Vietnam National University, Hanoi
International School

==========

M.A. Thesis

SURGICAL TOOL INSTANCE SEGMENTATION BASED ON DEEP LEARNING FOR MINIMALLY INVASIVE SURGERY

TRAN LONG QUANG ANH

Field: **Master of Informatics and Computer Engineering**
Code: **8480111.01QTD**

Hanoi - 2025

Vietnam National University, Hanoi
International School

==========

M.A. Thesis

SURGICAL TOOL INSTANCE SEGMENTATION BASED ON DEEP LEARNING FOR MINIMALLY INVASIVE SURGERY

TRAN LONG QUANG ANH

Field: **Master of Informatics and Computer Engineering**
Code: **8480111.01QTD**
Supervisor: **Dr. Kim Dinh Thai**

Hanoi - 2025

CERTIFICATE OF ORIGINALITY

I, the undersigned, hereby certify my authority of the study project report entitled *"Surgical Tool Instance Segmentation based on Deep learning for Minimally Invasive Surgery"* submitted in partial fulfillment of the requirements for the degree of Master Informatics and Computer Engineering. Except where the reference is indicated, no other person's work has been used without due acknowledgement in the text of the thesis.

Hanoi, 22 June, 2025

Tran Long Quang Anh

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deepest gratitude to my supervisor, Dr. Kim Dinh Thai, for his invaluable guidance, support, and patience throughout the entire process of this thesis. His insightful advice and encouragement have been instrumental in shaping my research and enhancing my understanding of the subject matter.

I would also like to extend my heartfelt thanks to my professors and colleagues at International School, Vietnam National University, whose knowledge and discussions have greatly contributed to my academic growth. Their feedback and suggestions have helped refine my work and broaden my perspectives. A special thank you goes to my family and friends, who have always been a source of motivation and unwavering support. Their constant encouragement and belief in me have given me the strength to overcome challenges and complete this journey.

Finally, I would like to acknowledge all individuals and institutions that have provided assistance, resources, and inspiration during my research. This thesis would not have been possible without their contributions.

Hanoi, 22 June, 2025

Tran Long Quang Anh

ABSTRACT

Minimally Invasive Surgery (MIS) offers significant benefits over open surgery, including reduced postoperative pain, faster recovery, less scarring, and quicker healing. However, it poses challenges for surgeons due to indirect vision via endoscopic monitors, necessitating enhanced visual perception and precise instrument control.

This study addresses these challenges by optimizing YOLOv8 and YOLOv11 models, along with variants incorporating GhostConvolutions, Depthwise Convolution (DWConv), Mish, and GELU activation functions, for robust surgical tool instance segmentation. Leveraging the M2CAI16-Tool dataset, we employ a structured experimental approach to balance accuracy and computational efficiency.

Key findings reveal YOLOv11-DWConv as an efficient variant, achieving a 26% parameter reduction (7.4M) while retaining competitive detection mAP@0.5 (0.906), suitable for resource-constrained settings. Conversely, YOLOv11-GELU excels with superior detection accuracy (mAP@0.5: 0.910), highlighting GELU's enhanced localization capabilities. Real-time inference speeds (81 FPS for video, 75 FPS for live feeds) confirm practical applicability for intraoperative guidance.

Instance segmentation results facilitate objective skill assessment through instrument usage patterns, revealing procedural efficiency variations. This underscores the technology's potential for surgical evaluation.

Despite these advances, limitations persist, including trade-offs between accuracy and efficiency, robustness to endoscopic imaging challenges, and dataset constraints. Future directions involve exploring advanced compression techniques, adaptive preprocessing, expanded multi-institutional datasets, and integrating Transformer architectures and Self-Supervised Learning.

This research advances AI-driven surgical instrument detection and segmentation, offering optimized models that enhance safety, efficiency, and objective assessment in minimally invasive procedures, paving the way for improved surgical workflows.

LIST OF ABBREVIATIONS

Abbreviation	Meaning
MIS	Minimally Invasive Surgery
YOLO	You only look once
DWConv	Depthwise Convolutions
GELU	Gaussian Error Linear Units
mAP	mean Average Precision
FPS	Frames Per Second
AI	Artificial Intelligence
CAS	Computer-Assisted Surgery
CNN	Convolutional Neural Networks
NIH	National Institutes of Health
LIDC-IDRI	Lung Image Database Consortium
MSD Medical	Segmentation Decathlon
BUSI	Breast Ultrasound Images Dataset
ViT	Vision Transformers
SSL	Self-Supervised Learning

List of Figures

1.1 Minimally Invasive Surgery 2

2.1 Laparoscopic surgical instrument segmentation 8

3.1 Annotated frames from the M2CAI16-Tool dataset across training, validation, and test subsets 14
3.2 Object detection output using YOLOv8 16
3.3 Network architecture of YOLOv8 16
3.4 Network architecture of YOLOv11 17
3.5 C3k2 module in YOLOv11 18
3.6 SPPF module in YOLOv11 18
3.7 C2PSA module in YOLOv11 19
3.8 Schematic of Ghost Convolution 20
3.9 Profile of the Mish activation function 22
3.10 First and second derivatives of the Mish function 23
3.11 Profile of the GELU activation function 24
3.12 First and second derivatives of the GELU function 24
3.13 YOLOv8 with GhostConv backbone 25
3.14 YOLOv11 with GhostConv backbone 26
3.15 Structure of the C3Ghost module 26
3.16 YOLOv8 with C3Ghost backbone 27
3.17 YOLOv11 with C3Ghost backbone 27
3.18 YOLOv8 with DWConv backbone 28
3.19 YOLOv11 with DWConv backbone 29
3.20 Conv module with Mish and GELU activation functions 29
3.21 YOLOv8 and YOLOv11 training process 31
3.22 YOLOv8-Ghost and YOLOv11-Ghost training process 31
3.23 YOLOv8-C3Ghost and YOLOv11-C3Ghost training process 31
3.24 YOLOv8-DWConv and YOLOv11-DWConv training process 32
3.25 YOLOv8-Mish and YOLOv11-Mish training process 32
3.26 YOLOv8-GELU and YOLOv11-GELU training process 32

4.1 Successful detection and segmentation of surgical instruments in endoscopic videos 37
4.2 Misclassification examples in surgical instrument detection 37
4.3 Surgical Tool Usage Timelines for Videos 1–4 in the M2CAI16-Tool dataset (green: ground truth, yellow: algorithm predictions) 39
4.4 Total instrument usage times 40

List of Tables

4.1 Detection performance metrics of YOLOv8 and YOLOv11 variants . . 35
4.2 Instance segmentation performance metrics of YOLOv8 and YOLOv11 variants 36

Contents

CERTIFICATE OF ORIGINALITY i

ACKNOWLEDGEMENTS ii

LIST OF ABBREVIATIONS iv

List of Figures v

List of Tables vii

Contents viii

1 Introduction 1

1.1 Background and Motivation 1
1.2 Problem Statement 3
1.3 Objectives and Scope 3
1.4 Contributions 4
1.5 Thesis Structure 5

2 Literature Review 7

2.1 Minimally Invasive Surgery 7
2.2 Surgical Tool Detection and Segmentation 7
2.3 Medical Image Analysis 8
2.3.1 Deep Learning in Healthcare 8
2.3.2 Common Medical Datasets 10
2.4 Instance Segmentation 10
2.4.1 Traditional Methods 11
2.4.2 Deep Learning-Based Methods 11
2.5 Limitations of Existing Methods 12

3 Methodology 13

3.1 Data Acquisition and Preprocessing 13
3.2 YOLO Model 14
3.2.1 YOLOv8 15

3.2.2 YOLOv11 17
3.3 Network Components and Activation Functions 19
3.3.1 Ghost Module 19
3.3.2 Depthwise Convolution 20
3.3.3 Mish Function 22
3.3.4 GELU Function 23
3.4 Proposed Model Architectures 25
3.4.1 YOLO-Ghost Model 25
3.4.2 YOLO-Depthwise Convolution 28
3.4.3 YOLO-Mish and YOLO-GELU 29
3.5 Model Training 30
3.6 Evaluation Metrics 33

4 Experimental Results **34**
4.1 Inference Speed Assessment 34
4.2 Quantitative Results 35
4.3 Qualitative Results 36
4.4 Evaluation of Surgical Performance 37
4.5 Discussion 40

5 Conclusion **43**
5.1 Recap of the Main Contributions 43
5.2 Limitations and Future Directions 44

Publications **45**

References **46**

Chapter 1

Introduction

1.1 Background and Motivation

Background

In recent decades, **Minimally Invasive Surgery (MIS)** has emerged as one of the most significant advancements in modern surgical practices, marking a major breakthrough compared to traditional surgical methods. The work of Philipp Bozzini in 1806, which led to the development of a device called the *Lichtleiter* for observing internal cavities of the human body, is considered the foundation of modern endoscopy and an early precursor to MIS [1]. With the aid of endoscopic instruments, MIS enables surgeons to perform procedures through small incisions rather than large surgical openings as in conventional open surgery. This approach offers several notable advantages, such as reduced postoperative pain, lower risk of infection, shorter hospital stays, and accelerated patient recovery. In 1985, Erich Mühe performed the first laparoscopic cholecystectomy, paving the way for the subsequent development and widespread adoption of laparoscopic surgery [2].

However, MIS also presents considerable challenges for surgeons. Performing procedures through small ports restricts the maneuverability of surgical instruments, requiring a high level of dexterity and control. Additionally, since MIS relies on images transmitted from an endoscopic camera, the surgeon's field of view is limited, making it more difficult to accurately identify surgical instruments and surrounding tissues. These limitations can affect surgical precision, particularly in procedures that demand a high degree of accuracy, such as neurosurgery, cardiovascular surgery, and abdominal surgery.

Motivation

One of the most *critical aspects* of supporting surgeons in laparoscopic procedures is the ability to **accurately recognize and segment surgical instruments in real**

Figure 1.1: Minimally Invasive Surgery

time. The precise identification of instrument location, shape, and movement not only facilitates navigation but also plays a crucial role in *Computer-Assisted Surgery* [3] and robotic-assisted surgery. The recognition and segmentation of surgical instruments have significant potential applications that enhance both **surgical precision** and **patient safety**. These systems can highlight instruments on the screen, allowing for easier tracking and reducing the risk of confusion, while also playing a crucial role in preventing surgical errors by providing alerts in cases of misplaced or retained instruments, a serious risk that can lead to severe complications. Furthermore, in robotic-assisted surgery, accurate instrument recognition enables surgical robots to identify tools and surrounding tissues with greater precision, improving the accuracy of surgical maneuvers. Beyond real-time applications, these technologies also enhance medical training by offering more realistic and accurate simulated environments for surgical residents to learn and practice. Given these benefits, the development of advanced **artificial intelligence (AI)** models capable of reliably recognizing and segmenting surgical instruments has become an *urgent necessity*, paving the way for improved accuracy and efficiency in laparoscopic and minimally invasive procedures.

With the rapid advancements in AI and deep learning, the field of medical image analysis has achieved significant breakthroughs. **Deep learning models**, particularly *Convolutional Neural Networks (CNNs)*, have demonstrated outstanding performance in medical image processing, ranging from pathological tissue segmentation to lesion

classification and anatomical structure recognition. In the context of laparoscopic surgery, deep learning models can be employed to segment surgical instruments in images and videos captured from endoscopic cameras. Several state-of-the-art models, such as *U-Net* [4], *DeepLabV3+* [5], *Mask R-CNN* [6], and *YOLO* [7], have been explored and applied to this task. However, due to the unique characteristics of endoscopic images, the segmentation of surgical instruments remains a *challenging problem* that requires further research and improvement.

1.2 Problem Statement

In **Minimally Invasive Surgery (MIS)**, the core challenge addressed in this thesis is to accurately detect and segment surgical instruments in real-time endoscopic images. Given input as endoscopic video frames, the desired output is the precise position, type, and segmentation mask of instruments (e.g., Bipolar, Scissors) amidst complex conditions—variable lighting, occlusions, and tissue noise [8]. This is critical for *Computer-Assisted Surgery (CAS)* and robotic-assisted surgery, where precision and safety hinge on reliable, real-time tracking with latency below milliseconds to ensure seamless integration into surgical workflows [9]. Current **deep learning** methods struggle with accuracy, speed, and adaptability, necessitating a robust solution.

This problem's resolution enhances surgical precision by delivering real-time instrument data—position and type—for navigation and robotic automation, reducing errors and improving patient outcomes. It enables intelligent *CAS* systems to optimize workflows using **artificial intelligence**, advancing surgical technology.

Moreover, this thesis leverages detection and segmentation outputs to evaluate surgeons' skills by analyzing instrument usage patterns, such as frequency of use, duration per tool, and movement efficiency (e.g., trajectory smoothness). These metrics provide objective insights into dexterity and precision, enabling personalized training, enhancing simulators, and standardizing surgical quality. Thus, this research tackles real-time instrument recognition while transforming skill assessment and surgical education.

1.3 Objectives and Scope

This study aims to develop an advanced method for recognizing and segmenting surgical instruments in **Minimally Invasive Surgery (MIS)** by enhancing the *YOLO (You Only Look Once)* model, renowned for its high speed and accuracy in object

detection. Applying YOLO to surgical environments is challenging due to variable lighting, occlusions, overlapping instruments, and tissue noise. To address these issues, the research focuses on three key objectives.

The first objective is to apply the YOLO architecture to accurately detect the position and type of surgical instruments in endoscopic images. This involves creating a well-annotated dataset, optimizing data preparation, and using preprocessing techniques to improve model performance under complex surgical conditions.

The second objective is to enhance the YOLO model for endoscopic surgery by modifying its architecture to boost recognition accuracy while maintaining computational efficiency. This includes reducing model parameters, applying data augmentation to handle real-world variations, and fine-tuning on a specialized endoscopic dataset to enhance generalization across diverse surgical scenarios.

The third objective is to evaluate the enhanced YOLO model using standard metrics, such as *mean Average Precision (mAP)*, precision, recall, and *Frames Per Second (FPS)*, to ensure its effectiveness and reliability in real-time surgical applications.

The study will develop and test the YOLO-based model on a dataset of endoscopic images featuring seven instrument types: Bipolar, Clipper, Hook, Irrigator, Scissors, Specimen Bag, and Grasper. The dataset will be preprocessed, including labeling, normalization, and splitting into training, validation, and test sets, to align with YOLO's requirements. Enhancements to the model, such as integrating *Ghost modules* [10] and *Depthwise Convolution (DWConv)* [11], will improve detection accuracy and reduce computational costs, making it suitable for resource-constrained surgical settings.

The scope of this research centers on optimizing YOLO for surgical instrument recognition in MIS to improve detection accuracy, speed, and generalization. Beyond real-time surgical assistance, this work supports robotic-assisted surgery, surgical automation, and medical training by providing objective metrics on instrument movement and positioning. These metrics enable the evaluation of surgeons' technical skills, such as dexterity and precision, facilitating personalized training, enhancing surgical simulators, and standardizing surgical quality, thus advancing surgical proficiency and patient outcomes.

1.4 Contributions

The recognition and segmentation of surgical instruments in endoscopic images is a critical challenge in **Minimally Invasive Surgery (MIS)**, impacting *Computer-Assisted Surgery (CAS)* and robotic-assisted surgery. This study advances this field

by enhancing the *YOLOv8* and *YOLOv11* models to improve the accuracy and efficiency of surgical instrument detection and segmentation in real-world conditions. The primary contributions are:

(1) **Enhancing YOLO Models.** This study fine-tunes *YOLOv8* and *YOLOv11* on a dataset of endoscopic images with seven instrument types—Bipolar, Clipper, Hook, Irrigator, Scissors, Specimen Bag, and Grasper—using preprocessing and data augmentation to improve detection accuracy under complex conditions. A comparative analysis of the models, based on accuracy, speed, and robustness, identifies the optimal model for surgical applications, enhancing procedural accuracy and patient safety.

(2) **Performance Optimization.** This study integrates *Depthwise Convolution (DWConv)* [11] and *Ghost Convolution* [10] into the YOLO architecture to reduce computational costs while maintaining accuracy. A comparative analysis, using metrics like *Frames Per Second (FPS)*, determines the best approach for surgical applications, balancing efficiency and complexity.

(3) **Improving Non-Linearity.** This study investigates *Mish* [12] and *GELU* [13] activation functions to enhance YOLO models' learning capabilities. A comparative analysis of convergence speed, gradient stability, and accuracy identifies the optimal function, improving model robustness in medical image analysis.

(4) **Evaluating Surgical Efficiency.** This study uses detection and segmentation results on the test dataset to evaluate surgical efficiency, analyzing metrics like instrument movement smoothness and positioning accuracy to assess surgeons' skills, supporting training programs, simulators, and surgical quality standardization, thus improving proficiency and patient outcomes.

1.5 Thesis Structure

This thesis, spanning six chapters, explores surgical instrument recognition using the YOLO model. **Chapter 1** introduces Minimally Invasive Surgery (MIS), highlighting the importance and challenges of instrument recognition, followed by the research objectives and key contributions. **Chapter 2** reviews existing studies on detection and segmentation, focusing on deep learning applications in medical image analysis and the limitations of current methods. **Chapter 3** outlines the methodology, covering data collection, preprocessing, network architecture design, and model

training with evaluation criteria. **Chapter 4** presents experimental results, analyzing performance metrics and visualization of detection and segmentation outcomes. **Chapter 5** discusses these findings, addressing research limitations and proposing future improvements. **Chapter 6** concludes by summarizing contributions, emphasizing clinical significance, and suggesting potential applications and developments.

Chapter 2

Literature Review

2.1 Minimally Invasive Surgery

Minimally Invasive Surgery (MIS) is a technique using small incisions, typically under 2 cm, with specialized instruments and miniature cameras to perform procedures while minimizing tissue damage. Introduced by Dr. John E. A. Wickham in 1987, MIS reduces postoperative pain, shortens recovery time, and improves patient outcomes compared to traditional open surgery [14]. Its origins date back to the 19th-century cystoscope, followed by key advancements like the Veress needle (1938) for pneumoperitoneum, the Hasson technique (1970) for open laparoscopy, and the "video-endoscopy" era sparked by solid-state cameras in 1982. A milestone came in 1981 with Kurt Semm's first laparoscopic appendectomy, solidifying MIS's role in modern surgery [15].

MIS includes techniques like laparoscopic and thoracoscopic surgery, relying on endoscopes for real-time visualization and precise instrument manipulation through tiny incisions. Widely applied in fields such as gastrointestinal surgery, urology, and gynecology, MIS offers reduced pain, faster recovery, lower infection risk, and minimal scarring, enhancing patient satisfaction and hospital efficiency. However, challenges include high training and equipment costs, limiting accessibility, and its unsuitability for some complex cases where open surgery remains preferable.

Advancements like robot-assisted surgery and **artificial intelligence**-driven systems are shaping the future of MIS, improving precision and expanding its applications [16]. These innovations promise safer, more efficient procedures, redefining surgical care and patient outcomes.

2.2 Surgical Tool Detection and Segmentation

Surgical tool detection and segmentation are pivotal in advancing modern surgery by identifying the position and shape of instruments, enhancing efficiency and safety.

These processes support **Computer-Assisted Surgery (CAS)** and robotic systems by providing real-time tool tracking for precise navigation and control, reducing risks during procedures [17]. Beyond intraoperative use, segmentation aids surgical skill assessment, procedural planning, and workflow analysis through detailed movement data, improving training and clinical outcomes. It also drives innovations like robotic surgery and augmented reality (AR), where accurate segmentation enables high-precision maneuvers and enhanced visualization, shaping the future of medical technology.

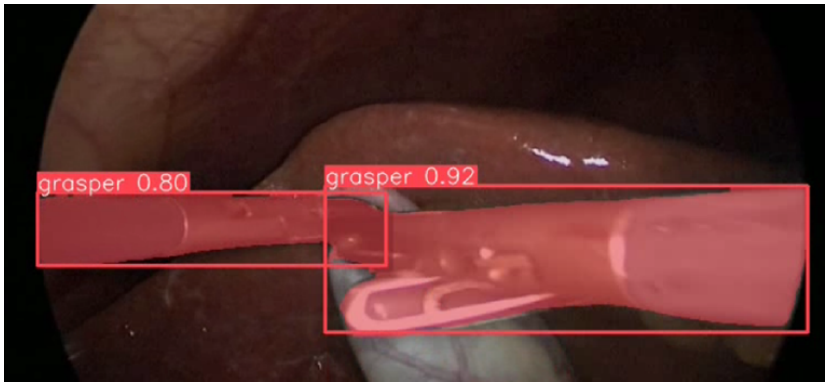


Figure 2.1: Laparoscopic surgical instrument segmentation

Before **deep learning**, segmentation relied on traditional methods like thresholding, edge detection, region-based approaches, and model-based techniques. Thresholding separated tools from backgrounds using intensity but faltered under variable lighting [18]. Edge detection identified boundaries yet struggled with noise, while region-based methods depended on feature selection, often failing with similar backgrounds. Model-based segmentation used predefined shapes but lacked adaptability to deformations or occlusions [19]. These limitations spurred the shift to deep learning for more robust solutions.

Challenges in surgical tool segmentation include variability in instrument shape and size, motion and deformation during surgery, changing lighting conditions, and occlusions from blood or tissue. Noise from smoke or fluids further degrades image quality, complicating accurate detection [20]. Addressing these issues requires integrating traditional techniques with advanced deep learning models to improve reliability and precision in real-time surgical applications.

2.3 Medical Image Analysis

2.3.1 Deep Learning in Healthcare

The evolution of **deep learning** has reshaped medical image analysis over recent years, moving beyond traditional methods that depended on manually crafted fea-

tures to data-driven approaches offering superior accuracy and adaptability. Studies have increasingly harnessed *Convolutional Neural Networks (CNNs)* to tackle essential tasks, starting with disease classification where architectures like ResNet and EfficientNet emerged as powerful tools for identifying abnormalities in X-rays and MRIs [21]. This progress extended to segmentation, with models such as U-Net and DeepLab refining the delineation of tissues and surgical instruments, a leap forward from earlier techniques [22]. Research then explored anomaly detection, employing autoencoders and GANs to uncover irregularities in scans, enhancing diagnostic precision [23]. These advancements converged in *Computer-Aided Diagnosis (CAD)* systems, which have evolved from basic support tools to sophisticated aids for clinical decision-making, reflecting deep learning's growing impact across healthcare imaging applications [24].

The literature reveals a broadening scope of deep learning applications, driven by its ability to learn directly from raw medical images. Initial efforts focused on classification, where CNNs outperformed traditional methods in detecting pathologies across diverse modalities like CT and ultrasound [21]. Subsequent studies advanced segmentation, with models like nnU-Net improving precision in outlining anatomical structures and pathological regions, critical for surgical planning [22]. Concurrently, anomaly detection gained traction, as GAN-based approaches proved effective in spotting subtle deviations in complex scans, addressing gaps left by earlier methods [23]. This trajectory culminated in enhanced CAD systems, now integral to clinical workflows, leveraging deep learning to handle increasingly intricate diagnostic tasks and improve patient outcomes [24].

Despite these strides, research highlights persistent challenges in applying deep learning to medical imaging. Early studies struggled with limited labeled datasets, a barrier due to the expertise and time required for annotation, prompting exploration of unsupervised and self-supervised learning to lessen reliance on manual labels [25]. Another issue emerged as domain shift, where models trained on specific datasets faltered on new data due to variations in imaging protocols or patient populations, leading to the adoption of transfer learning to boost adaptability [26]. Ongoing investigations continue to address these hurdles, refining deep learning techniques to ensure robust, efficient, and widely accessible tools for medical image analysis, poised to further transform diagnostic practices.

2.3.2 Common Medical Datasets

The advancement of **deep learning** in medical image analysis hinges on high-quality datasets, which provide diverse images and expert annotations essential for model development. Research has progressively curated datasets to address varied objectives, from disease diagnosis to surgical tool recognition, shaping the evolution of data-driven medical imaging. Early efforts focused on X-ray analysis, with datasets like ChestX-ray14 (112,120 images, 14 diseases) [27] and MIMIC-CXR (over 370,000 images) [28] enabling pneumonia and lung cancer detection studies. These paved the way for tuberculosis research using Montgomery and Shenzhen datasets.

Subsequent studies expanded to MRI and CT imaging, where BraTS [29] emerged for brain tumor segmentation, offering annotated MRI scans to refine tumor delineation algorithms. Similarly, LIDC-IDRI [30] provided CT scans with nodule annotations for lung cancer detection, while the Medical Segmentation Decathlon (MSD) [31] broadened the scope with multi-organ MRI and CT data, fostering generalizable segmentation approaches. In endoscopic surgery, datasets like EndoVis and Cholec80 [32] introduced real-world surgical images and videos, annotated for instrument detection and procedural analysis, supporting intelligent surgical systems.

The literature also highlights datasets in specialized domains. For ultrasound, BUSI enabled breast cancer detection with 780 annotated images, while histopathological datasets like Camelyon16/17 and PAIP 2019 advanced metastasis and liver cancer analysis through annotated pathology images. Despite their foundational role, these datasets face challenges, including limited size, device variability, and annotation demands, prompting research into multi-dataset integration to enhance model accuracy and adaptability in clinical applications.

2.4 Instance Segmentation

Instance segmentation, a critical task in computer vision, has evolved as a specialized form of image segmentation, dividing images into distinct objects rather than just regions, unlike semantic segmentation. Research highlights its growing importance in medical image analysis, particularly in **Computer-Assisted Surgery (CAS)** and robotic surgery, where distinguishing individual surgical instruments enhances procedural accuracy and safety. Initial studies focused on basic segmentation, but the need to identify each tool uniquely in complex surgical scenes spurred the development of instance segmentation, laying the groundwork for advanced surgical applications.

2.4.1 Traditional Methods

Early efforts in instance segmentation leaned on classical image processing techniques, adapting methods like thresholding, edge-based, and region-based segmentation for medical imaging. Watershed Segmentation emerged as a key approach, exploiting intensity differences to define object boundaries [33], yet its sensitivity to noise and overlap limited reliability in surgical contexts. Concurrently, Graph Cut and GrabCut techniques modeled images as graphs, separating objects via intensity-based cuts, though performance waned in scenes with unclear edges. Active Contour Models followed, using adaptable contours to capture flexible instrument shapes [34], but struggled with initialization and noise, particularly under occlusions. These traditional methods, while foundational, proved inadequate for the dynamic challenges of surgical environments—overlapping tools, variable lighting, and noise—prompting a shift to **deep learning** approaches for improved precision and robustness.

2.4.2 Deep Learning-Based Methods

The advent of **deep learning** has revolutionized image segmentation, with *Convolutional Neural Networks (CNNs)* surpassing traditional methods in accuracy and robustness, despite higher computational demands. Advances in hardware acceleration have mitigated these costs, enabling real-time applications in medical imaging. Research has progressed from early CNN-based models to sophisticated architectures tailored for segmentation, each addressing specific challenges in the field.

U-Net, introduced by Ronneberger et al. (2015) [4], marked a pivotal shift with its encoder-decoder design and skip connections, preserving spatial details for precise medical image segmentation. Its contracting path extracts hierarchical features via convolutions and pooling, while the expanding path restores resolution, making it ideal for tasks like tumor and instrument delineation. Variants like U-Net++ [35] enhanced this with dense skip connections for finer multi-scale fusion, and Attention U-Net [36] added focus on key regions, boosting accuracy in complex scenes. The 3D U-Net [37] extended this to volumetric data, improving segmentation in MRI and CT scans.

SegNet, proposed by Badrinarayanan et al. (2017) [38], emerged as a lightweight alternative, optimizing efficiency with stored pooling indices instead of skip connections. Its encoder captures features, and the decoder reconstructs spatial details using these indices, prioritizing speed for real-time medical applications, though at some cost to fine-detail accuracy. DeepLab, evolving through versions from v1 (2015) to v3+ (2018) by Chen et al. [39], introduced Atrous Convolution and ASPP to capture

multi-scale context, refining boundaries with an encoder-decoder structure, proving effective for endoscopic and high-resolution imaging [40].

Mask R-CNN, developed by He et al. (2017) [6], advanced instance segmentation by extending Faster R-CNN with a mask prediction branch. Leveraging ResNet-FPN for feature extraction and ROI Align for precise alignment, it excels in distinguishing individual surgical tools, enhancing CAS and robotic surgery applications. These models collectively illustrate a trajectory of increasing sophistication, addressing accuracy, efficiency, and adaptability in medical segmentation.

2.5 Limitations of Existing Methods

Research on computer vision for surgical tool detection and segmentation has progressed significantly, yet real-world applications, particularly in endoscopic surgery, reveal persistent limitations. Early studies established robust frameworks, but challenges emerged in complex surgical environments. Lighting variations, driven by instrument movement and camera angles, disrupt color- and contrast-based algorithms, reducing detection accuracy. Occlusion from tissues or overlapping tools further hampers edge- and region-based methods, while biological artifacts—blood, smoke, and soft tissues—obscure instruments, complicating object recognition.

Advanced **deep learning** models like Mask R-CNN [6] and DeepLabv3+ [40] have elevated segmentation precision, yet their computational demands hinder real-time performance critical for surgical safety, where even millisecond delays pose risks. Additionally, the similarity in instrument shapes and colors, exacerbated by varying angles and proximity, often leads to misclassification in models such as U-Net and Mask R-CNN. Recognizing these gaps, recent efforts, including this study, explore enhanced models to boost accuracy and efficiency, addressing the dual need for precision and speed in surgical applications.

Chapter 3

Methodology

3.1 Data Acquisition and Preprocessing

This study employs two endoscopic datasets for laparoscopic cholecystectomy: the M2CAI16-Tool dataset [41] and the Cholec80 dataset [32]. The M2CAI16-Tool dataset, sourced from the 2016 M2CAI Tool Presence Detection Challenge, comprises 15 high-resolution laparoscopic surgery videos recorded at the University Hospital of Strasbourg. Each video captures real operative conditions, annotated for the presence of seven surgical instruments: Bipolar, Clipper, Hook, Irrigator, Scissors, Specimen Bag, and Grasper. Complementing this, the Cholec80 dataset includes 80 cholecystectomy videos performed by 13 surgeons, acquired at 25 frames per second, with annotations detailing tool usage and surgical phases, enhancing its utility for procedural analysis.

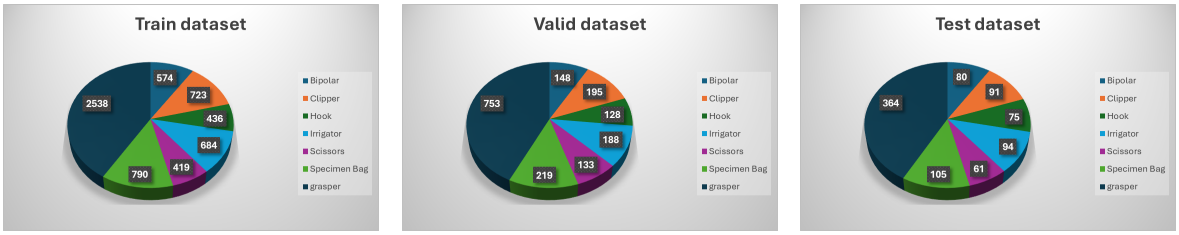
Given the absence of bounding box coordinates and segmentation masks in the M2CAI16-Tool dataset, manual annotation was performed using Roboflow. High-quality frames (5,249 in total) were extracted from videos 1 to 10, selected to encompass diverse surgical scenarios, including variable illumination and instrument occlusions. Annotations involved defining bounding boxes with normalized center coordinates $(x_{\text{center}}, y_{\text{center}})$ and dimensions $(\text{width}, \text{height})$ in the range [0, 1], alongside segmentation masks delineated by polygon vertices $(x_1, y_1, \ldots, x_n, y_n)$. The annotated data was exported in YOLO format, adhering to the structure below:

$$[\text{class\_id}, x_{\text{center}}, y_{\text{center}}, \text{width}, \text{height}, x_1, y_1, x_2, y_2, \ldots, x_n, y_n], \tag{3.1}$$

where class_id $\in \{0, 1, \ldots, 6\}$ denotes the instrument type (e.g., Bipolar, Clipper), $(x_{\text{center}}, y_{\text{center}})$ and $(\text{width}, \text{height})$ are normalized bounding box parameters, and (x_i, y_i) for $i = 1, \ldots, n$ represent the segmentation mask vertices, all scaled to [0, 1] relative to frame dimensions.

Data preprocessing ensured compatibility with the YOLO model. Frames were resized to 640×480 pixels for uniformity and normalized to a [0, 1] intensity scale to

facilitate training convergence. The annotated frames were partitioned into training (3,674 frames, 70%), validation (1,050 frames, 20%), and test (525 frames, 10%) sets, adhering to a 7:2:1 ratio. This division optimizes training coverage, validation tuning, and generalization assessment. Figure 3.1 illustrates representative annotated frames from each subset, highlighting instrument diversity and annotation quality.



(a) Training (3,674 images) (b) Validation (1,050 images) (c) Test (525 images)

Figure 3.1: Annotated frames from the M2CAI16-Tool dataset across training, validation, and test subsets

3.2 YOLO Model

The YOLO (You Only Look Once) model, a **Convolutional Neural Network (CNN)**-based framework, has redefined object detection by integrating region proposal and classification into a single-step process, achieving real-time performance with high accuracy [42]. Unlike traditional two-stage detectors like R-CNN, YOLO's unified architecture processes images holistically, offering a significant advancement over sequential methods. Since its inception by Redmon et al. (2016), YOLO has evolved through multiple iterations, enhancing its capabilities for object detection, instance segmentation, and pose estimation, driven by contributions from various research groups.

YOLOv1 [42] introduced the single-stage paradigm, leveraging a streamlined CNN to achieve unprecedented speed, though with trade-offs in precision compared to region-based methods. YOLOv2 [43] improved accuracy via Batch Normalization and anchor boxes, expanding detection to over 9,000 classes. YOLOv3 [44] adopted Darknet-53 as its backbone, incorporating multi-scale feature maps to enhance detection across object sizes, balancing speed and accuracy. Subsequent developments, such as YOLOv4 [45], refined the Darknet framework with advanced training strategies, while YOLOv5 [46] optimized scalability and deployment efficiency. YOLOv6 [47] and YOLOv7 [48] further improved computational efficiency, with applications extending to robotics and high-performance tasks.

Recent iterations have pushed the boundaries of YOLO's capabilities. YOLOv8

[49] introduced network optimizations and enhanced training protocols, improving segmentation and pose estimation. YOLOv9 [50] incorporated Programmable Gradient Information (PGI) to refine gradient updates, bolstering robustness, while YOLOv10 [51] adopted an NMS-free approach, achieving state-of-the-art performance with reduced latency. The latest, YOLOv11 [52], developed by Ultralytics, integrates these advancements, offering superior accuracy, speed, and versatility across detection, segmentation, and classification tasks. Its customizability and performance make it a leading model for real-time applications.

This study selects YOLOv8 and YOLOv11 for improvement and evaluation, leveraging their stability and precision. YOLOv8 provides a well-validated baseline, while YOLOv11, the most recent iteration at the time of this research, demonstrates marked improvements in accuracy and efficiency, as evidenced by recent literature [52]. These models are particularly suited for high-precision surgical instrument recognition in endoscopic image analysis, addressing real-time processing demands and robust segmentation in complex surgical environments. Details of the YOLOv8 and YOLOv11 models are presented in detail in the following subsections.

3.2.1 YOLOv8

YOLOv8, an advanced iteration of the YOLO framework, leverages a **Convolutional Neural Network (CNN)** architecture to achieve state-of-the-art performance in real-time object detection and segmentation. Central to its design is the adoption of CSPNet (Cross Stage Partial Network) as the backbone, paired with an FPN+PAN (Feature Pyramid Network + Path Aggregation Network) neck, optimizing feature extraction and multi-scale aggregation. CSPNet minimizes computational redundancy, enhancing efficiency, while FPN+PAN ensures robust detection across diverse object sizes and aspect ratios, critical for complex datasets such as surgical imagery.

A pivotal advancement in YOLOv8 is its shift to an anchor-free detection mechanism, departing from the anchor box reliance of predecessors like YOLOv3 and YOLOv5 [46]. This eliminates the need for extensive hyperparameter tuning, reducing computational overhead and improving adaptability to varied object morphologies. The anchor-free approach accelerates training convergence and enhances generalization, yielding superior accuracy on heterogeneous datasets. Additionally, YOLOv8 integrates Focal Loss, defined as:

$$FL(p_t) = -(1-p_t)^{\gamma}\log(p_t), \tag{3.2}$$

where p_t is the predicted probability, and γ adjusts focus on difficult samples to mitigate class imbalance in object detection, enhancing precision.

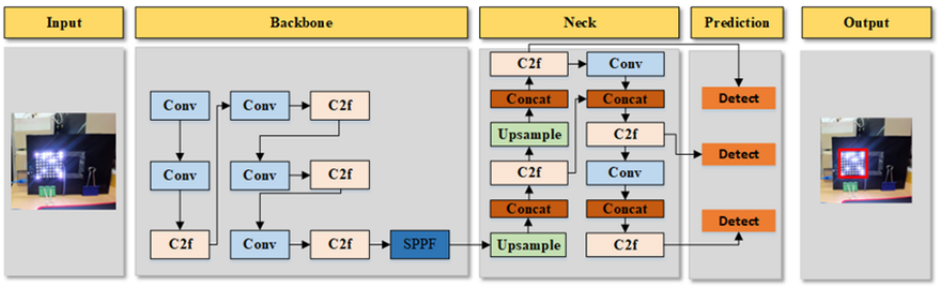


Figure 3.2: Object detection output using YOLOv8

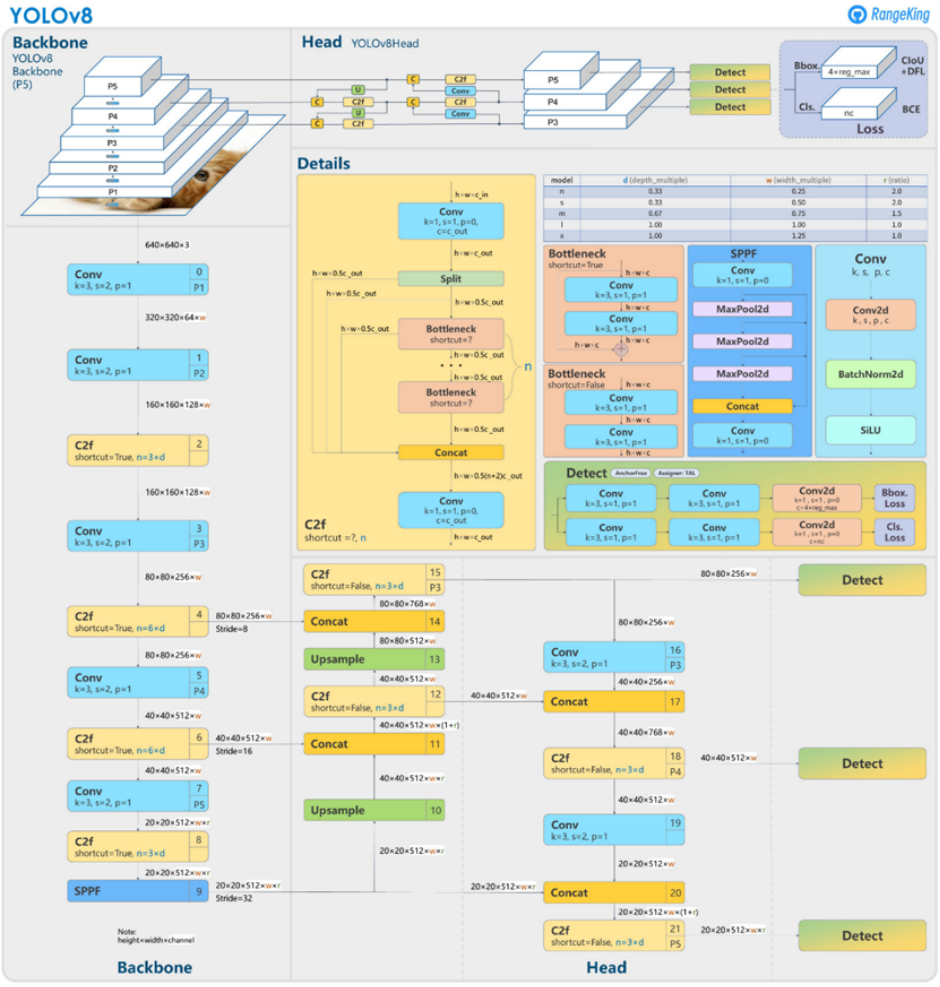


Figure 3.3: Network architecture of YOLOv8

Training efficiency is further augmented through PyTorch-based optimizations and Mixed Precision Training, leveraging GPU resources to minimize latency and memory usage without compromising accuracy. YOLOv8 also employs advanced data augmentation strategies, notably Mosaic and Mixup Augmentation. Mosaic Augmentation combines four images into a single frame with randomized cropping and scaling, enriching dataset variability and improving robustness to scale and occlusion variations. Mixup Augmentation blends two images via weighted averaging of pixels and labels, enhancing the model's capacity to discern objects in cluttered environ-

ments, a key advantage for medical imaging applications.

Figures 3.2 and 3.3 illustrate YOLOv8's detection output and network architecture, respectively. These enhancements—anchor-free design, optimized loss, and augmentation—elevate YOLOv8's performance, making it highly efficient and scalable for real-time vision tasks. As evaluated in this study, its developer-friendly interface, including a Python API and CLI, further facilitates deployment, positioning YOLOv8 as a leading solution for precise surgical instrument recognition in endoscopic analysis.

3.2.2 YOLOv11

YOLOv11, unveiled at the YOLO Vision 2024 Conference, represents the latest advancement in the YOLO (You Only Look Once) series, enhancing real-time object detection within a **Convolutional Neural Network (CNN)** framework. Building upon its predecessors, YOLOv11 introduces significant architectural and training innovations, achieving superior accuracy, efficiency, and scalability across multiple vision tasks, including object detection, instance segmentation, pose estimation, oriented object detection, and image classification.

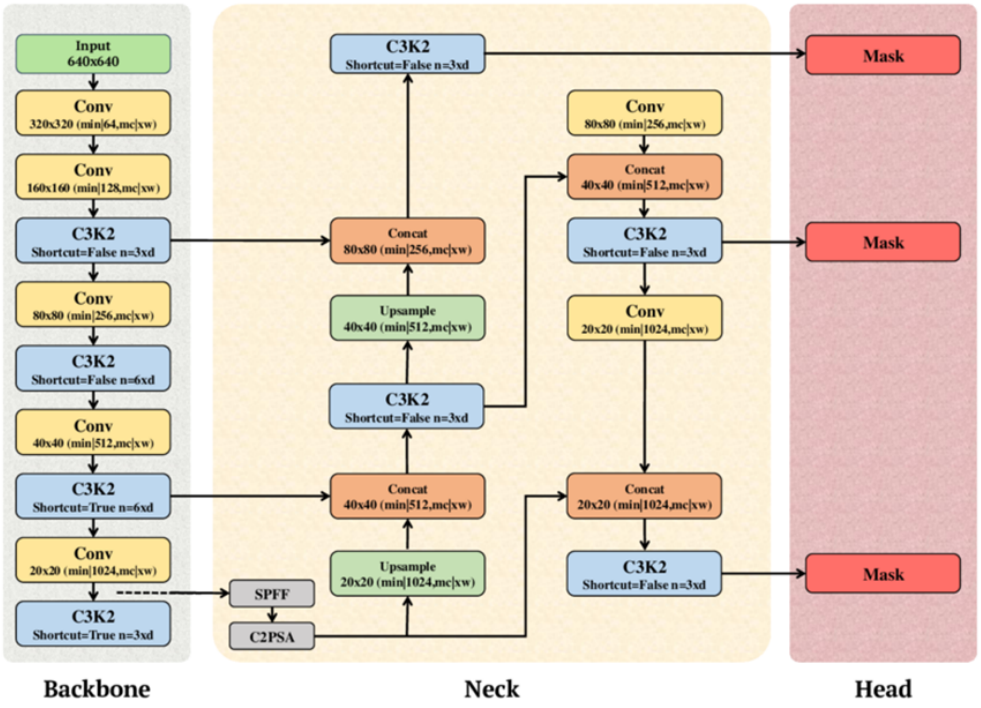


Figure 3.4: Network architecture of YOLOv11

The backbone of YOLOv11 replaces the C2f module with C3k2, a refined structure utilizing a kernel size of 2 to reduce parameter count while preserving robust feature extraction capabilities. This optimization, depicted in Figure 3.5, enhances computational efficiency, enabling faster inference without compromising performance, ideal

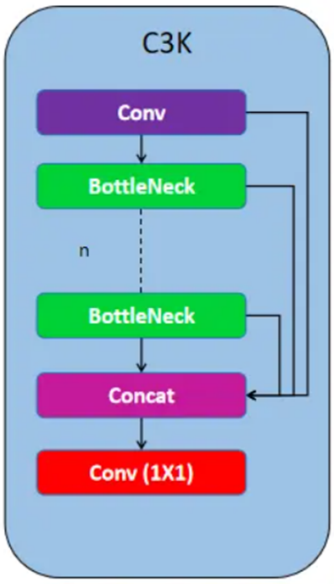


Figure 3.5: C3k2 module in YOLOv11

for real-time applications such as surgical tool recognition. The SPPF (Spatial Pyramid Pooling - Fast) module, illustrated in Figure 3.6, is further optimized to aggregate multi-scale features, improving processing speed and feature quality.

The neck integrates C3k2 and C2PSA (Convolutional Block with Parallel Spatial Attention), as shown in Figure 3.7, to streamline feature transmission and enhance multi-layer aggregation. C2PSA improves spatial attention, bolstering detection of small or occluded objects, while C3k2 ensures efficient multi-scale feature processing, critical for handling diverse object sizes and orientations in endoscopic imagery. The head employs C3k2 Blocks for high-level feature refinement and CBS Blocks (Convolution-BatchNorm-SiLU) for stability, with Batch Normalization standardizing feature distributions and SiLU activation introducing smooth non-linearity, enhancing convergence and generalization.

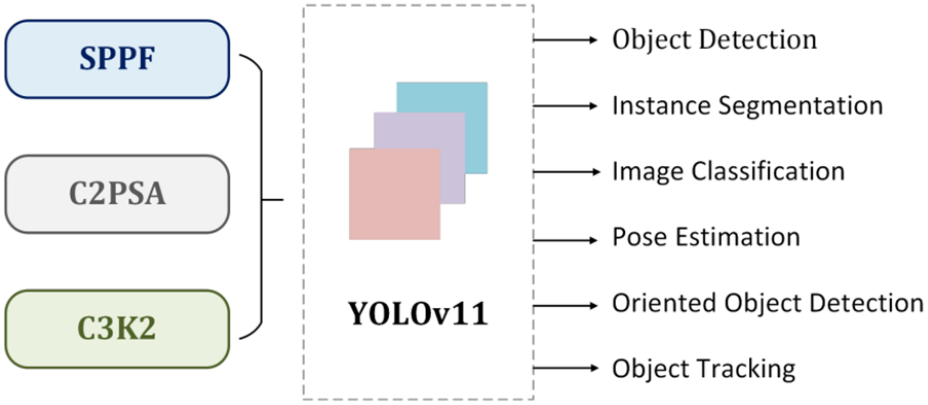


Figure 3.6: SPPF module in YOLOv11

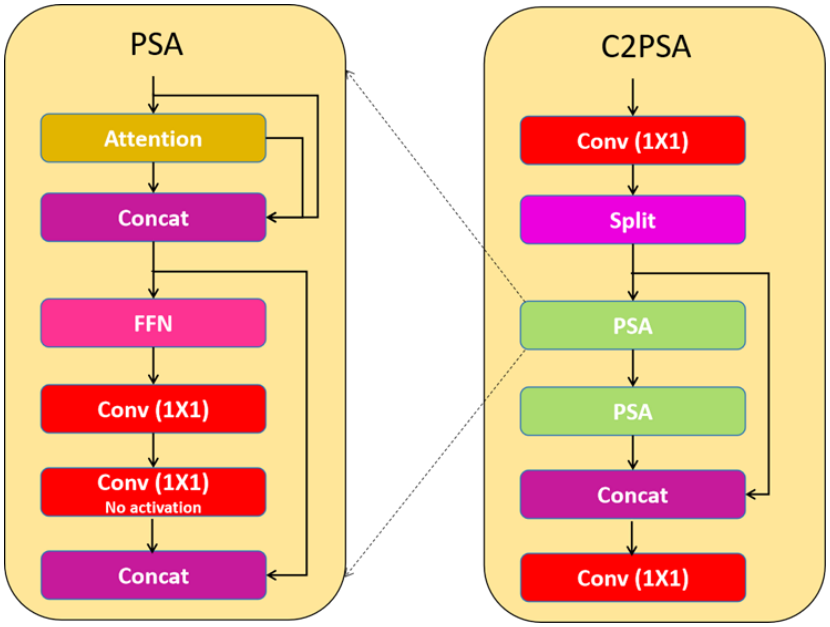


Figure 3.7: C2PSA module in YOLOv11

YOLOv11 reduces computational demands while maintaining high accuracy, offering scalable variants (Nano to Extra-Large) suitable for deployment across edge devices and high-performance systems. Its architecture, detailed in Figure 3.4, supports robotics, healthcare, and security applications, with particular efficacy in medical imaging due to its precision and speed. These advancements position YOLOv11 as a versatile platform, extending beyond object detection to a comprehensive computer vision ecosystem.

3.3 Network Components and Activation Functions

This study aims to enhance the performance of YOLOv8 and YOLOv11 by optimizing parameter efficiency and improving evaluation metrics through targeted experimentation with network modules and activation functions. Specifically, we evaluate the Ghost Module, Depthwise Convolution, Mish, and GELU, recognized for their ability to refine feature extraction and inference efficiency in **Convolutional Neural Networks (CNNs)**, critical for real-time object detection in complex datasets.

3.3.1 Ghost Module

Optimizing computational efficiency in CNNs while preserving accuracy remains a persistent challenge, particularly for real-time applications. Ghost Convolution (GhostConv), introduced by Han et al. (2020) [10], addresses this by reducing parameter and computational demands without sacrificing feature extraction efficacy.

Premised on the redundancy within standard convolution feature maps, GhostConv employs a two-stage process: (i) a primary convolution computes a subset of output channels, and (ii) a secondary stage generates additional channels via lightweight transformations, such as depthwise convolution.

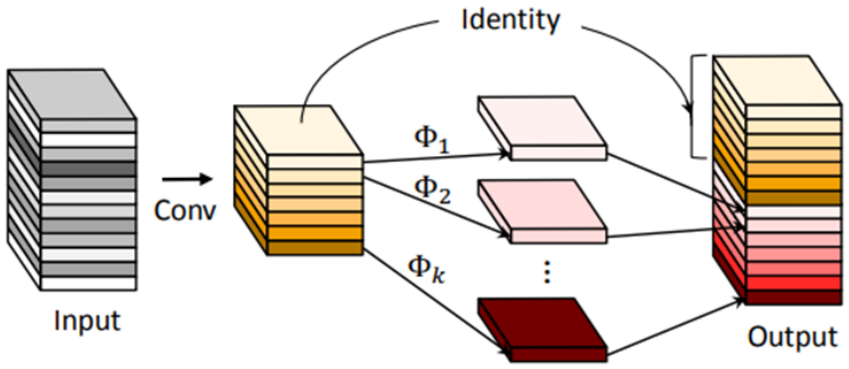


Figure 3.8: Schematic of Ghost Convolution

Figure 3.8 illustrates this mechanism, which achieves equivalent feature map dimensionality to standard convolution with substantially fewer operations. For YOLOv8 and YOLOv11, designed for rapid inference, GhostConv reduces computational complexity by approximately 50%, enhancing deployment on resource-constrained edge devices like NVIDIA Jetson and Qualcomm Snapdragon platforms. This efficiency gain maintains detection accuracy, making it an effective substitute for traditional convolution layers in deep architectures.

Beyond resource optimization, GhostConv mitigates overfitting by leveraging a reduced parameter set to learn diverse features, improving model generalization, particularly in data-scarce scenarios. This robustness is vital for surgical tool detection, where variability in imaging conditions prevails. Our evaluation of GhostConv within YOLOv8 and YOLOv11 seeks to balance performance and computational cost, potentially advancing real-time object detection efficacy in medical imaging applications.

3.3.2 Depthwise Convolution

For lightweight models optimized for mobile devices and embedded systems, Depthwise Convolution has become a key technique for minimizing parameter count and computational costs while preserving effective feature extraction. This method has been widely adopted in architectures such as MobileNet, EfficientNet, and YOLO to improve image processing efficiency without significantly compromising accuracy. With the continued development of YOLOv8 and YOLOv11, experimenting with replacing standard convolution layers with Depthwise Convolution could lead to

faster inference speeds, reduced memory consumption, and improved deployability on edge devices, all while maintaining high accuracy. Depthwise Convolution is an efficient variation of standard convolution, designed to reduce computational complexity while preserving essential feature extraction capabilities. Unlike conventional convolution operations, where a single filter is applied across all input channels, Depthwise Convolution processes each channel independently using a separate filter. This specialized approach significantly reduces the number of multiplications and additions, leading to improved computational efficiency without compromising accuracy.

The Depthwise Convolution process consists of two primary steps. In the first stage, known as the backbone feature extraction step, each input channel undergoes convolution with a unique filter instead of applying a shared filter across all channels. This channel-wise processing allows the model to efficiently extract distinct spatial patterns while significantly lowering the computational cost compared to standard convolution. By reducing redundant operations, Depthwise Convolution enhances the model's ability to process high-dimensional feature maps with minimal latency. Following the Depthwise Convolution step, the extracted feature maps must be combined and integrated to form a more meaningful representation. This is achieved through Pointwise Convolution (1×1 Convolution), where a 1×1 filter is applied across the output channels to learn inter-channel dependencies and reconstruct a complete feature representation. This step is crucial, as it ensures that spatial information extracted during Depthwise Convolution is effectively reorganized and refined for subsequent layers.

By separating spatial and channel-wise computations, Depthwise Convolution not only enhances computational efficiency but also reduces memory usage, making it an ideal choice for resource-constrained environments such as mobile devices, embedded systems, and real-time computer vision applications. Its integration into modern deep learning architectures, including YOLO models and lightweight CNNs, has proven to be highly effective in maintaining a balance between speed, accuracy, and efficiency.

The formulas for computing the number of operations in Standard Convolution and Depthwise Separable Convolution clearly illustrate the computational efficiency of the latter:

Standard Convolution:

$$FLOPs = H \times W \times C_{in} \times C_{out} \times K^2 \tag{3.3}$$

Depthwise Separable Convolution:

$$FLOPs = H \times W \times (C_{in} \times K^2 + C_{in} \times C_{out}) \tag{3.4}$$

where H, W are the input dimensions, C_{in}, C_{out} are the number of input and output channels, and K is the kernel size.

Depthwise Convolution is an efficient method for reducing computational costs while maintaining high object detection performance. Integrating Depthwise Convolution can enhance inference speed, deployment efficiency on edge devices, and hardware resource optimization.

3.3.3 Mish Function

In deep neural networks (DNNs), the choice of activation function critically influences training dynamics and generalization. Mish, introduced by Misra (2019) [12], is a smooth, non-monotonic function designed to enhance these properties, defined as:

$$\text{Mish}(x) = x \cdot \tanh(\text{softplus}(x)), \quad \text{where} \quad \text{softplus}(x) = \ln(1 + e^x). \tag{3.5}$$

Figure 3.9 illustrates its profile, highlighting its continuous differentiability, unbounded range, and stable gradient characteristics, which distinguish it from traditional functions like ReLU and SiLU.

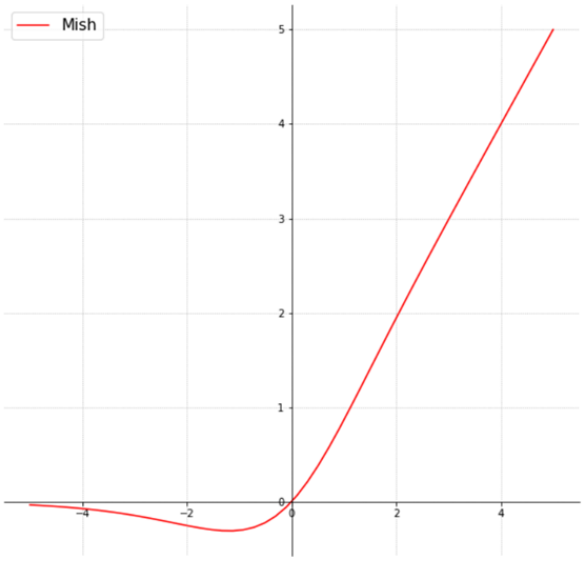


Figure 3.9: Profile of the Mish activation function

Mish mitigates the vanishing gradient issue prevalent in deep architectures by ensuring smooth gradient flow, contrasting with ReLU's abrupt zeroing of negative inputs. This smoothness, evidenced by its first and second derivatives in Figure 3.10, facilitates efficient optimization and faster convergence. Unlike ReLU, which discards

negative values, Mish retains them, preserving richer feature representations and enhancing expressiveness—an advantage over SiLU's bounded behavior. This property proves particularly beneficial in tasks requiring fine-grained feature extraction, such as medical image analysis, where subtle details are paramount.

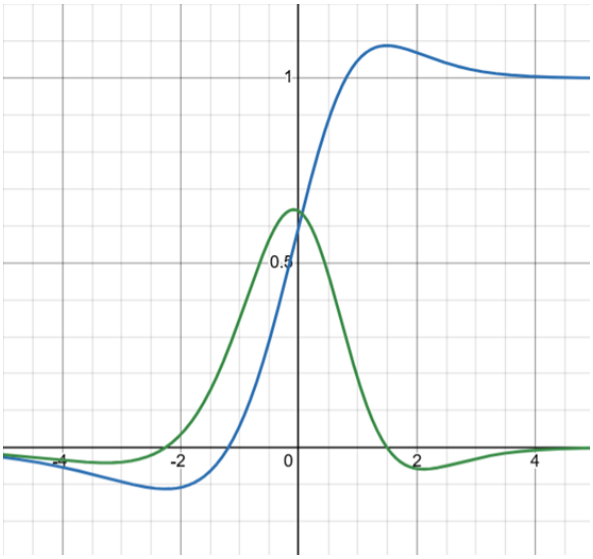


Figure 3.10: First and second derivatives of the Mish function

Integration of Mish into YOLOv8 and YOLOv11 enhances detection accuracy without additional computational cost. Experimental evidence from YOLOv4 demonstrates a 2–3% increase in mean Average Precision (mAP) when replacing Leaky ReLU with Mish, a gain attributed to improved gradient stability and feature retention [12]. In YOLOv11, with its complex multi-layer design, Mish's ability to maintain positive gradients for negative inputs reduces gradient starvation, optimizing backpropagation and boosting robustness for small or occluded object detection in endoscopic imagery. These improvements position Mish as a superior alternative to conventional activation functions, enhancing precision and training efficiency in real-time vision applications.

3.3.4 GELU Function

The activation function governs convergence, learning efficiency, and accuracy in deep neural networks (DNNs). The Gaussian Error Linear Unit (GELU), proposed by Hendrycks and Gimpel (2016) [13], introduces a smooth, unbounded, and adaptive non-linear transformation, defined as:

$$\mathrm{GELU}(x) = x\Phi(x) = x \cdot \frac{1}{2}\left[1 + \mathrm{erf}\left(\frac{x}{\sqrt{2}}\right)\right], \tag{3.6}$$

where $\Phi(x)$ is the cumulative distribution function of the standard Gaussian, and $\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\, dt$ is the error function.

Approximations include $0.5x\left(1+\tanh\left[\sqrt{\frac{2}{\pi}}(x+0.044715x^3)\right]\right)$ and $x\sigma(1.702x)$, with σ as the sigmoid function.

Figure 3.11 depicts its profile, showcasing its continuous differentiability.

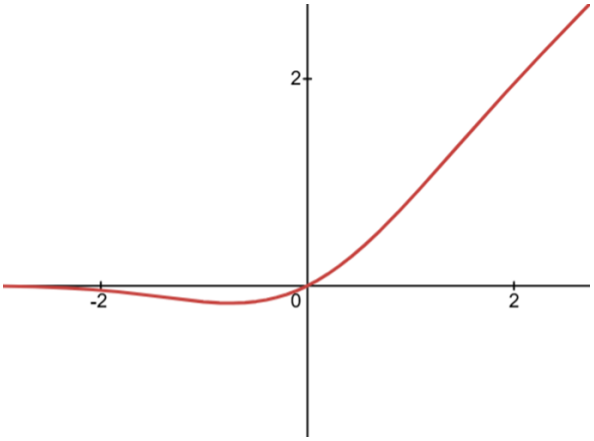


Figure 3.11: Profile of the GELU activation function

GELU's adaptive non-linearity, driven by the Gaussian weighting, adjusts dynamically to input magnitude, outperforming ReLU's fixed thresholding and SiLU's bounded output. This adaptability, coupled with its unbounded range, preserves strong feature representations, avoiding ReLU's loss of negative inputs. Its smoothness, illustrated by the first and second derivatives in Figure 3.12, enhances gradient flow compared to ReLU and Swish, mitigating vanishing gradient issues and accelerating convergence in deep architectures, notably Transformers and CNNs.

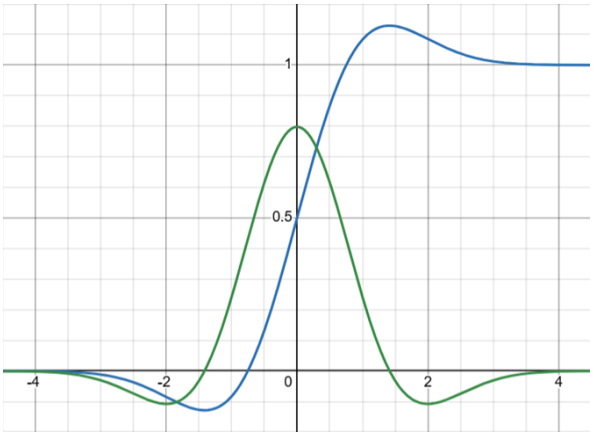


Figure 3.12: First and second derivatives of the GELU function

In YOLOv8 and YOLOv11, integrating GELU into the backbone, neck, and head enhances object detection precision, particularly for small or occluded objects in endoscopic imagery. Experimental evidence from Transformer models like BERT and

ViT demonstrates GELU's superior accuracy over ReLU on large-scale datasets such as ImageNet [13], a benefit attributed to its nuanced feature retention. In YOLO contexts, GELU improves mean Average Precision (mAP) by leveraging adaptive nonlinearity, outperforming SiLU in fewer epochs, thus optimizing training efficiency without added computational cost. This positions GELU as a compelling alternative for real-time detection tasks requiring high accuracy and robustness.

3.4 Proposed Model Architectures

This study proposes enhancements to the YOLOv8 and YOLOv11 frameworks by integrating optimized modules and activation functions to reduce parameter count while improving detection accuracy and inference efficiency in real-time applications. Three distinct architectural modifications are evaluated: YOLO-Ghost, YOLO-Depthwise Convolution, and YOLO-Mish/GELU, each targeting specific components of the original models to address computational complexity and performance trade-offs.

3.4.1 YOLO-Ghost Model

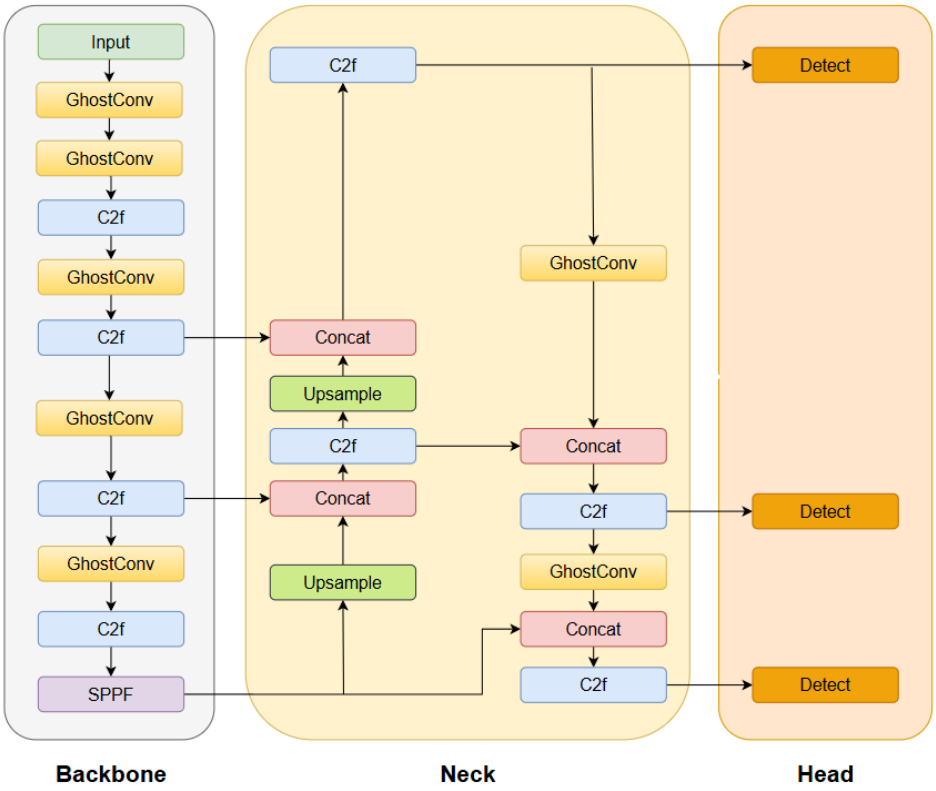


Figure 3.13: YOLOv8 with GhostConv backbone

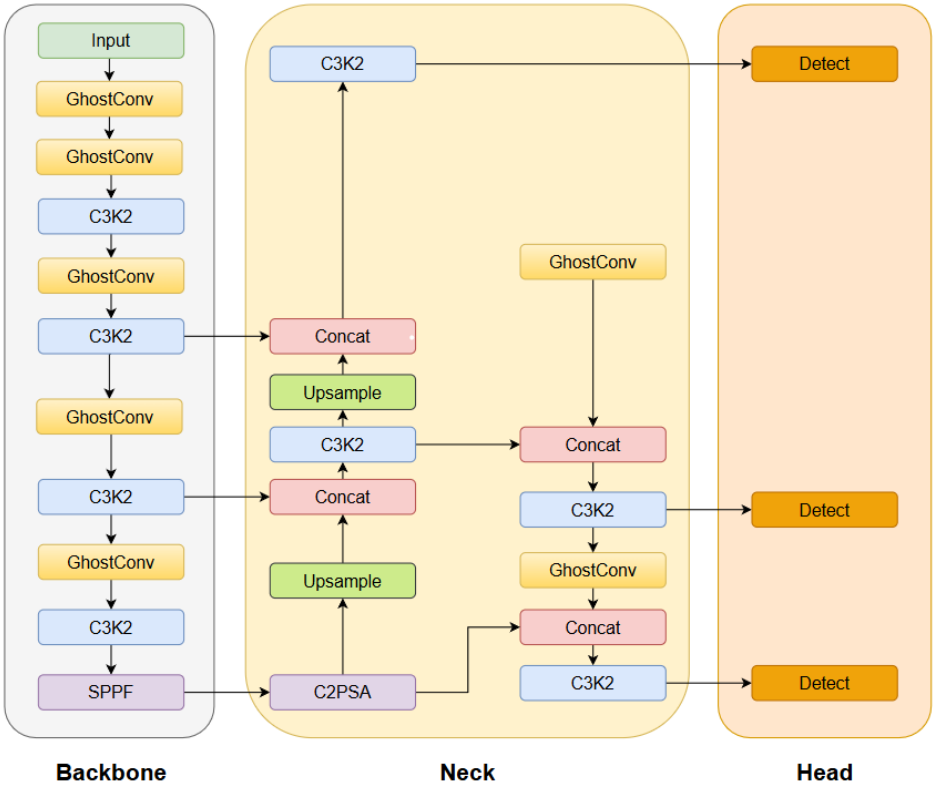


Figure 3.14: YOLOv11 with GhostConv backbone

To enhance computational efficiency in YOLOv8 and YOLOv11, we incorporate the Ghost Module through two experimental configurations. In the first trial, the backbone's Conv modules are substituted with GhostConv modules, reducing parameter redundancy while preserving feature extraction efficacy, yielding YOLOv8-GhostConv and YOLOv11-GhostConv, as depicted in Figures 3.13 and 3.14, respectively.

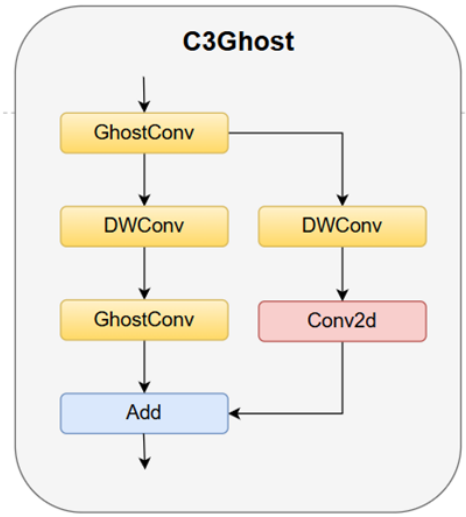


Figure 3.15: Structure of the C3Ghost module

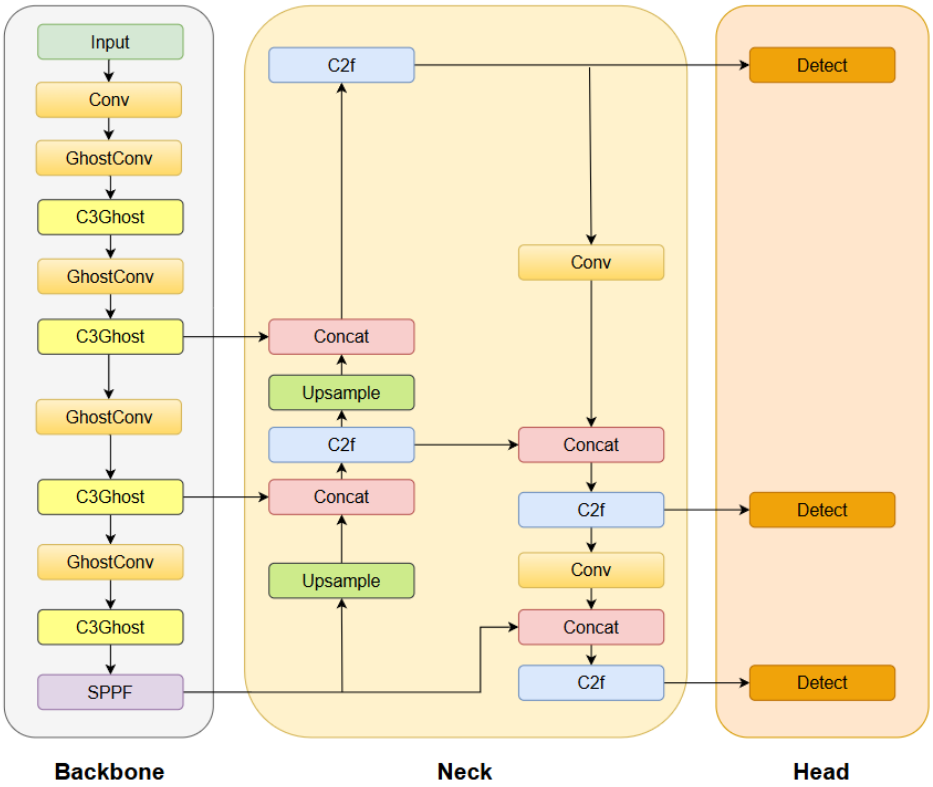


Figure 3.16: YOLOv8 with C3Ghost backbone

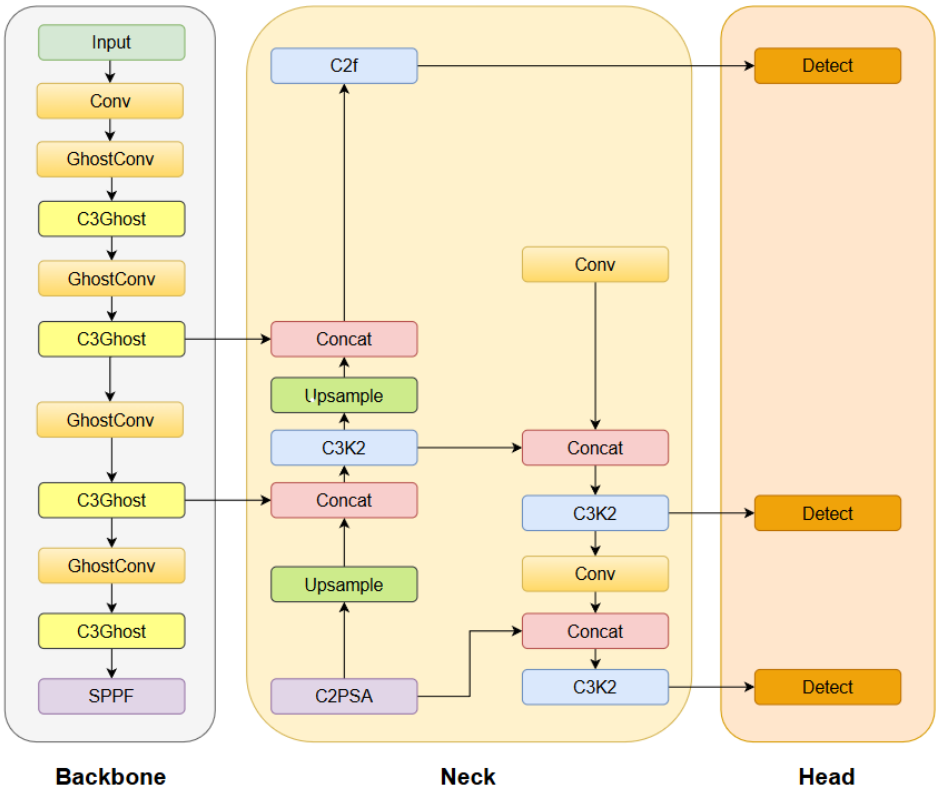


Figure 3.17: YOLOv11 with C3Ghost backbone

In the second trial, we extend this modification by replacing both Conv modules with GhostConv and C3K2 modules with C3Ghost modules, resulting in YOLOv8-

C3Ghost and YOLOv11-C3Ghost, shown in Figures 3.16 and 3.17. The C3Ghost module, illustrated in Figure 3.15, integrates a Depthwise (DW) convolution between two GhostConv layers, augmented by a shortcut pathway (DWConv followed by Conv2D), substantially reducing parameters while enhancing feature richness via skip connections. These backbone-focused modifications retain the original neck and head structures to preserve core YOLO characteristics, optimizing inference speed for resource-constrained environments.

3.4.2 YOLO-Depthwise Convolution

The YOLO-Depthwise Convolution model modifies the backbone of YOLOv8 and YOLOv11 by replacing Conv modules with Depthwise Convolution (DWConv), significantly reducing parameter count and computational load. This substitution yields YOLOv8-DWConv and YOLOv11-DWConv, depicted in Figures 3.18 and 3.19, respectively. DWConv's channel-wise processing minimizes operations compared to standard convolution, enhancing inference efficiency while maintaining detection performance, making it suitable for real-time surgical tool recognition where computational resources are limited.

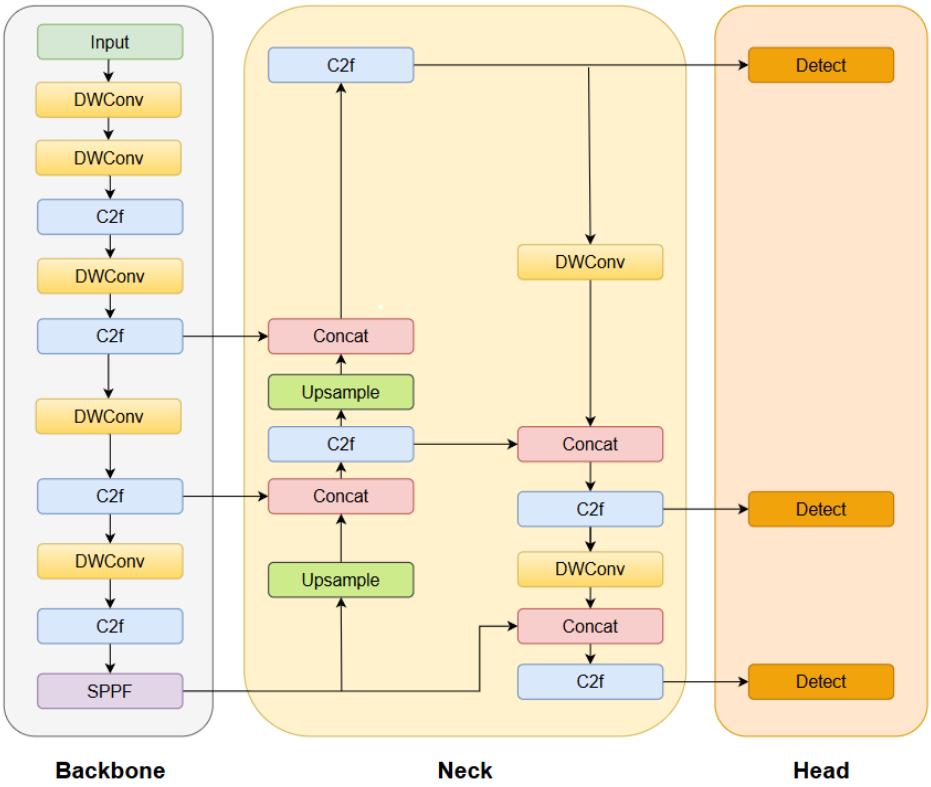


Figure 3.18: YOLOv8 with DWConv backbone

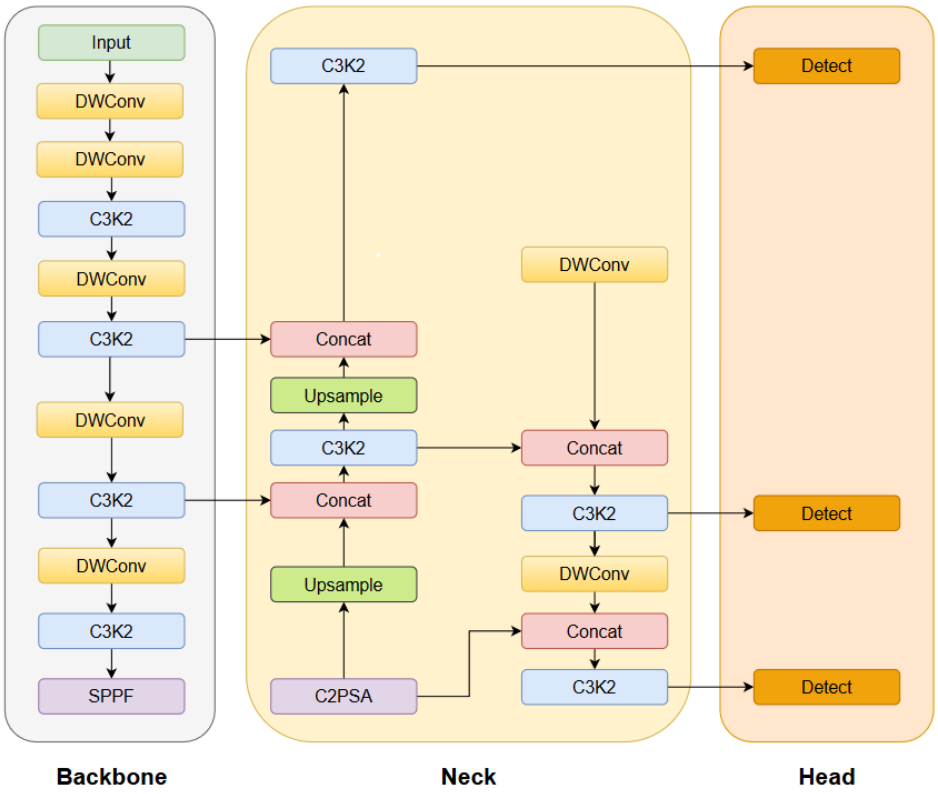


Figure 3.19: YOLOv11 with DWConv backbone

3.4.3 YOLO-Mish and YOLO-GELU

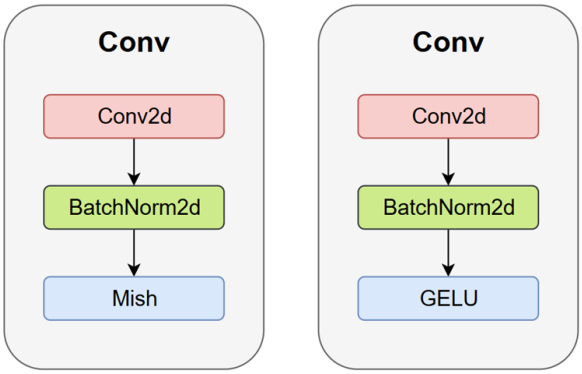


Figure 3.20: Conv module with Mish and GELU activation functions

To improve gradient flow and feature learning in YOLOv8 and YOLOv11, we propose replacing the SiLU activation function within Conv modules with Mish and GELU, creating YOLO-Mish and YOLO-GELU variants. Figure 3.20 illustrates this modification, where the original Conv2D-BatchNorm2D-SiLU structure is adapted to incorporate Mish or GELU. These activation functions, known for their smooth gradients and adaptive non-linearity, enhance convergence speed and detection accuracy without altering the backbone or neck architecture, preserving the models' foundational design while optimizing performance for complex object detection tasks.

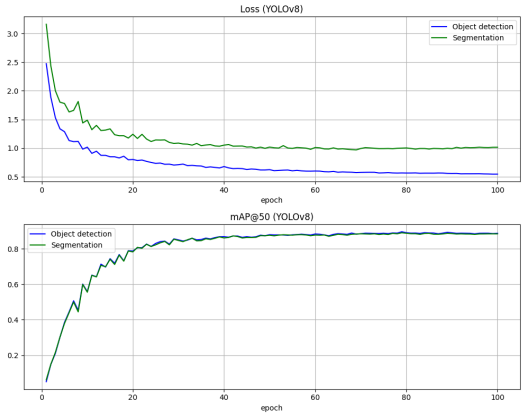
3.5 Model Training

This study evaluates the performance of YOLOv8 and YOLOv11 variants for surgical instrument detection and segmentation through a structured training process. To optimize efficiency, training is conducted in parallel using a local RTX-3070 GPU and Google Colab Pro's A1 GPU, a cloud-based platform enabling scalable computation without local setup requirements. Leveraging this hybrid approach accelerates convergence by balancing local and cloud resources. Each model—YOLOv8 and YOLOv11—offers five variants (nano, small, medium, large, extra-large), differing in parameter scale yet retaining identical architectures. For computational efficiency and experimental consistency, we adopt the small (s) variants, balancing accuracy and resource demands.

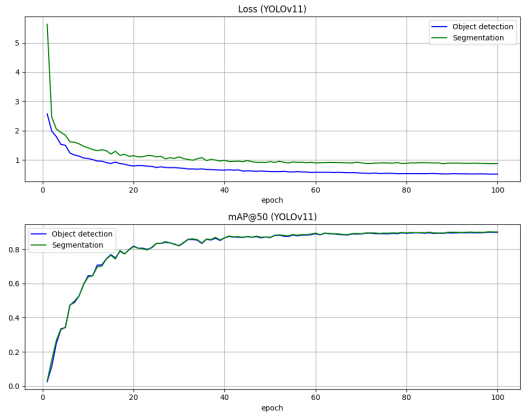
Hyperparameters are standardized to ensure reproducible outcomes: 100 epochs, batch size of 16, input image size of 640×640 pixels, seed value of 0, dropout rate of 0.0, IoU threshold of 0.7, initial learning rate (lr_0) of 0.01, final learning rate (lrf) of 0.01, momentum of 0.937, and weight decay of 0.0005. A warm-up phase is implemented with 3 epochs, momentum of 0.8, and bias learning rate of 0.1, facilitating gradual parameter adaptation. The loss function is tuned with box loss at 7.5, class loss at 0.5, distribution focal loss (DFL) at 1.5, pose loss at 12.0, key object loss (kobj) at 1.0, and synchronization batches (nbs) at 64, optimizing both detection and segmentation precision.

Data augmentation enhances robustness to endoscopic image variability, employing hue shift (hsv_h) of 0.015, saturation shift (hsv_s) of 0.7, and value shift (hsv_v) of 0.4. These settings mitigate color inconsistencies, improving generalization to real-world surgical scenarios. This configuration ensures high accuracy and computational efficiency, with validation loss and mAP@50 metrics tracked across epochs, providing insights into model convergence and performance.

During training, we obtain line graphs depicting loss values and mAP@50 based on the validation set for each model, as shown below:

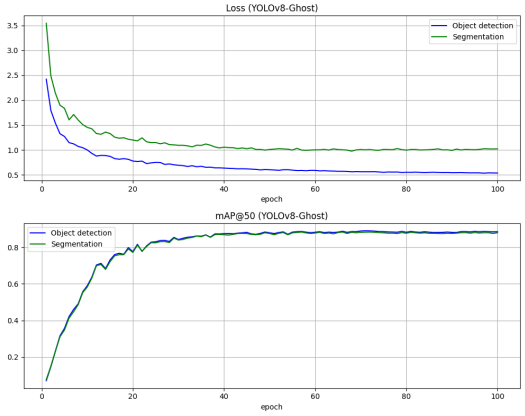


(a) YOLOv8

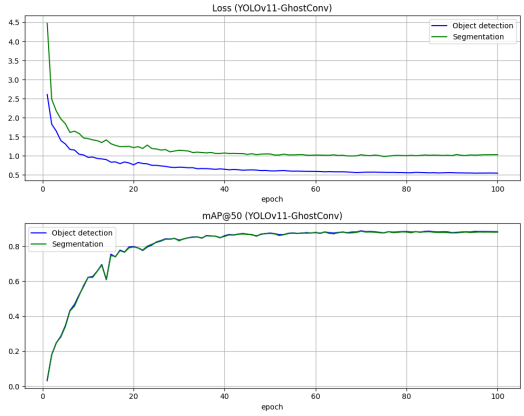


(b) YOLOv11

Figure 3.21: YOLOv8 and YOLOv11 training process

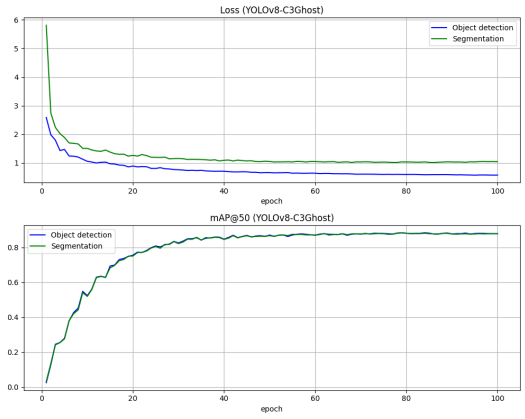


(a) YOLOv8-Ghost

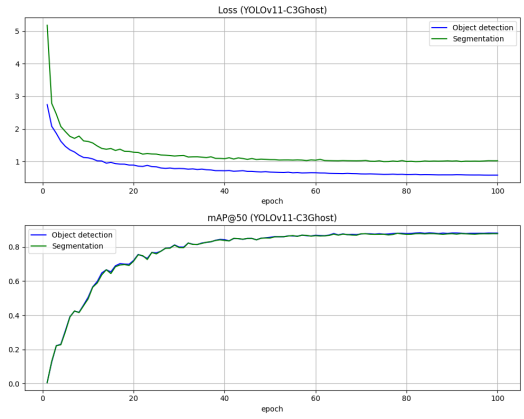


(b) YOLOv11-Ghost

Figure 3.22: YOLOv8-Ghost and YOLOv11-Ghost training process

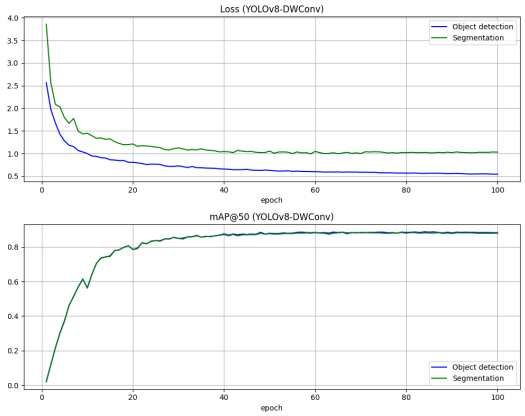


(a) YOLOv8-C3Ghost

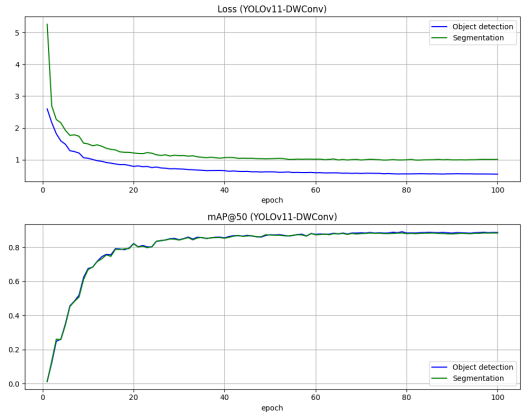


(b) YOLOv11-C3Ghost

Figure 3.23: YOLOv8-C3Ghost and YOLOv11-C3Ghost training process

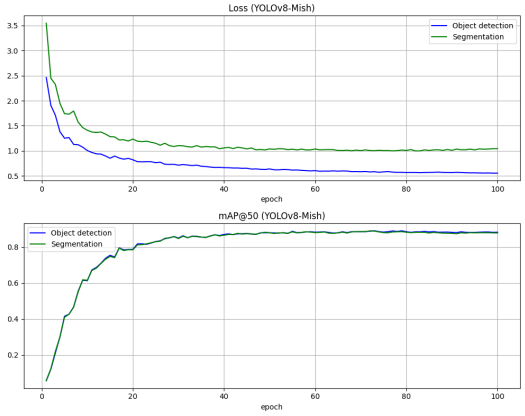


(a) YOLOv8-DWConv

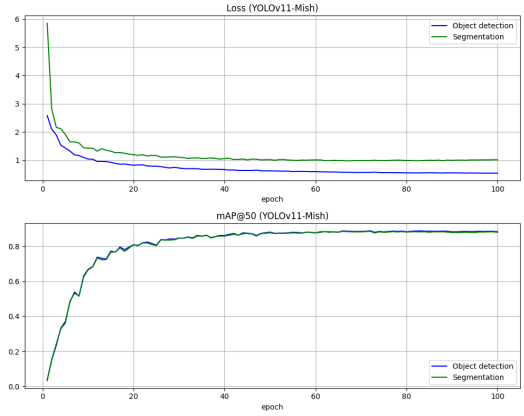


(b) YOLOv11-DWConv

Figure 3.24: YOLOv8-DWConv and YOLOv11-DWConv training process

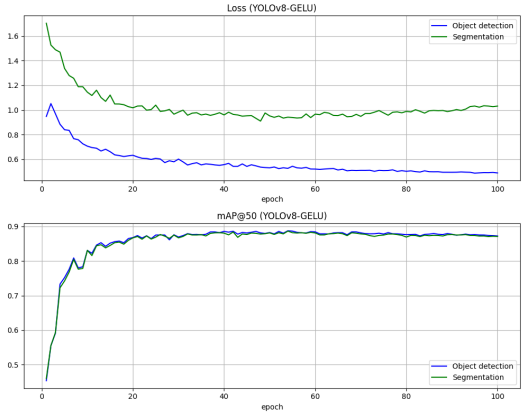


(a) YOLOv8-Mish

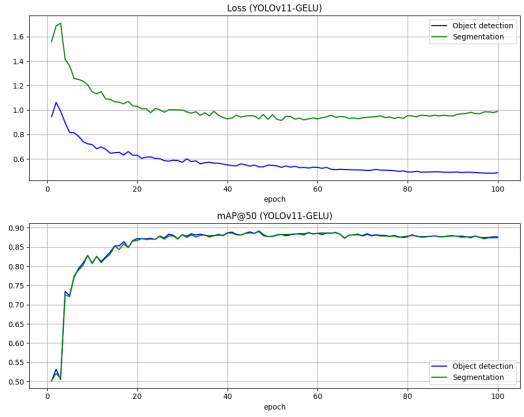


(b) YOLOv11-Mish

Figure 3.25: YOLOv8-Mish and YOLOv11-Mish training process



(a) YOLOv8-GELU



(b) YOLOv11-GELU

Figure 3.26: YOLOv8-GELU and YOLOv11-GELU training process

3.6 Evaluation Metrics

Evaluating the efficacy of object detection models is critical for validating their performance in detecting surgical instruments within endoscopic imagery. This study employs Precision, Recall, Mean Average Precision (mAP), and Frames Per Second (FPS) to assess the proposed YOLOv8 and YOLOv11 variants, providing a comprehensive framework to measure accuracy, coverage, and real-time applicability in surgical contexts. Precision, quantifying the accuracy of instrument predictions, is defined as:

$$P = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{3.7}$$

where TP represents correctly identified tools and FP denotes erroneous detections. High precision ensures minimal false positives, vital for reliable tool identification during surgery, though it must be complemented by Recall to address missed instruments.

Recall, evaluating the model's ability to detect all surgical instruments present, is calculated as:

$$R = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{3.8}$$

where FN indicates undetected tools. Elevated recall signifies comprehensive coverage in operative settings, yet it may increase false positives, necessitating a balanced assessment with Precision. Mean Average Precision (mAP) integrates these metrics, offering a holistic evaluation of detection accuracy. mAP@50, measured at an Intersection over Union (IoU) threshold of 0.5, defined as:

$$\text{IoU} = \frac{\text{Intersection}}{\text{Union}}, \tag{3.9}$$

reflects precision at moderate localization. For a more stringent evaluation, mAP@50:95 (COCO mAP) computes the mean across IoU thresholds from 0.50 to 0.95 in 0.05 increments:

$$\text{mAP}_{50:95} = \frac{1}{10} \sum_{\text{IoU}=0.50}^{0.95} \text{mAP}_{\text{IoU}}, \tag{3.10}$$

assessing robustness across diverse localization demands, such as varying instrument sizes and occlusions in surgical scenarios.

Frames Per Second (FPS) measures inference speed by counting frames processed per second, a critical factor in real-time performance for intraoperative tool tracking, where delays affect surgical precision. Alongside Precision and Recall for detection fidelity, mAP for localization and classification accuracy, and FPS for operational efficiency, these metrics ensure a thorough evaluation of enhanced YOLO models, optimizing instrument detection in surgical applications.

Chapter 4

Experimental Results

This chapter presents experimental outcomes, encompassing both qualitative and quantitative evaluations of the proposed YOLOv8 and YOLOv11 variants on our instance segmentation our test dataset. Additionally, it includes an analysis of surgical skill assessment based on instrument detection results, providing insights into model performance and practical utility in real-world surgical scenarios. These findings validate the efficacy of the enhanced architectures and inform their applicability for intraoperative tool tracking and skill evaluation.

4.1 Inference Speed Assessment

Inference speed is a critical determinant of practical utility for surgical instrument detection models in endoscopic procedures. This study assesses the processing efficiency of YOLOv8 and YOLOv11 using Frames Per Second (FPS) on an NVIDIA RTX 3070 GPU with the our dataset, trained over 100 epochs, requiring approximately 2 hours per model. Two inference conditions are evaluated: offline processing of pre-recorded endoscopic videos and real-time processing of live camera feeds.

Offline inference yields mean FPS values of 85.2 ± 2.1 for YOLOv8 and 78.4 ± 1.8 for YOLOv11, demonstrating robust performance on static data. Real-time inference, however, results in reduced FPS—81.3 ± 2.0 for YOLOv8 and 75.6 ± 1.7 for YOLOv11—owing to additional computational overhead from camera sampling, image preprocessing, and data transfer latency. These findings, expressed with standard deviations to reflect variability across runs, indicate that both models sustain high inference rates suitable for intraoperative applications, with YOLOv8 exhibiting a marginal advantage in processing speed. The observed real-time performance decrement aligns with expected system constraints, affirming the models' efficacy for time-sensitive surgical tool tracking despite operational challenges.

4.2 Quantitative Results

This section evaluates the performance of YOLOv8, YOLOv11, and their optimized variants (GhostConv, C3Ghost, DWConv, Mish, GELU) in detecting and segmenting surgical instruments using the M2CAI16-Tool dataset. Precision (P), Recall (R), mean Average Precision (mAP@0.5, mAP@0.5:0.95), and parameter count are assessed to measure accuracy, coverage, and computational efficiency, vital for real-time surgical applications. Detection and instance segmentation results are reported in Tables 4.1 and 4.2, comparing YOLOv8-based and YOLOv11-based models sequentially.

Table 4.1: Detection performance metrics of YOLOv8 and YOLOv11 variants

Model	**Params (M)**	**P**	**R**	**mAP@0.5**	**mAP@0.5:0.95**
YOLOv8	11.8	**0.888**	0.855	0.902	0.801
YOLOv8-Ghost	10.6	0.862	0.875	0.899	0.799
YOLOv8-C3Ghost	8.9	0.871	0.853	0.891	0.779
YOLOv8-DWConv	9.5	0.863	0.851	0.894	0.786
YOLOv8-Mish	11.8	0.861	0.868	0.903	0.791
YOLOv8-GELU	11.8	**0.888**	0.840	0.898	0.803
YOLOv11	10.1	**0.888**	0.846	0.900	0.804
YOLOv11-Ghost	8.7	0.887	0.860	0.904	0.800
YOLOv11-C3Ghost	7.9	0.879	0.846	0.887	0.778
YOLOv11-DWConv	**7.4**	0.869	0.863	0.906	**0.806**
YOLOv11-Mish	10.1	0.872	0.865	0.900	0.801
YOLOv11-GELU	10.1	0.857	**0.886**	**0.910**	**0.821**

Detection results on our testing dataset are reported in Table 4.1. YOLOv8, with 11.8M parameters, achieves a baseline mAP@0.5:0.95 of 0.801, while YOLOv11, at 10.1M parameters, slightly improves to 0.804. Among variants, YOLOv11-GELU outperforms with mAP@0.5:0.95 = 0.821, benefiting from GELU's robust gradient flow, though its recall trails YOLOv8-Ghost (10.6M). YOLOv11-DWConv, with a minimal 7.4M parameters (27% reduction), delivers mAP@0.5:0.95 = 0.806, balancing efficiency and accuracy. In contrast, YOLOv8-C3Ghost and YOLOv11-C3Ghost (8.9M and 7.9M) record lower mAP@0.5:0.95 values of 0.779 and 0.778, reflecting a trade-off in localization precision for reduced parameters.

Instance segmentation results on the same dataset are detailed in Table 4.2. YOLOv8 achieves mAP@0.5:0.95 of 0.798 with 11.8M parameters, while YOLOv11-GELU leads at 0.802 with 10.1M, improving recall and robustness. YOLOv11-DWConv (7.4M) maintains 0.792, whereas YOLOv8-Ghost and YOLOv11-Ghost (10.6M and

8.7M) achieve 0.790–0.793 with 10–14% fewer parameters. YOLOv8-C3Ghost and YOLOv11-C3Ghost (8.9M and 7.9M) show the lowest mAP@0.5:0.95 at 0.775 and 0.772, reflecting reduced segmentation accuracy.

YOLOv11-GELU's enhanced mAP@0.5:0.95 (0.821 detection, 0.802 instance segmentation) at 10.1M parameters optimizes detection under occlusion, while YOLOv11-DWConv's 7.4M parameters prioritize efficiency with competitive accuracy. GhostConv variants balance both, and C3Ghost sacrifices precision, guiding model selection for intraoperative tracking based on accuracy or resource constraints.

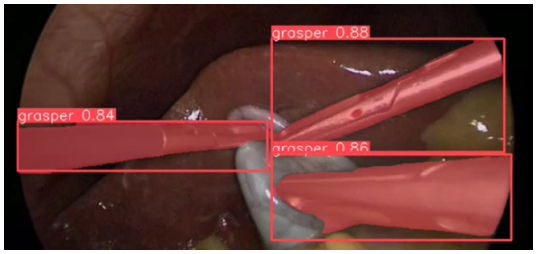
Table 4.2: Instance segmentation performance metrics of YOLOv8 and YOLOv11 variants

Model	**Params (M)**	**P**	**R**	**mAP@0.5**	**mAP@0.5:0.95**
YOLOv8	11.8	**0.892**	0.859	0.906	0.798
YOLOv8-Ghost	10.6	0.862	0.874	0.897	0.790
YOLOv8-C3Ghost	8.9	0.871	0.852	0.891	0.775
YOLOv8-DWConv	9.5	0.871	0.845	0.892	0.783
YOLOv8-Mish	11.8	0.872	0.858	0.905	0.790
YOLOv8-GELU	11.8	0.889	0.842	0.897	0.789
YOLOv11	10.1	0.891	0.850	0.903	0.795
YOLOv11-Ghost	8.7	0.887	0.861	0.903	0.793
YOLOv11-C3Ghost	7.9	0.879	0.846	0.886	0.772
YOLOv11-DWConv	**7.4**	0.875	0.857	0.905	0.792
YOLOv11-Mish	10.1	0.882	0.855	0.898	0.785
YOLOv11-GELU	10.1	0.862	**0.881**	**0.908**	**0.802**

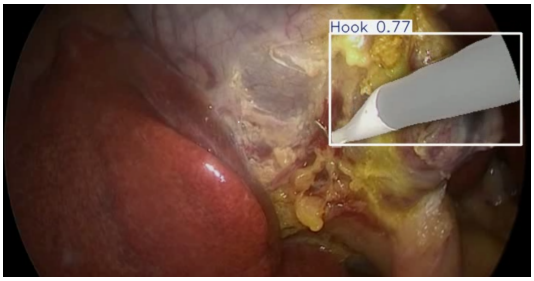
4.3 Qualitative Results

This section presents qualitative outcomes from experiments evaluating surgical instrument detection in endoscopic videos using YOLOv8, YOLOv11, and their variants on the M2CAI16-Tool dataset. Figure 4.1 showcases the models' capability to accurately detect and segment instruments across diverse surgical scenes, despite challenges such as variable lighting and overlapping objects. Examples include precise identification of a Grasper, Hook, Irrigator, and combined Grasper-Specimen Bag instances, demonstrating robust performance in complex environments.

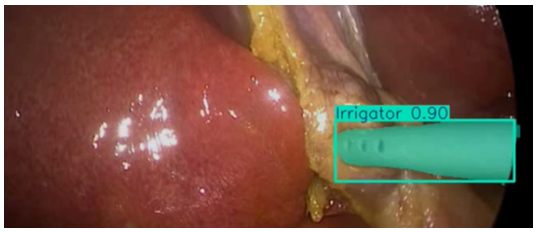
Despite these strengths, misclassification errors persist in certain frames, as illustrated in Figure 4.2. In one instance, a body part is mistaken for a Specimen Bag with a prediction probability of 0.45, while in another, a Hook is incorrectly identified under low-light conditions. These errors stem from imperfect lighting, object overlap,



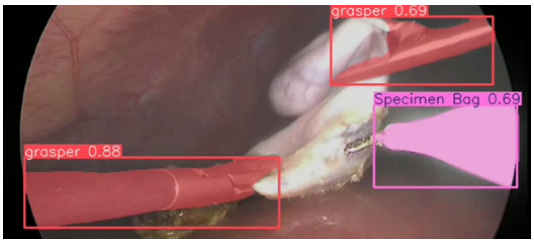
(a) Grasper detection



(b) Hook detection



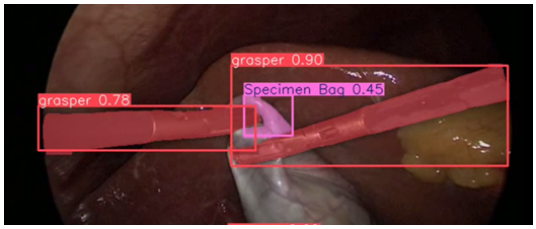
(c) Irrigator detection



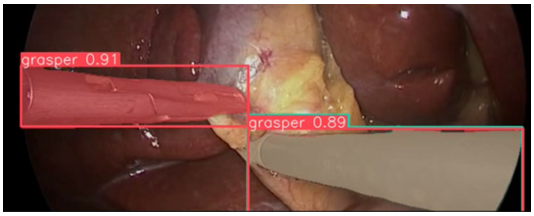
(d) Grasper and Specimen Bag detection

Figure 4.1: Successful detection and segmentation of surgical instruments in endoscopic videos

and visual similarities between instruments and background, complicating accurate prediction. Raising the detection threshold could reduce false positives, though it risks missing true instruments, necessitating a balanced approach to optimize intra-operative tool tracking.



(a) Misclassification of a body part as Specimen Bag (probability = 0.45)



(b) Misclassification of Hook in low-light conditions

Figure 4.2: Misclassification examples in surgical instrument detection

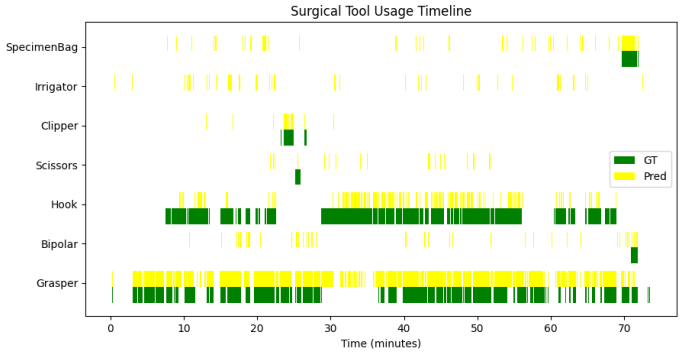
4.4 Evaluation of Surgical Performance

This section assesses surgical performance across five test videos from the M2CAI16-Tool dataset, evaluating instrument detection accuracy and inferring surgeon skill based on tool usage patterns. Detection accuracy is determined by comparing the algorithm's predicted instrument usage (yellow) with the ground truth (GT) presence of instruments, which assesses laparoscopic proficiency through observable metrics such as tool usage frequency, switching patterns, and procedural efficiency derived

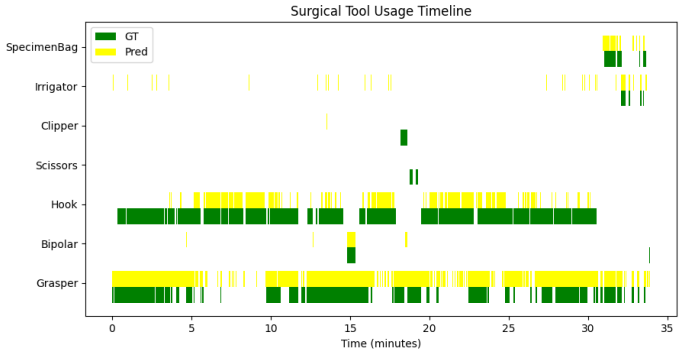
from GT data. However, manual GOALS evaluation is limited by subjectivity, time-intensive annotation, and potential bias in interpreting complex surgical scenarios, necessitating automated approaches for enhanced objectivity. Instrument usage timelines provide a quantitative basis for this analysis, focusing on efficiency and skill inference.

Figure 4.3 presents Surgical Tool Usage Timelines for Videos 1–4, comparing the algorithm's predictions (yellow) with GT labels (green) across instruments (Grasper, Bipolar, Hook, Scissors, Clipper, Irrigator, Specimen Bag). In Video 1 (70 minutes), Grasper usage aligns closely (GT: 85% of frames), while Hook detection shows a slight underestimation (GT: 12%), likely due to occlusion. Video 2 (35 minutes) exhibits high Grasper usage (GT: 78%) and consistent Bipolar-Hook overlap. Video 3 (40 minutes) captures varied Grasper-to-Hook transitions (GT: 15 switches), with minor discrepancies in Irrigator usage. Video 4 (30 minutes) underestimates Specimen Bag presence (GT: 8%), though Grasper usage remains reliable (GT: 80%). These results indicate robust detection, with minor deviations attributed to occlusion and lighting challenges.

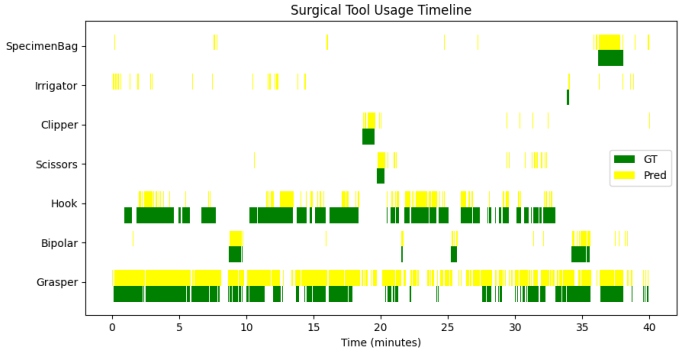
Based on the Surgical Tool Usage Timelines depicted in Figure 4.3, an analysis of surgical skill across the four videos reveals a clear proficiency hierarchy. Video 4 exhibits the most efficient tool utilization, marked by the lowest frequency of tool switching and continuous use of the Hook instrument, with minimal Bipolar application suggesting effective hemostasis or a less invasive approach. Videos 2 and 3 display comparable tool usage patterns, featuring moderate switching frequency, consistent Grasper and Hook dominance, and intermediate Bipolar usage, indicating a balanced yet less optimized technique. In contrast, Video 1 presents a less efficient or potentially more complex scenario, distinguished by the highest tool switching frequency and the broadest instrument range, including the unique use of Scissors. The frequent Bipolar usage in Video 1, compared to others, may reflect increased cauterization needs. Overall, these timelines establish a skill gradient, with Video 4 demonstrating the highest efficiency and focused manipulation, followed by Videos 2 and 3 at an intermediate level, and Video 1 indicating a more complex or less streamlined procedure based on tool usage dynamics.



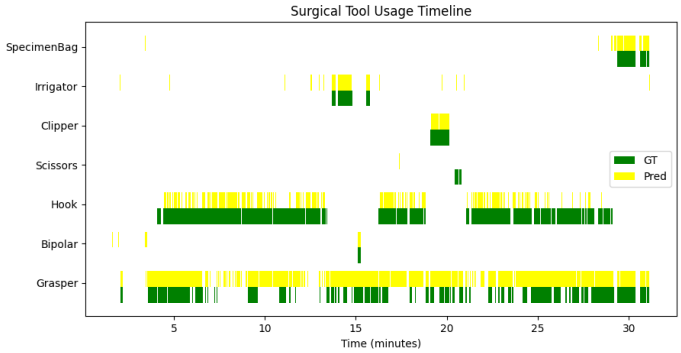
(a) Video 1



(b) Video 2



(c) Video 3



(d) Video 4

Figure 4.3: Surgical Tool Usage Timelines for Videos 1–4 in the M2CAI16-Tool dataset (green: ground truth, yellow: algorithm predictions)

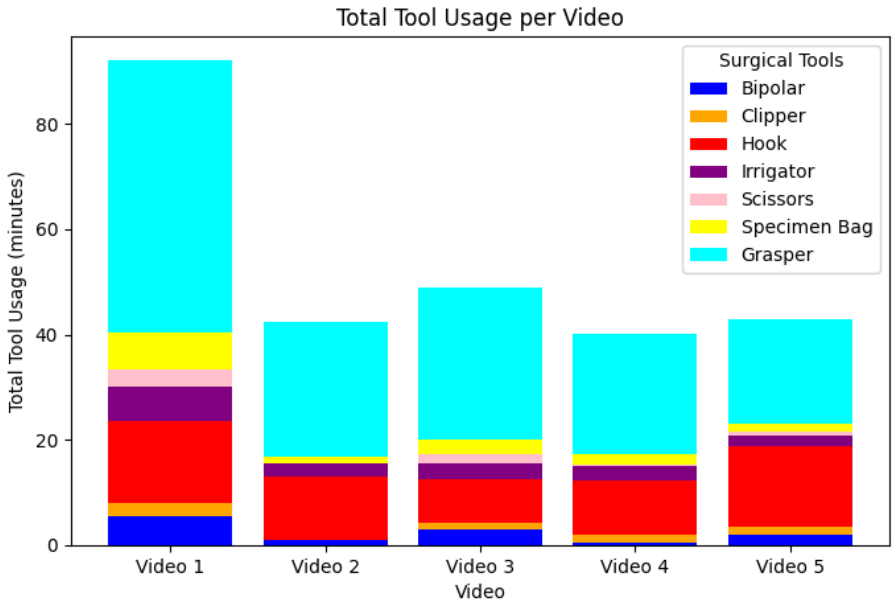


Figure 4.4: Total instrument usage times

Figure 4.4, illustrating the total instrument usage times per video, provides further insights into the surgical skills demonstrated. Video 1 exhibits the highest total tool usage, suggesting a longer or more complex procedure compared to the others. Notably, Video 1 is the only video where Scissors are utilized, indicating a possible need for intricate dissection or tissue manipulation not required in the other videos. While Grasper usage is consistently dominant across all videos, its significantly higher total usage in Video 1 contributes substantially to its extended duration. Videos 2, 3, and 4 show considerably lower and relatively similar total tool usage times, implying more efficient procedures. Video 4, in particular, presents the lowest total usage time, hinting at the most streamlined and potentially efficient surgical execution among the analyzed videos. The consistent use of Hook and Grasper across Videos 2, 3, and 4, coupled with reduced usage of other tools like Bipolar and Irrigator compared to Video 1, suggests a more focused and potentially less interventional surgical approach in these videos, further supporting the inference of varying levels of surgical efficiency and procedural complexity across the analyzed cases.

4.5 Discussion

The experimental findings of this study reveal a nuanced landscape of performance trade-offs within YOLOv8 and YOLOv11, and their optimized variants for surgical instrument detection and segmentation. Our investigation underscores that no single model universally excels; rather, the optimal choice hinges on the specific demands

of the application, particularly the balance between accuracy and computational efficiency.

The YOLOv11-DWConv variant emerges as a compelling solution for resource-constrained settings. Its remarkable 26% parameter reduction compared to the base YOLOv11, coupled with the attainment of the highest detection mAP@0.5 (0.906) and near-top segmentation accuracy, highlights the efficacy of Depthwise Convolution in minimizing computational burden without significantly sacrificing performance. This efficiency gain is paramount for real-world surgical deployment, especially on edge devices where computational resources are limited. Conversely, while the original YOLOv8 achieves the highest segmentation mAP@0.5 (0.906), its larger parameter footprint renders it comparatively more demanding, suggesting a less efficient trade-off for applications where computational constraints are paramount.

The GhostConv and C3Ghost modifications effectively address model size concerns, achieving notable parameter reductions. However, this efficiency comes at the expense of a minor dip in segmentation accuracy, indicating a potential trade-off where fine-grained segmentation precision is critical. These lighter architectures may be particularly advantageous in scenarios prioritizing speed and deployment simplicity over absolute segmentation detail. In contrast, the integration of advanced activation functions, Mish and GELU, offers a distinct advantage in detection performance. Specifically, YOLOv11-GELU achieves the highest overall detection mAP@0.5 (0.91), signifying enhanced object localization capabilities. The elevated Recall exhibited by Mish-integrated models further suggests improved performance in complex scenes with overlapping instruments, potentially due to Mish's superior gradient flow and feature retention.

These results underscore a fundamental principle: surgical instrument detection and segmentation model selection necessitates a careful consideration of application-specific priorities. For scenarios demanding maximal accuracy, even at a higher computational cost, YOLOv11-GELU stands as a robust choice. Conversely, for resource-limited environments where real-time performance is paramount, YOLOv11-DWConv presents an excellent balance of efficiency and accuracy. GhostConv and C3Ghost variants offer further avenues for extreme model compression, albeit with a slight compromise in segmentation precision.

Looking ahead, our findings pave the way for future research to address remaining challenges. Firstly, bridging the accuracy-efficiency gap remains crucial. Further exploration of model compression techniques that minimize accuracy degradation, alongside hardware optimizations for efficient inference, is warranted. Secondly, enhancing robustness to the inherent complexities of endoscopic imagery, including

lighting variations, occlusions, and biological artifacts, remains a critical direction. Future work should investigate advanced data augmentation strategies, adaptive image preprocessing techniques, and model architectures inherently resilient to these challenges. Finally, addressing dataset limitations, particularly regarding size and diversity, is essential for improving model generalizability and clinical applicability. Strategies such as synthetic data generation and multi-institutional data aggregation could significantly enhance model robustness and real-world performance. By tackling these challenges, future iterations of YOLO-based models can realize their full potential to revolutionize surgical workflows, ultimately leading to safer, more efficient, and more precise minimally invasive procedures.

Chapter 5

Conclusion

5.1 Recap of the Main Contributions

This study systematically optimizes and evaluates YOLOv8, YOLOv11, and their variants (GhostConv, C3Ghost, DWConv, Mish, GELU) for surgical instrument detection and segmentation in minimally invasive surgery (MIS), using the M2CAI16-Tool dataset. Through a rigorous experimental framework, we assess processing speed (FPS), detection accuracy (mAP), and real-world deployability, providing insights into balancing accuracy and computational efficiency for AI-driven surgical applications.

The study compares YOLOv8 and YOLOv11, along with their optimized variants, to reveal performance trade-offs. YOLOv11-DWConv reduces parameters by 26% (7.4M) compared to YOLOv11 while maintaining a high detection mAP@0.5 (0.906), demonstrating the effectiveness of Depthwise Convolution for resource-constrained edge devices. YOLOv11-GELU achieves high detection accuracy with mAP@0.5 (0.910) and mAP@0.5:0.95 (0.821) at 10.1M parameters, highlighting GELU's adaptive activation benefits in complex scenes. GhostConv and C3Ghost reduce model size (10.6M to 7.9M) but lower segmentation precision, while Mish improves recall by enhancing gradient flow.

These findings offer significant scientific and practical implications for AI-driven surgical imaging. YOLOv11-DWConv's efficiency enables real-time assistance on edge devices, critical for intraoperative tool tracking, while YOLOv11-GELU's accuracy supports precision-critical tasks. Real-time inference speeds of 81 FPS (video-based) and 75 FPS (live camera) validate practical applicability. The performance evaluation across architectures guides model selection based on application needs, addressing a key challenge in resource-limited surgical environments. Future work should focus on narrowing the accuracy-efficiency gap, enhancing robustness to endoscopic imaging challenges, and expanding dataset diversity to improve clinical generalizability, paving the way for safer and more efficient MIS procedures.

5.2 Limitations and Future Directions

This study, while demonstrating notable advancements in YOLOv8 and YOLOv11 variants for surgical instrument detection and segmentation, identifies several limitations that inform future research. A primary constraint is the trade-off between accuracy and computational efficiency, necessitating the development of advanced compression techniques and hardware-optimized strategies to preserve performance. Additionally, the models' robustness to endoscopic imaging challenges—such as variable lighting, occlusions, and biological artifacts—remains limited, highlighting the need for adaptive illumination correction, occlusion-resilient detection, and enhanced segmentation methods. Lastly, the dataset's restricted size and diversity constrain generalizability, underscoring the importance of expanding data through multi-institutional collaborations and synthetic generation to improve clinical applicability.

These findings affirm the efficacy of the optimized YOLO-based models in balancing accuracy and efficiency for minimally invasive surgery (MIS), laying a foundation for AI-driven surgical assistance. Future research should explore Transformer-based architectures and Self-Supervised Learning (SSL) to enhance feature extraction, alongside Edge AI optimizations for real-time deployment. Addressing these challenges will advance surgical workflows, enabling safer, more efficient, and precise MIS procedures, and ultimately elevating healthcare quality.

Publications

The contributions are supported by publications resulting from this master's thesis research under the *Master of Informatics and Computer Engineering (MICE)* program, guided by Dr. Kim Dinh Thai and Dr. Manh-Hung Ha, providing validations for the proposed YOLO enhancements:

(1) An effective method for detecting personal protective equipment at real construction sites using the improved YOLOv5s with Siou Loss Function. [53]

(2) Emotional inference from speech signals informed by multiple stream DNNs based non-local attention mechanism. [54]

(3) Surgical tool detection and pose estimation using YOLOv8-pose model: A study on clipper tool. [55]

(4) Robust surgical tool detection in laparoscopic surgery using YOLOv8 model. [56]

Within the research group, I was given the valuable opportunity to delve deeply into the YOLO (You Only Look Once) object detection framework. Through collaborative discussions, hands-on experiments, and critical analyses, I was able to gain a comprehensive understanding of how the YOLO model operates, how it balances speed and accuracy, and how it can be adapted to meet various application demands. This exploration was not limited to theoretical learning, but extended to practical implementation, performance evaluation, and the examination of potential areas for enhancement. The supportive academic environment and guidance from experienced mentors allowed me to cultivate both technical knowledge and research skills, which were essential for identifying meaningful directions for improvement. These experiences directly contributed to the development and validation of the proposed YOLO enhancements presented in this thesis.

References

[1] B. Radojcić et al. “History of Minimally Invasive Surgery”. In: *Medicinski pregled* (n.d.). URL: https://pubmed.ncbi.nlm.nih.gov/20491389/.

[2] Wikimedia Foundation. *Erich Mühe*. Wikipedia. Mar. 2024. URL: https://en.wikipedia.org/wiki/Erich_M%C3%BChe.

[3] Ludwig Adams et al. “Computer-assisted surgery”. In: *IEEE Computer graphics and applications* 10.3 (1990), pp. 43–51.

[4] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-net: Convolutional networks for biomedical image segmentation”. In: *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*. Springer. 2015, pp. 234–241.

[5] Liang-Chieh Chen et al. “Rethinking atrous convolution for semantic image segmentation”. In: *arXiv preprint arXiv:1706.05587* (2017).

[6] Kaiming He et al. “Mask r-cnn”. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2961–2969.

[7] Ultralytics. *Home - Ultralytics YOLO Docs*. Online documentation. Feb. 2025. URL: https://docs.ultralytics.com/.

[8] Shubhangi Nema, Abhishek Mathur, and Leena Vachhani. “Plug-in for visualizing 3D tool tracking from videos of Minimally Invasive Surgeries”. In: *arXiv preprint arXiv:2401.09472* (2024).

[9] Shubhangi Nema and Leena Vachhani. “Surgical instrument detection and tracking technologies: Automating dataset labeling for surgical skill assessment”. In: *Frontiers in Robotics and AI* 9 (2022), p. 1030846.

[10] Kai Han et al. “Ghostnet: More features from cheap operations”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 1580–1589.

[11] François Chollet. “Xception: Deep learning with depthwise separable convolutions”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1251–1258.

[12] Diganta Misra. “Mish: A self regularized non-monotonic activation function”. In: *arXiv preprint arXiv:1908.08681* (2019).

[13] Dan Hendrycks and Kevin Gimpel. “Gaussian error linear units (gelus)”. In: *arXiv preprint arXiv:1606.08415* (2016).

[14] Ibrahim Alkatout et al. “The development of laparoscopy—a historical overview”. In: *Frontiers in surgery* 8 (2021), p. 799442.

[15] B. Radojcić et al. “Medicinski pregled”. In: *Medicinski pregled* 62.11-12 (2009), pp. 597–602.

[16] Luca Bertolaccini and Gaetano Rocco. “History and development of minimally invasive surgery: VATS surgery”. In: *Shanghai Chest* 3 (2019).

[17] Mengyu Zhou et al. “A lightweight segmentation network for endoscopic surgical instruments based on edge refinement and efficient self-attention”. In: *PeerJ Computer Science* 9 (2023), e1746.

[18] Wikimedia Foundation. *Thresholding (Image Processing)*. Wikipedia. Aug. 2024. URL: https://en.wikipedia.org/wiki/Thresholding_(image_processing).

[19] Wikimedia Foundation. *Gradient Vector Flow*. Wikipedia. Feb. 2025. URL: https://en.wikipedia.org/wiki/Gradient_vector_flow.

[20] Zijian Wu et al. “Augmenting efficient real-time surgical instrument segmentation in video with point tracking and Segment Anything”. In: *Healthcare Technology Letters* 12.1 (2025), e12111.

[21] Andre Esteva et al. “Dermatologist-level classification of skin cancer with deep neural networks”. In: *nature* 542.7639 (2017), pp. 115–118.

[22] Fabian Isensee et al. “nnu-net: Self-adapting framework for u-net-based medical image segmentation”. In: *arXiv preprint arXiv:1809.10486* (2018).

[23] Christoph Baur et al. *Autoencoders for unsupervised anomaly segmentation in brain MR images: a comparative study*. 2021.

[24] Heang-Ping Chan, Lubomir M Hadjiiski, and Ravi K Samala. “Computer-aided diagnosis in the era of deep learning”. In: *Medical physics* 47.5 (2020), e218–e227.

[25] D. Novotny et al. “Semi-Convolutional Operators for Instance Segmentation”. In: *Lecture Notes in Computer Science*. 2018, pp. 89–105. DOI: 10.1007/978-3-030-01246-5_6.

[26] David Novotny et al. “Semi-convolutional operators for instance segmentation”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, pp. 86–102.

[27] Xiaosong Wang et al. “Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 2097–2106.

[28] Alistair EW Johnson et al. “MIMIC-CXR, a de-identified publicly available database of chest radiographs with free-text reports”. In: *Scientific data* 6.1 (2019), p. 317.

[29] Maria Correia de Verdier et al. “The 2024 Brain Tumor Segmentation (BraTS) challenge: glioma segmentation on post-treatment MRI”. In: *arXiv preprint arXiv:2405.18368* (2024).

[30] Samuel G Armato III et al. “The lung image database consortium (LIDC) and image database resource initiative (IDRI): a completed reference database of lung nodules on CT scans”. In: *Medical physics* 38.2 (2011), pp. 915–931.

[31] Amber L Simpson et al. “A large annotated medical image dataset for the development and evaluation of segmentation algorithms”. In: *arXiv preprint arXiv:1902.09063* (2019).

[32] Andru P Twinanda et al. “Endonet: a deep architecture for recognition tasks on laparoscopic videos”. In: *IEEE transactions on medical imaging* 36.1 (2016), pp. 86–97.

[33] Ravimal Bandara. “Image segmentation using unsupervised watershed algorithm with an over-segmentation reduction technique”. In: *arXiv preprint arXiv:1810.03908* (2018).

[34] Xu Chen et al. “Learning active contour models for medical image segmentation”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 11632–11640.

[35] Zan Li et al. “Residual-attention UNet++: a nested residual-attention U-net for medical image segmentation”. In: *Applied Sciences* 12.14 (2022), p. 7149.

[36] Ozan Oktay et al. “Attention u-net: Learning where to look for the pancreas. arXiv”. In: *arXiv preprint arXiv:1804.03999* 10 (2018).

[37] Özgün Çiçek et al. “3D U-Net: learning dense volumetric segmentation from sparse annotation”. In: *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2016: 19th International Conference, Athens, Greece, October 17-21, 2016, Proceedings, Part II 19*. Springer. 2016, pp. 424–432.

[38] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. “Segnet: A deep convolutional encoder-decoder architecture for image segmentation”. In: *IEEE transactions on pattern analysis and machine intelligence* 39.12 (2017), pp. 2481–2495.

[39] Liang-Chieh Chen et al. “Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs”. In: *IEEE transactions on pattern analysis and machine intelligence* 40.4 (2017), pp. 834–848.

[40] Keisuke Hamamoto et al. “DeepLabv3”. In: *Artificial Intelligence and Robotics: 8th International Symposium, ISAIR 2023, Beijing, China, October 21–23, 2023, Revised Selected Papers*. Springer Nature. 2024, p. 181.

[41] Andru P Twinanda et al. “Single-and multi-task architectures for tool presence detection challenge at M2CAI 2016”. In: *arXiv preprint arXiv:1610.08851* (2016).

[42] Joseph Redmon et al. “You only look once: Unified, real-time object detection”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 779–788.

[43] Joseph Redmon and Ali Farhadi. “YOLO9000: better, faster, stronger”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 7263–7271.

[44] Joseph Redmon and Ali Farhadi. “Yolov3: An incremental improvement”. In: *arXiv preprint arXiv:1804.02767* (2018).

[45] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. “Yolov4: Optimal speed and accuracy of object detection”. In: *arXiv preprint arXiv:2004.10934* (2020).

[46] Glenn Jocher et al. “ultralytics/yolov5: v7. 0-yolov5 sota realtime instance segmentation”. In: *Zenodo* (2022).

[47] Chuyi Li et al. “YOLOv6: A single-stage object detection framework for industrial applications”. In: *arXiv preprint arXiv:2209.02976* (2022).

[48] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. “YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2023, pp. 7464–7475.

[49] Mupparaju Sohan et al. “A review on yolov8 and its advancements”. In: *International Conference on Data Intelligence and Cognitive Informatics*. Springer. 2024, pp. 529–545.

[50] Chien-Yao Wang, I-Hau Yeh, and Hong-Yuan Mark Liao. “Yolov9: Learning what you want to learn using programmable gradient information”. In: *European conference on computer vision*. Springer. 2024, pp. 1–21.

[51] Alyaa Aloraibi, Armaneesa Naaman Hasoon, and Moumena Salah Yassen. “Yolov10: Toward a Promising Improvement of Gun Detection Based on a Proposed Image Enhancement Technique”. In: *Available at SSRN 5029993* ().

[52] Rahima Khanam and Muhammad Hussain. “Yolov11: An overview of the key architectural enhancements”. In: *arXiv preprint arXiv:2410.17725* (2024).

[53] Manh-Tuan Do et al. “An Effective Method for Detecting Personal Protective Equipment at Real Construction Sites Using the Improved YOLOv5s with SIoU Loss Function”. In: *2023 RIVF International Conference on Computing and Communication Technologies (RIVF)*. 2023, pp. 430–434. DOI: 10.1109/RIVF60135.2023.10471799.

[54] Manh-Hung Ha et al. “Emotional Inference from Speech Signals Informed by Multiple Stream DNNs Based Non-Local Attention Mechanism”. In: *EAI Endorsed Transactions on Industrial Networks and Intelligent Systems* 11.4 (2024).

[55] Thai Dinh Kim et al. “Surgical Tool Detection and Pose Estimation using YOLOv8-pose Model: A Study on Clipper Tool”. In: *2024 9th International Conference on Integrated Circuits, Design, and Verification (ICDV)*. 2024, pp. 225–229. DOI: 10.1109/ICDV61346.2024.10617290.

[56] Hai-Binh Le et al. “Robust Surgical Tool Detection in Laparoscopic Surgery using YOLOv8 Model”. In: *2023 International Conference on System Science and Engineering (ICSSE)*. 2023, pp. 537–542. DOI: 10.1109/ICSSE58758.2023.10227217.