 93.9% to 94.7%, suggesting a more accurate localization of the wound regions.
- Dice Coefficient improved from 96.9% to 97.3% reinforcing the consistency of segmented wound regions compared to ground truth.
- Precision showed a slight variation from 98.6% to 98.0%, maintaining high confidence in correctly classified detections.

The increase in IoU and Dice Coefficient suggests that the model benefits from a larger and more diverse dataset, leading to better generalization and more accurate wound segmentation. On the other hand, when compared the previous results (trained on the original dataset) with the new ones, the model achieves notable improvements:

- Mean IoU improved from 89.9% to 93.9%, demonstrating a significant enhancement in the model's ability to accurately localize wound boundaries.
- Dice Coefficient increased from 94.6% to 96.9%, highlighting a better overlap between predicted and actual wound regions.
- Precision improved from 90.1% to approximately 98.0%, reflecting a lower false positive rate and more reliable predictions.

These improvements indicate that the employed model and hyperparameter tuning have played a crucial role in optimizing wound detection accuracy.

In general, the model's high Dice Coefficient and IoU suggest that it is highly reliable for wound segmentation, which is critical for automated wound monitoring systems. The Precision ensures that detected wounds are less prone to false positives, while the minor variation in Precision between the initial and expanded datasets suggests that additional data does not compromise classification reliability, but rather improves segmentation accuracy.

5.2 Wound Detection and Risk Classification Model

As mention in section 4.3, YOLOv8 was also applied to detect the wound region and classify the risk of infection. Two classification strategies were considered: a binary classification, distinguishing between no risk (class 0) and at risk (class 1), and a multiclass classification, where the model predicts both the type of wound and the risk of infection simultaneously, involving six classes: $CW_0, CW_1, DW_0, DW_1, LW_0, LW_1$. Each

classification strategy was also evaluated with and without data augmentation, resulting in four experimental conditions: Binary – No Augmentation; Binary – With Augmentation; Multiclass – No Augmentation, Multiclass – With Augmentation. For each experimental conditions, the YOLOV8 model underwent a hyperparameter optimization similar to the Wound Detection Model, with the help of Optuna, and for each a total of five trials were conducted (Table I.3, I.4, I.5 and I.6 of Annex I).

Table 5.3 summarizes the performance metrics for each configuration's best hyperparameter, including precision, recall, F1-score and kappa score, while Table 5.4 summarizes the optimal hyperparameters identified for each experimental condition.

Table 5.3: Classification performance metrics across different experimental configurations

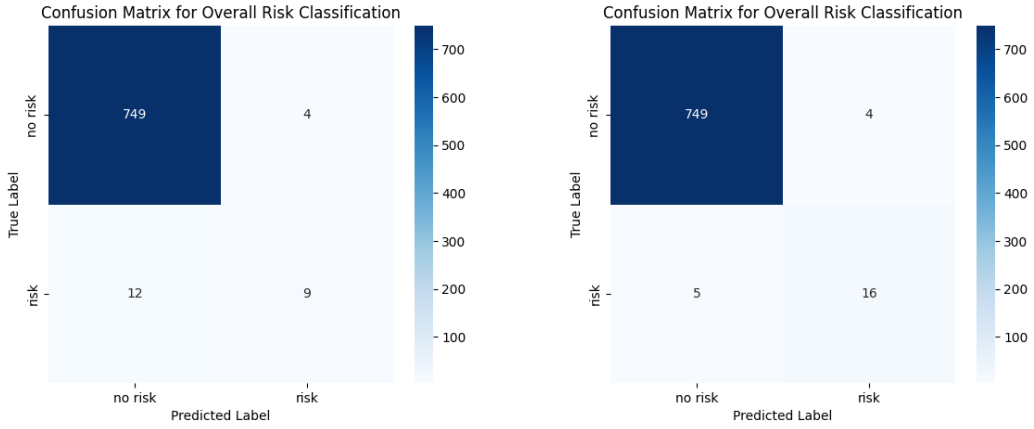
Configuration	**Precision**	**Recall**	**F1-Score**	**Kappa**
Binary – No Augmentation	97.6%	97.9%	97.7%	0.519
Binary – With Augmentation	98.8%	98.8%	98.8%	0.775
Multiclass – No Augmentation	95.1%	95.5%	94.6%	0.892
Multiclass – With Augmentation	95.5%	95.9%	95.7%	0.902

Table 5.4: Best hyperparameter configuration found for each experimental condition

Configuration	**Epochs**	**Batch Size**	**lr0**	**Weight Decay**
Binary – No Augmentation	60	16	0.0032	0.000022
Binary – With Augmentation	55	32	0.0032	0.0000014
Multiclass – No Augmentation	55	16	0.00025	0.00067
Multiclass – With Augmentation	60	8	0.000051	0.00047

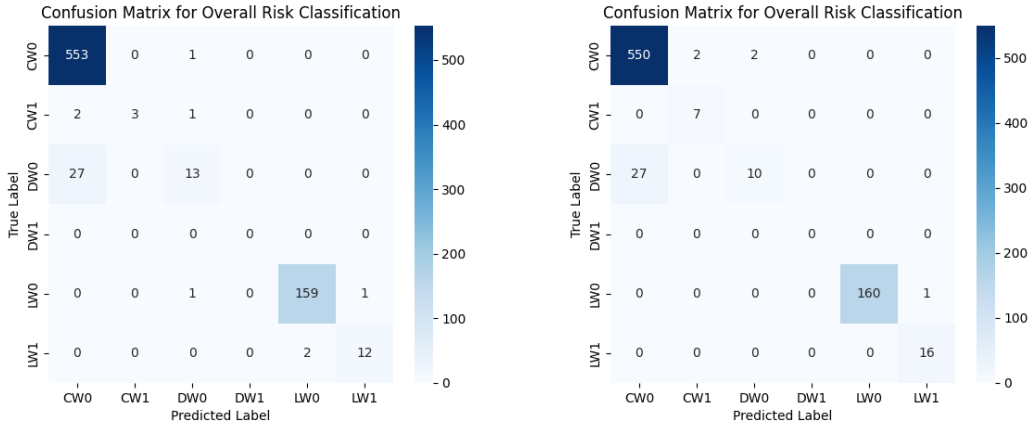
The results presented in Table 5.3 show that the binary classification models achieved superior performance across most evaluation metrics when compared to the multiclass classification ones, suggesting that the binary formulation of the problem is easier for the model to optimize and more robust to limited data variability. The best results were observed under the augmented condition (F1-score of 98.8%, kappa of 0.775) and the corresponding confusion matrices are shown in Figure 5.1, where a reduction in false negatives is clearly observable following augmentation. Notably, the number of missed high-risk cases (false negatives) decreased from 12 to 5, an important improvement in the clinical context, where failing to detect infection risk may lead to delayed intervention.

In contrast, multiclass classification provides more detailed predictions, associating each wound not only with risk status but also with wound type. While this complexity introduces more room for misclassifications, the performance remained strong. The confusion matrix analysis, Figure 5.2, shows that the augmentation of the data led to increased correct predictions for rare classes, with correct predictions for CW_1 increasing from 3 to 7 and LW_1 increasing from 12 to 16 after the augmentation.



(a) (b)

Figure 5.1: Confusion matrices for binary risk classification. (a): Without augmentation. (b): With data augmentation.



(a) (b)

Figure 5.2: Confusion matrices for multiclass risk classification. (a): Without augmentation. (b): With data augmentation.

The Cohen's kappa coefficient was substantially higher for multiclass classification in both settings (0.892 and 0.902, respectively), indicating that, despite lower absolute scores in some standard metrics, the multiclass models demonstrate stronger agreement between predicted and actual labels after adjusting for class imbalance and chance-level performance. This is particularly meaningful in real-world clinical settings, where such agreement is crucial to ensure consistent and reliable decision-making.

Across both classification strategies, data augmentation had a positive impact on all evaluated metrics. The binary model improved from an F1-score of 97.7% to 98.8%, while its kappa score increased significantly from 0.519 to 0.775. A similar trend was observed in the multiclass setting, where F1-score improved from 94.6% to 95.0%, and kappa score from 0.892 to 0.902. These improvements can be attributed to the increased variability and

diversity in the training data introduced through augmentation, which likely helped the models generalize better and avoid overfitting.

Moreover, the optimized hyperparameters in Table 5.4 reflect subtle adjustments needed to balance learning stability and generalization. Notably, the multiclass models, particularly with augmentation, required lower learning rates and smaller batch sizes, suggesting greater sensitivity to overfitting and the need for fine-grained updates during training. This sensitivity may stem from the increased number of classes and the associated complexity of learning both wound type and infection risk simultaneously.

From a clinical perspective, the binary model offers a high-performing and relatively simple approach to screening wounds for infection risk, being well suited to scenarios requiring quick, low-complexity assessments. However, the multiclass model presents a more detailed and informative output, enabling not only risk stratification but also wound-type-specific insights. Given its strong agreement metrics, particularly when supported by data augmentation, the multiclass strategy holds promise for integration into more comprehensive wound assessment pipelines, where such detailed predictions can guide tailored treatment plans.

5.2.1 Comparison between dataset sizes

Additionally, the binary model was trained on the initial dataset without data augmentation (Table I.7 of Annex I), with the best hyperparameters achieving a precision of 94.7%, recall of 94.9%, F1-score of 94.7%, and Cohen's kappa of 0.717, as shown in Table 5.5. While slightly lower than the tuned binary model with augmentation, these results indicate that the original dataset already contained meaningful signal, and that performance can be enhanced further through augmentation and hyperparameter tuning.

Table 5.5: Performance metrics of Binary Classification on different sizes of the dataset

Dataset	**Precision**	**Recall**	**F1-Score**	**Kappa**
Initial	94.7%	94.9%	94.7%	0.717
Expanded	97.6%	97.9%	97.7%	0.519

5.2.2 Comparison between different cross validation strategies

To ensure the robustness and clinical applicability of the proposed infection risk classification models, a patient-wise cross-validation strategy was employed (Table I.8 of Annex I). In this setting, images from the same patient were never split between training and testing sets, ensuring that model evaluation reflects generalization to unseen individuals, rather than memorization of patient-specific visual features. The best results under patient-wise cross-validation, Table 5.6, showed a moderate drop in performance compared to the normal cross validation. Despite these, the performance remained consistently high, confirming that the model's predictive capabilities extend beyond the patients seen during training.

Table 5.6: Performance metrics across different cross validation strategies on Binary Classification

Configuration	Precision	Recall	F1-Score	Kappa
Cross Validation	97.6%	97.9%	97.7%	0.519
Cross Validation by patients	82.7%	78.3%	80.4%	0.600

This form of cross-validation better simulates real-world deployment, where a model is applied to new patients whose wounds it has never encountered. Therefore, the use of patient-wise validation provides a more trustworthy estimate of how the system would perform in clinical settings and is strongly recommended for future research in this domain.

5.3 Wound Temporal Analysis

To evaluate wound healing progression a temporal analysis framework was developed, offering insights into wound healing progression and the emergence of infection risk following hospital discharge. By leveraging a YOLOv8-based binary wound classification model, which assigns each detected wound region a probability of being at risk or not, it was possible to assess patient-specific trends over time. Each patient's image was mapped to a time-point expressed as days since discharge which allowed a longitudinal tracking of infection risk, enabling the visual comparison between stable patients and those who developed infection indicators.

As shown in Figure 5.3, patients with no signs of infection maintained consistently low predicted probabilities throughout the follow-up period. This consistency suggests that the model successfully captures visual indicators of stable wound healing, offering a potential tool for remote monitoring of recovery.

In contrast, patients who were eventually classified as having a wound at risk exhibited more variability and higher peak probabilities, as observed in Figure 5.4. For some individuals, risk scores began rising weeks in advance, suggesting that the model may be sensitive to early pathological changes not yet evident clinically.

To further investigate this trend, a focused 7-day window preceding the first high-risk classification was analyzed. Figure 5.5 reveals that while some patients exhibit relatively stable risk probabilities during this period, a few show a mild upward trends, approaching the 0.4 infection probability. However, across most cases, there is no strong or uniform increase in infection probability leading up to the event, which suggests that while the model may capture subtle visual deterioration in some wounds, the predictive signal remains modest and may benefit from additional context or modalities, such as wound size progression or clinical symptoms, to improve early detection accuracy.

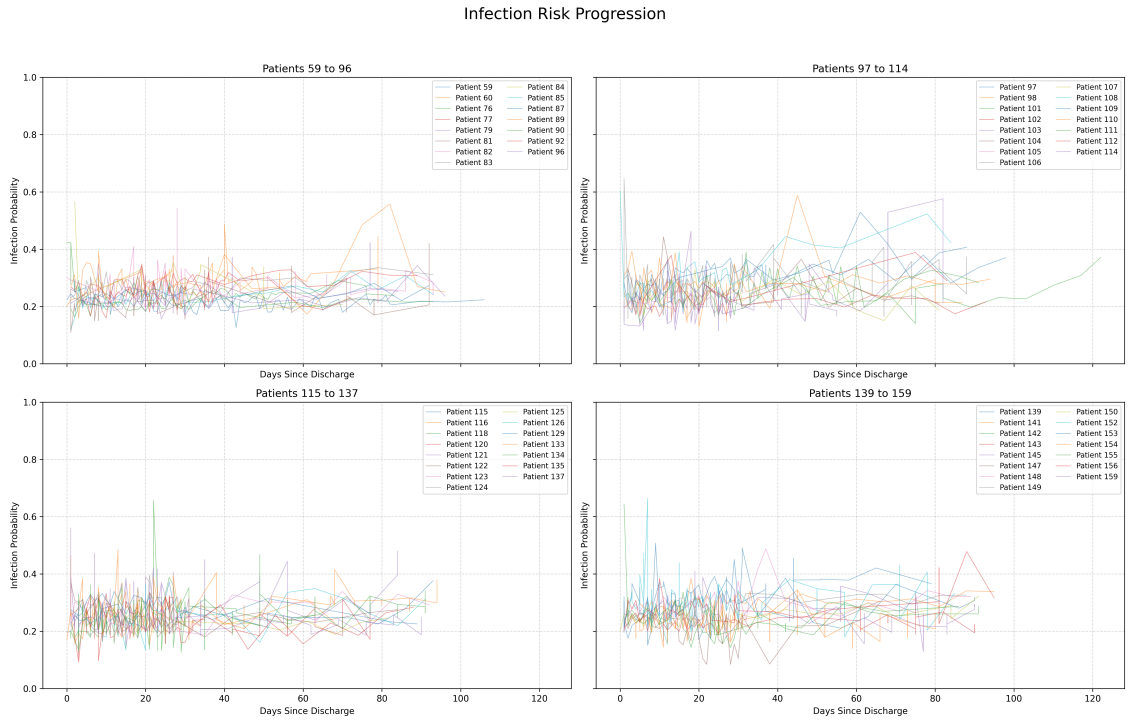


Figure 5.3: Infection risk probability trends for patients without infection risk across the follow-up period. Each subplot covers a group of patients for better visualization.

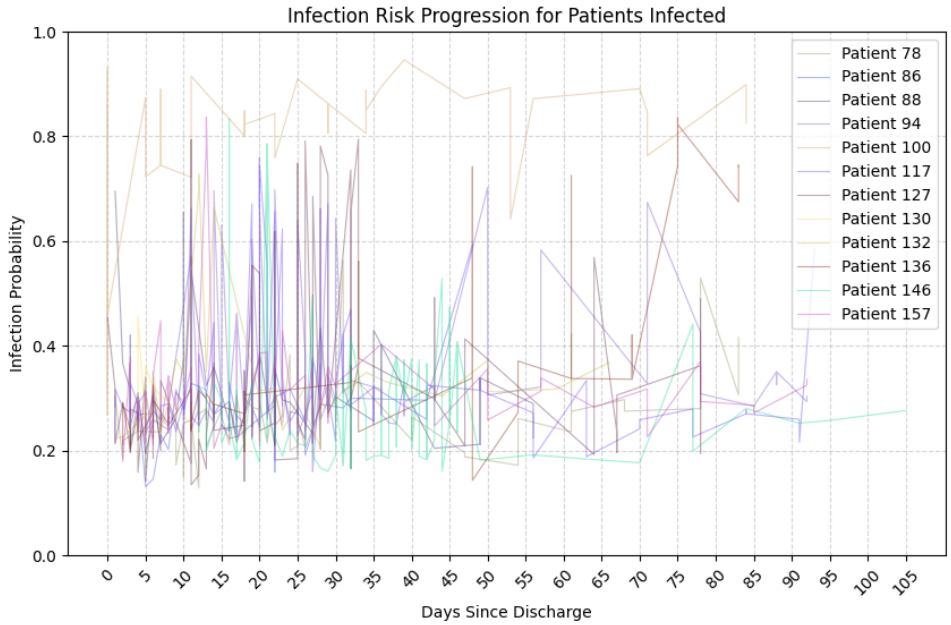


Figure 5.4: Infection risk probability progression for patients who eventually developed signs of infection, shown over the full monitoring period.

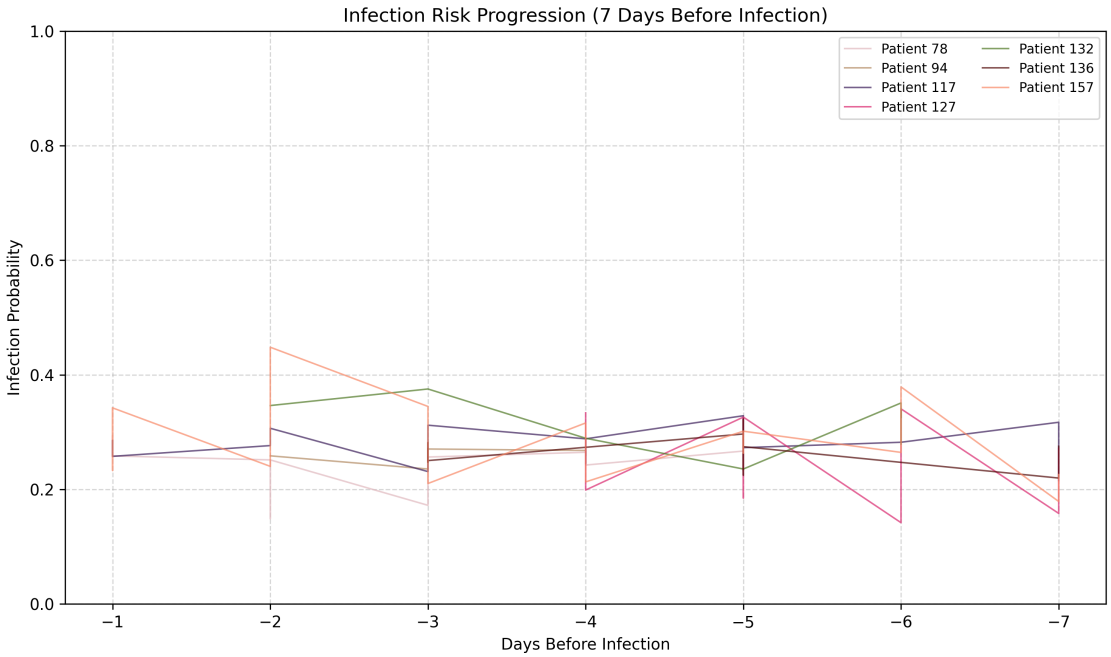


Figure 5.5: Infection risk probability during the 7 days preceding the first high-risk prediction

6

Conclusion

This chapter summarizes the main findings of the dissertation, highlights its contributions to AI-based wound monitoring, and outlines key limitations and future research directions.

6.1 Main Conclusions

An early and accurate detection of alterations in post-surgical wounds is essential to prevent the progression to SSIs, which remain a significant cause of post-operative complications, especially in cardio-thoracic surgery. This dissertation addressed this need for automated wound assessment by developing and evaluating a DL-based system capable of detecting cardio-thoracic surgical wounds and estimating the risk of infection through image analysis. The work was conducted within the scope of the CardioFollow.AI project, a telemonitoring plataform that enables daily data collection following hospital discharge.

A significant contribution of this work lies in the construction of a high-quality, structured dataset that includes manually annotated wound images, infection risk labels, and patient metadata in CSV format. This resource not only enabled the development and evaluation of the proposed system but also retains long-term value for future research, clinical validation, and the development of new algorithms.

The primary contribution of this study was the creation of a fully automated pipeline for wound region detection and infection risk classification. YOLOv8 was employed for wound detection, achieving Mean IoU of 94.7%, Dice Coefficient of 97.3%, and a Precision of 98.0%, despite the variations in acquisition conditions. A secondary classification step then predicted infection risk based on either a binary (No Risk vs. At Risk) or multiclass (wound type + risk level) framework. Experimental results demonstrated that the binary classification approach with data augmentation yielded the most robust performance, with an F1-score of 98.8% and a kappa score of 0.775 , while the multi-class model offered additional clinical detail with slightly lower F1-score, 95.7% and a kappa score of 0.902. As a result, the binary classification strategy is preferable for rapid screening scenarios, while the multiclass approach holds greater promise for fully automated and informative wound assessment systems in clinical practice.

To enhance the understanding of wound progression, a temporal analysis framework was also implemented to visualize infection risk over time for each patient. The combination of model-driven wound analysis and time-based visualization provides a valuable tool for post-discharge monitoring by enabling clinicians to distinguish between stable and deteriorating cases at a glance.

In conclusion, this work successfully delivered an AI-based system for surgical site monitoring, integrating object detection, risk classification, and temporal analysis. It represents a fundamental step toward reducing clinical workload, enhancing early detection of infections, and contributing to improved postoperative outcomes for the patients. With future improvements in dataset variability this system can be further refined and integrated into clinical workflows within the CardioFollow.AI platform.

6.2 Limitations and Future Work

As demonstrated, the proposed system shows potential in wound detection and infection risk classification. However, to reach clinical applicability, improvements are necessary at both the data and modeling levels.

A primary limitation of the current system lies in the imbalanced dataset, with a relatively small number of infected wound samples, which may have biased the performance and generalization of the classification model. Future work should focus on expanding the dataset in terms of representation of the infected class to improve the clinical relevance of the classifier. Additionally, improvements to the image acquisition process could involve integrating short video recordings instead of static images. Video data would provide richer spatial-temporal information, enabling more robust and dynamic analysis without compromising user autonomy.

Although YOLOv8 achieved strong performance, it would be relevant to experiment with different training/validation/test splits to assess their impact on model robustness. If performance declines, strategies such as downsampling the dominant class or benchmarking newer model architectures should be further explored. Moreover, due to computational constraints, hyperparameter tuning was limited, and a more exhaustive search could further improve model outcomes.

Future work could also enrich temporal analysis by incorporating features clinically relevant to wound progression, such as redness intensity, wound area, and texture irregularity, extracted using classical image processing or DL models. Instead of relying solely on static risk scores, tracking how predictions evolve over time may help distinguish between normal healing and early signs of infection. Additionally, integrating structured clinical metadata, such as comorbidities or laboratory results, could enhance prediction accuracy and support more personalized risk assessments.

Collectively, these proposed improvements in data acquisition, model architecture, temporal analysis, and multi-modal integration represent valuable next steps toward a more robust, interpretable, and clinically viable automated wound monitoring system.

Bibliography

[1] J. M. Lourenço. *The NOVAthesis LaTeX Template User's Manual*. NOVA University Lisbon. 2021. URL: https://github.com/joaomlourenco/novathesis/raw/main/template.pdf.

[2] World Health Organization. *The top 10 causes of death*. URL: https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death. (accessed: 10.09.2024).

[3] National Heart, Lung, and Blood Institute. *Heart Surgery*. URL: https://www.nhlbi.nih.gov/health/heart-surgery. (accessed: 20.09.2024).

[4] Value for Health CoLAB. *CardioFollow AI*. URL: https://cardiofollowai.vohcolab.org. (accessed: 27.01.2025).

[5] Pereira, Catarina and Guede-Fernández, Federico and Vigário, Ricardo and Coelho, Pedro and Fragata, José and Londral, Ana. "Image Analysis System for Early Detection of Cardiothoracic Surgery Wound Alterations Based on Artificial Intelligence Models". In: *Applied Sciences (Switzerland)* 13.4 (2023). ISSN: 20763417. DOI: 10.3390/app13042120.

[6] D. E. Wood. "Cardiothoracic surgery: A specialty divided or as one". In: *The Journal of Thoracic and Cardiovascular Surgery* 137.1 (2009), pp. 1–9. ISSN: 0022-5223. DOI: 10.1016/j.jtcvs.2008.10.030. URL: https://doi.org/10.1016/j.jtcvs.2008.10.030.

[7] Columbia University Irving Medical Center, Department of Surgery, New York, NY. *Heart Surgery*. URL: https://columbiasurgery.org/heart/approaches-heart-surgery#:~:text=Inopen-heartsurgery(or,theskinissuturedclosed). (accessed: 20.05.2024).

[8] Melly, Ludovic and Torregrossa, Gianluca and Lee, Timothy and Jansens, Jean Luc and Puskas, John D. "Fifty years of coronary artery bypass grafting". In: *Journal of Thoracic Disease* 10.3 (2018), pp. 1960–1967. ISSN: 20776624. DOI: 10.21037/jtd.2018.02.43.

[9] B. J. Bachar and B. Manna. "Coronary Artery Bypass Graft". In: StatPearls [Internet] (2025). [Updated 2023 Aug 8]. URL: https://www.ncbi.nlm.nih.gov/books/NBK507836.

[10] El-Ansary, Doa and LaPier, Tanya Kinney and Adams, Jenny and Gach, Richard and Triano, Susan and Katijjahbe, Md Ali and Hirschhorn, Andrew D and Mungovan, Sean F and Lotshaw, Ana and Cahalin, Lawrence P. "An Evidence-Based Perspective on Movement and Activity following Median Sternotomy". In: *Physical Therapy* 99.12 (2019), pp. 1587–1601. ISSN: 15386724. DOI: 10.1093/ptj/pzz126.

[11] European association for cardio-thoracic surgery. *Median sternotomy*. URL: https://mmcts.org/tutorial/80. (accessed: 13.04.2024).

[12] Matache, Radu and Dumitrescu, Mihai and Bobocea, Andrei and Cordoș, Ioan. "Median sternotomy - gold standard incision for cardiac surgeons". In: *Journal of Clinical and Investigative Surgery* 1 (2016-05), pp. 33–40. DOI: 10.25083/2559.5555.11.3340.

[13] Thoracic key. *Sternotomy*. URL: https://thoracickey.com/surgical-techniques-3/. (accessed: 05.05.2024).

[14] A. Young and C.-E. McNaught. "The physiology of wound healing". In: *Surgery (Oxford)* 29.10 (2011), pp. 475–479. ISSN: 0263-9319. DOI: https://doi.org/10.1016/j.mpsur.2011.06.011. URL: https://www.sciencedirect.com/science/article/pii/S0263931911001323.

[15] Schultz, Gregory S. and Chin, Gloria A. and Moldawer, Lyle and Diegelmann, Robert F. and Fitridge, Robert and Thompson, Matthew. "Principles of wound healing". In: *Mechanisms of Vascular Disease: A Reference Book for Vascular Specialists*. The University of Adelaide Press, 2011, 423–450.

[16] Ismail Aly, Mohamed E. and Dannoun, Moayad and Jimenez, Carlos J. and Sheridan, Robert L. and Lee, Jong O. "Operative Wound Management". In: *Total Burn Care, Fifth Edition* 42.12 (2017), 114–130.e2. DOI: 10.1016/B978-0-323-47661-4.00012-5.

[17] N. library of Medicine. *Infection*. URL: https://www.ncbi.nlm.nih.gov/medgen/811352. (accessed: 20.04.2024).

[18] Centers for Disease Control and Prevention. *SSIs Definition*. URL: https://www.cdc.gov/surgical-site-infections/about/index.html. (accessed: 25.05.2024).

[19] J. Fragata. *Vou ser operado ao coração, e agora?* 1ª edição, EDITORA D'IDEIAS, 2022. ISBN: 9789895345779.

[20] J. Hopkins Medicine. *Surgical Site Infections*. URL: https://www.hopkinsmedicine.org/health/conditions-and-diseases/surgical-site-infections. (accessed: 20.04.2024).

[21] S. Direct. *Image Analysis (Medical Imaging)*. URL: https://www.sciencedirect.com/topics/medicine-and-dentistry/image-analysis-medical-imaging. (accessed: 24.04.2024).

[22] Kasban, Hany and El-bendary, Mohsen and Salama, Dina. "A Comparative Study of Medical Imaging Techniques". In: *International Journal of Information Science and Intelligent System* 4 (2015-04), pp. 37–58.

[23] V. Tyagi. *Understanding Digital Image Processing*. Taylor Francis, 2018-09. ISBN: 9781315123905. DOI: 10.1201/9781315123905.

[24] Mohafez, Hamidreza and Ahmad, Siti and Roohi, Sharifah and Hadizadeh, Maryam. "Wound Healing Assessment Using Digital Photography: A Review". In: *Journal of Biomedical Engineering and Medical Imaging* 3 (2016-10). DOI: 10.14738/jbemi.35.2203.

[25] Nilufa Yeasmin, Most and Al Amin, Md and Jamal Joti, Tasmim and Aung, Zeyar and Abdul Azim, Mohammad. "Advances of AI in Image-Based Computer-Aided Diagnosis: A Review". In: *Array* 23 (2024-09). DOI: 10.1016/j.array.2024.100357.

[26] S. Direct. *Medical Image Segmentation*. URL: https://www.sciencedirect.com/topics/engineering/medical-image-segmentation#definition. (accessed: 17.05.2024).

[27] Yu, Ying and Wang, Chunping and Fu, Qiang and Kou, Renke and Huang, Fuyu and Yang, Boxiong and Yang, Tingting and Gao, Mingliang. "Techniques and Challenges of Image Segmentation: A Review". In: *Electronics* 12 (2023-03), p. 1199. DOI: 10.3390/electronics12051199.

[28] Xu, Yan and Quan, Rixiang and Xu, Weiting and Huang, Yi and Chen, Xiaolong and Liu, Fengyuan. "Advances in Medical Image Segmentation: A Comprehensive Review of Traditional, Deep Learning and Hybrid Approaches". In: *Bioengineering* 11.10 (2024). ISSN: 23065354. DOI: 10.3390/bioengineering11101034.

[29] A. Seth and S. Sharma. "Semantic Segmentation: A Systematic Analysis From State-of-the-Art Techniques to Advance Deep Networks". In: *Journal of Information Technology Research* 15 (2022-01), pp. 1–28. DOI: 10.4018/JITR.299388.

[30] Elyan, Eyad and Vuttipittayamongkol, Pattaramon and Johnston, Pamela and Martin, Kyle and McPherson, Kyle and Moreno-García, Carlos and Jayne, Chrisina and Sarker, Md. Mostafa Kamal. "Computer vision and machine learning for medical image analysis: recent advances, challenges, and way forward". In: *Artificial Intelligence Surgery* 2 (2022-03). DOI: 10.20517/ais.2021.15.

[31] D. Ruiz, G. Salomon, and E. Todt. *Can Giraffes Become Birds? An Evaluation of Image-to-image Translation for Data Generation*. 2020-01. DOI: 10.48550/arXiv.2001.03637.

[32] A. Kaplan and M. Haenlein. "Siri, Siri, in my hand: Who's the fairest in the land? On the interpretations, illustrations, and implications of artificial intelligence". In: *Business Horizons* 62.1 (2019), pp. 15–25. ISSN: 0007-6813. DOI: https://doi.org/10.1016/j.bushor.2018.08.004. URL: https://www.sciencedirect.com/science/article/pii/S0007681318301393.

[33] Furizal, Furizal and Ma'arif, Alfian and Rifaldi, Dianda. "Application of Machine Learning in Healthcare and Medicine: A Review". In: *Journal of Robotics and Control (JRC)* 4 (2023-09), p. 2023. DOI: 10.18196/jrc.v4i5.19640.

[34] Z. Shen, Y. Zhang, L. Wei, H. Zhao, Q. Yao. "Automated Machine Learning: From Principles to Practices". In: *arXiv preprint arXiv:1810.13306* 37.4 (2018). URL: https://arxiv.org/abs/1810.13306.

[35] J. García Cabello. "Mathematical Neural Networks". In: *Axioms* 11.2 (2022). ISSN: 2075-1680. DOI: 10.3390/axioms11020080. URL: https://www.mdpi.com/2075-1680/11/2/80.

[36] A. Kane and A. Hussain. "Artificial Neural Networks - An overview". In: *Indian Drugs* 30.5 (1993), pp. 168–178. ISSN: 0019462X. DOI: 10.58496/mjcsc/2023/015.

[37] IBM. *Deep Learning*. URL: https://www.ibm.com/topics/deep-learning. (accessed: 05.05.2024).

[38] V. Phung and E. Rhee. "A deep learning approach for classification of cloud image patches on small datasets". In: *Journal of Information and Communication Convergence Engineering* 16 (2018-01), pp. 173–178. DOI: 10.6109/jicce.2018.16.3.173.

[39] Ghosh, Anirudha and Sufian, Abu and Sultana, Farhana and Chakrabarti, Amlan and De, Debashis. *Fundamental Concepts of Convolutional Neural Network*. Vol. 172. January. 2020, pp. 519–567. ISBN: 9783030326449. DOI: 10.1007/978-3-030-32644-9_36.

[40] B. Mehlig. "Convolutional Networks". In: *Machine Learning with Neural Networks* (2021), pp. 141–156. DOI: 10.1017/9781108860604.008.

[41] Bimpas, Athanasios and Violos, John and Leivadeas, Aris and Varlamis, Iraklis. "Leveraging pervasive computing for ambient intelligence: A survey on recent advancements, applications and open challenges". In: *Computer Networks* 239 (2024), p. 110156. ISSN: 1389-1286. DOI: https://doi.org/10.1016/j.comnet.2023.110156. URL: https://www.sciencedirect.com/science/article/pii/S1389128623006011.

[42] Redmon, Joseph and Divvala, Santosh and Girshick, Ross and Farhadi, Ali. "You Only Look Once: Unified, Real-Time Object Detection". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 779–788. DOI: 10.1109/CVPR.2016.91.

[43] J. Terven, D.-M. Córdova-Esparza, and J.-A. Romero-González. "A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS". In: *Machine Learning and Knowledge Extraction* 5.4 (2023), pp. 1680–1716. ISSN: 2504-4990. DOI: 10.3390/make5040083. URL: https://www.mdpi.com/2504-4990/5/4/83.

[44] O. Rainio, J. Teuho, and R. Klén. "Evaluation metrics and statistical tests for machine learning". In: *Scientific Reports* 14.1 (2024), pp. 1–14. ISSN: 20452322. DOI: 10.1038/s41598-024-56706-x. URL: https://doi.org/10.1038/s41598-024-56706-x.

[45] Y. Sanjaya, A. Gunawan, and E. Irwansyah. "Semantic Segmentation for Aerial Images: A Literature Review". In: *Engineering, MAthematics and Computer Science (EMACS) Journal* 2 (2020), pp. 133–139. DOI: 10.21512/emacsjournal.v2i3.6737.

[46] Hicks, Steven A. and Strümke, Inga and Thambawita, Vajira and Hammou, Malek and Riegler, Michael A. and Halvorsen, Pål and Parasa, Sravanthi. "On evaluation metrics for medical applications of artificial intelligence". In: *Scientific Reports* 12.1 (2022), pp. 1–9. ISSN: 20452322. DOI: 10.1038/s41598-022-09954-8. URL: https://doi.org/10.1038/s41598-022-09954-8.

[47] Ragab, Mohammed Gamal and Abdulkadir, Said Jadid and Muneer, Amgad and Alqushaibi, Alawi and Sumiea, Ebrahim Hamid and Qureshi, Rizwan and Al-Selwi, Safwan Mahmood and Alhussian, Hitham. "A Comprehensive Systematic Review of YOLO for Medical Object Detection (2018 to 2023)". In: *IEEE Access* 12.December (2024), pp. 57815–57836. ISSN: 21693536. DOI: 10.1109/ACCESS.2024.3386826.

[48] Cascella, Marco and Shariff, Mohammed Naveed and Bianco, Giuliano Lo and Monaco, Federica and Gargano, Francesca and Simonini, Alessandro and Ponsiglione, Alfonso Maria and Piazza, Ornella. "Employing the Artificial Intelligence Object Detection Tool YOLOv8 for Real-Time Pain Detection: A Feasibility Study". In: *Journal of Pain Research* 17.November (2024), pp. 3681–3696. ISSN: 11787090. DOI: 10.2147/JPR.S491574.

[49] Anisuzzaman, D. M. and Patel, Yash and Niezgoda, Jeffrey A. and Gopalakrishnan, Sandeep and Yu, Zeyun. "A Mobile App for Wound Localization Using Deep Learning". In: *IEEE Access* 10 (2022), pp. 61398–61409. DOI: 10.1109/ACCESS.2022.3179137.

[50] Aldughayfiq, Bader and Ashfaq, Farzeen and Jhanjhi, N. Z. and Humayun, Mamoona. "YOLO-Based Deep Learning Model for Pressure Ulcer Detection and Classification". In: *Healthcare (Switzerland)* 11.9 (2023), pp. 1–19. ISSN: 22279032. DOI: 10.3390/healthcare11091222.

[51] Sendilraj, Varun and Pilcher, William and Choi, Dahim and Bhasin, Aarav and Bhadada, Avika and Bhadadaa, Sanjay Kumar and Bhasin, Manoj. "DFUCare: deep learning platform for diabetic foot ulcer detection, analysis, and monitoring".

In: *Frontiers in Endocrinology* 15.September (2024), pp. 1–10. ISSN: 16642392. DOI: 10.3389/fendo.2024.1386613.

[52] Reifs Jiménez, David and Casanova-Lozano, Lorena and Grau-Carrión, Sergi and Reig-Bolaño, Ramon. *Artificial Intelligence Methods for Diagnostic and Decision-Making Assistance in Chronic Wounds: A Systematic Review*. Vol. 49. 1. 2025. ISBN: 0123456789. DOI: 10.1007/s10916-025-02153-8.

[53] M. Suvarna, G. Toney, and G. B. Swastik. "Classification of scalding burn using image processing methods". In: *2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*. 2017, pp. 1312–1315. DOI: 10.1109/ICICICT1.2017.8342759.

[54] Zahia, Sofia and Sierra-Sosa, Daniel and Garcia-Zapirain, Begonya and Elmaghraby, Adel. "Tissue classification and segmentation of pressure injuries using convolutional neural networks". In: *Computer Methods and Programs in Biomedicine* 159 (2018), pp. 51–58. ISSN: 0169-2607. DOI: https://doi.org/10.1016/j.cmpb.2018.02.018. URL: https://www.sciencedirect.com/science/article/pii/S0169260717314864.

[55] A. J. Ramachandram D, Ramirez-GarciaLuna JL, Fraser RDJ, Martínez-Jiménez MA, Arriaga-Caballero JE. "Fully Automated Wound Tissue Segmentation Using Deep Learning on Mobile Devices: Cohort Study". In: *JMIR Mhealth Uhealth* 10 (20022). DOI: 10.2196/36977.

[56] Tusar, Mehedi Hasan and Fayyazbakhsh, Fateme and Zendehdel, Niloofar and Mochalin, Eduard and Melnychuk, Igor and Gould, Lisa and Leu, Ming C. "AI-Powered Image-Based Assessment of Pressure Injuries Using You Only Look Once Version 8 (YOLOv8) Models". In: *Advances in Wound Care* 0.0 (), null. DOI: 10.1089/wound.2024.0245. URL: https://doi.org/10.1089/wound.2024.0245.

[57] Russell, Bryan C. and Torralba, Antonio and Murphy, Kevin P. and Freeman, William T. "LabelMe: A database and web-based tool for image annotation". In: *International Journal of Computer Vision* 77.1-3 (2008), pp. 157–173. ISSN: 09205691. DOI: 10.1007/s11263-007-0090-8.

[58] Ultralytics. *YOLOv8*. URL: https://docs.ultralytics.com/pt/models/yolov8/#supported-tasks-and-modes. (accessed: 01.03.2025).

[59] Chan Gao, Chan and Zhang, Qingzhu and Tan, Zheyu and Zhao, Genfeng and Gao, Sen and Kim, Eunyoung and Shen, Tao. "Applying optimized YOLOv8 for heritage conservation: enhanced object detection in Jiangnan traditional private gardens". In: *Heritage Science* 12.1 (2024), pp. 1–20. ISSN: 20507445. DOI: 10.1186/s40494-024-01144-1. URL: https://doi.org/10.1186/s40494-024-01144-1.

[60] R.-Y. Ju and W. Cai. "Fracture detection in pediatric wrist trauma X-ray images using YOLOv8 algorithm". In: *Scientific Reports* 13 (2023-11). DOI: 10.1038/s41598-023-47460-7.

[61] Akiba, Takuya and Sano, Shotaro and Yanase, Toshihiko and Ohta, Takeru and Koyama, Masanori. "Optuna: A Next-generation Hyperparameter Optimization Framework". In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. KDD '19. Anchorage, AK, USA: Association for Computing Machinery, 2019, 2623–2631. ISBN: 9781450362016. DOI: 10.1145/3292500.3330701. URL: https://doi.org/10.1145/3292500.3330701.

I

Hyperparameter Search Results

This annex presents the detailed results of the hyperparameter optimization trials performed using Optuna for both object detection and risk classification tasks. For each experiment, the number of epochs, batch size, learning rate, and weight decay are listed, along with the resulting precision, recall, and mAP at IoU threshold 0.5 (mAP50).

Table I.1: Results for the Object Detection Model with initial dataset

Trial	Epochs	Batch	Learning Rate	Weight Decay	Precision	Recall	mAP50
1	53	16	0.000357	9.11e-06	93.0%	88.2%	93.0%
2	46	16	0.0121	3.21e-06	91.1%	93.4%	94.0%
3	42	16	0.00380	0.000156	93.2%	92.4%	94.3%
4	35	8	0.000866	8.54e-05	93.0%	92.0%	94.0%
5	57	16	0.000972	1.01e-05	90.9%	93.0%	93.4%

Table I.2: Results for the Object Detection Model with expanded dataset

Trial	Epochs	Batch	Learning Rate	Weight Decay	Precision	Recall	mAP50
1	40	8	0.0100	0.000500	97.7%	96.3%	98.2%
2	44	16	0.00787	1.62e-06	95.8%	96.0%	97.8%
3	60	32	4.32e-05	2.45e-06	97.6%	96.5%	98.0%
4	33	32	0.0103	3.71e-06	96.7%	95.3%	97.0%
5	50	8	0.00453	4.98e-05	97.7%	96.1%	98.0%

Table I.3: Results for Wound Detection and Risk Classification – Binary Classification without Augmentation

Trial	Epochs	Batch	Learning Rate	Weight Decay	Precision	Recall	mAP50
1	60	16	0.00320	2.15e-05	83.5%	79.2%	81.9%
2	40	16	0.00772	4.62e-06	77.0%	77.8%	76.4%
3	60	32	4.30e-05	5.46e-06	82.2%	79.9%	80.1%
4	55	32	0.0103	9.11e-06	81.9%	77.4%	79.9%
5	50	8	0.0688	4.70e-05	87.9%	74.8%	80.4%

Table I.4: Results for Wound Detection and Risk Classification – Binary Classification with Augmentation

Trial	Epochs	Batch	Learning Rate	Weight Decay	Precision	Recall	mAP50
1	55	32	0.00321	1.42e-06	93.8%	92.9%	89.0%
2	50	8	0.0336	2.22e-06	92.7%	92.5%	94.1%
3	45	8	3.00e-05	3.59e-06	91.7%	92.0%	93.5%
4	57	8	8.58e-05	4.72e-06	93.0%	92.7%	94.1%
5	52	16	0.000790	9.93e-05	93.5%	92.4%	95.2%

Table I.5: Results for Wound Detection and Risk Classification – Multiclass Classification without Augmentation

Trial	Epochs	Batch	Learning Rate	Weight Decay	Precision	Recall	mAP50
1	45	16	7.65e-05	0.000394	79.3%	74.3%	76.0%
2	55	16	0.000255	0.000677	88.9%	75.9%	81.1%
3	40	8	0.0914	1.83e-05	78.7%	69.2%	72.0%
4	33	8	0.0273	5.77e-05	83.1%	66.0%	70.3%
5	58	32	1.26e-05	0.000312	83.7%	75.0%	77.8%

Table I.6: Results for Wound Detection and Risk Classification – Multiclass Classification with Augmentation

Trial	Epochs	Batch	Learning Rate	Weight Decay	Precision	Recall	mAP50
1	55	32	1.33e-05	3.79e-05	91.2%	90.0%	92.9%
2	36	32	2.89e-05	2.18e-05	91.2%	89.8%	84.4%
3	60	8	5.19e-05	0.0000475	93.2%	90.7%	92.9%
4	40	32	0.000783	1.44e-05	92.3%	89.9%	92.5%
5	42	16	0.000102	3.42e-06	92.4%	89.4%	92.2%

Table I.7: Results for Wound Detection and Risk Classification – Binary Classification without Augmentation for Initial Dataset

Trial	Epochs	Batch	Learning Rate	Weight Decay	Precision	Recall	mAP50
1	55	8	0.00505	1.26e-05	84.3%	80.0%	84.2%
2	36	8	0.0264	2.44e-05	80.1%	75.0%	79.2%
3	46	16	0.0213	1.81e-06	84.1%	81.1%	85.3%
4	34	8	0.00268	3.23e-05	80.1%	74.3%	78.0%
5	39	16	0.0421	6.88e-06	83.4%	78.6%	83.0%

Table I.8: Results for Wound Detection and Risk Classification – Binary Classification without Augmentation with Cross Validation by Patients

Trial	Epochs	Batch	Learning Rate	Weight Decay	Precision	Recall	mAP50
1	47	32	0.000812	0.000229	66.3%	60.1%	61.0%
2	56	16	0.000508	9.88e-06	65.0%	62.4%	61.9%
3	49	32	7.55e-05	0.000219	62.1%	62.5%	61.8%
4	42	16	0.0231	2.24e-05	64.4%	64.0%	61.7%
5	45	16	0.00603	1.45e-05	63.0%	64.5%	62.6%