Signal Processing Systems
Mekelweg 4,
2628 CD Delft
The Netherlands
https://sps.ewi.tudelft.nl/

SPS-2024-00

M.Sc. Thesis

Towards Robust Object Detection in Unseen Catheterization Laboratories

Zipeng Wang B.Sc.

Abstract

Deep-learning-based object detectors, while offering exceptional performance, are data-dependent and can suffer from generalization issues. In this thesis, we investigated deep neural networks for detecting people and medical instruments in the vision-based workflow analysis system inside Catheterization Laboratories (Cath Labs). The central problem explored in this thesis is the fact that the performance of the detector can degrade drastically if it is trained and tested on data from different Cath Labs.

Our research aimed to investigate the underlying causes of this specific performance degradation and find solutions to mitigate this issue. We employed the YOLOv8 object detector and created datasets from clinical procedures recorded at Reinier de Graaf Hospital (RdGG) and Philips Best Campus, supplemented with publicly accessible images. An aggregated version of object detection metrics was created for multi-camera system evaluation. Through a series of experiments complemented by data visualization, we discovered that the performance degradation primarily stems from data distribution shifts in the feature space. Notably, the object detector trained on non-sensitive online images can generalize to unseen Cath Labs, outperforming the model trained on a procedure recording from a different Cath Lab. The detector trained on the online images achieved an mAP@0.5 of 0.517 on the RdGG dataset. Furthermore, by switching to the most suitable camera for each object, the multi-camera system can further improve detection performance significantly. An aggregated 1-camera mAP@0.5 of 0.679 is achieved for single-object classes on the RdGG dataset.

Towards Robust Object Detection in Unseen Catheterization Laboratories

Thesis

submitted in partial fulfillment of the
requirements for the degree of

Master of Science

in

Electrical Engineering

by

Zipeng Wang B.Sc.
born in Chenzhou, China

This work was performed in:

Signal Processing Systems Group
Department of Microelectronics
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology

Delft University of Technology

Copyright © 2024 Signal Processing Systems Group
All rights reserved.

Delft University of Technology
Department of
Microelectronics

The undersigned hereby certify that they have read and recommend to the Faculty of Electrical Engineering, Mathematics and Computer Science for acceptance a thesis entitled **"Towards Robust Object Detection in Unseen Catheterization Laboratories"** by **Zipeng Wang B.Sc.** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: 29-01-2024

Chairman: _____

dr.ir. Justin Dauwels

Advisor: _____

ir. Rick M. Butler

Committee Members: _____

prof.dr.ir John J. van den Dobbelsteen

_____

Abstract

Deep-learning-based object detectors, while offering exceptional performance, are data-dependent and can suffer from generalization issues. In this thesis, we investigated deep neural networks for detecting people and medical instruments in the vision-based workflow analysis system inside Catheterization Laboratories (Cath Labs). The central problem explored in this thesis is the fact that the performance of the detector can degrade drastically if it is trained and tested on data from different Cath Labs.

Our research aimed to investigate the underlying causes of this specific performance degradation and find solutions to mitigate this issue. We employed the YOLOv8 object detector and created datasets from clinical procedures recorded at Reinier de Graaf Hospital (RdGG) and Philips Best Campus, supplemented with publicly accessible images. An aggregated version of object detection metrics was created for multi-camera system evaluation. Through a series of experiments complemented by data visualization, we discovered that the performance degradation primarily stems from data distribution shifts in the feature space. Notably, the object detector trained on non-sensitive online images can generalize to unseen Cath Labs, outperforming the model trained on a procedure recording from a different Cath Lab. The detector trained on the online images achieved an mAP@0.5 of 0.517 on the RdGG dataset. Furthermore, by switching to the most suitable camera for each object, the multi-camera system can further improve detection performance significantly. An aggregated 1-camera mAP@0.5 of 0.679 is achieved for single-object classes on the RdGG dataset.

Acknowledgments

While I have seen and learned a lot during my master's program, this thesis is a milestone at which I can proudly point and say, 'I did this'. My time at TU Delft has taught me not only fascinating knowledge about signal processing but also how to receive and give support. It is just like this thesis project, which would not have been possible without support from people, and for which I am deeply grateful.

First and foremost, I would like to express my deep gratitude to my supervisor Dr. Justin Dauwels. The journey of this thesis and other projects under his guidance has taught me countless invaluable lessons. It can go down pages to list them, which is difficult to do in an acknowledgment. However, the two most invaluable lessons are approaching problems with an engineering mindset and effectively presenting results. My special thanks go to Prof. John J. van den Dobbelsteen for his role in my thesis committee. I am looking forward to his insights during the thesis defense. I would like to thank Cristian Meo for his valuable advice during the thesis. Finally, I would like to give big thanks to my daily supervisor Rick Butler. His precious advice and support have made this work go much faster and smoother. Additionally, Rick is a very smart and interesting person, and I really had a good time working with him at Reinier de Graaf Hospital.

As the thesis emphasizes the data aspect, I would like to thank Philips and Reinier de Graaf Hospital for providing the procedure videos used in this project. My sincere thanks go to the patients and the people who prepared the videos.

On a personal note, this thesis would never have been possible without people who have given me strength. I would like to thank my friends and colleagues, with whom I have shared precious happy and bitter moments, especially in the Camelot building and on the 18th floor of the EWI Building. I am deeply grateful to my parents for their unconditional love and support, which I gained a better understanding of during the project. When you need time to heal, try the album *Carrie & Lowell* by Sufjan Stevens. At last, I would like to give a big hug to myself, for I always decide to be strong and happy with each new day and new sun.

Zipeng Wang B.Sc.
Delft, The Netherlands
29-01-2024

Contents

List of Figures

List of Tables

1 Introduction

1.1 Problem Statement

Object detection is a vital component in various video analysis systems. It detects objects of interest and proposes bounding boxes indicating their locations in 2D images. For workflow analysis, those detection results will be further processed by downstream modules, e.g., action recognition or scene interpretation. Due to regulations regarding privacy concerns, the availability of procedure recordings, especially annotated ones, is highly limited. For related applications, this data availability issue often results in the object detection model being trained on data with limited variety and having poor generalization ability. When the object detector encounters data from a novel environment, its performance can drop significantly, damaging the performance of the whole data analysis pipeline.

This project aimed to investigate the reasons for the performance degradation in a specific scenario, deploying the YOLOv8 object detector in previously unseen Catheterization Labs, and find potential solutions to alleviate this problem. In summary, our goals are to:

1. Investigate the reasons for detection performance degradation when the object detector YOLOv8 is deployed in a previously unseen Cath Lab.
2. Find solutions to alleviate this performance drop due to limited generalization ability.
3. Further improve and evaluate object detection performance on multi-camera systems.

1.2 Background

1.2.1 Catheterization Laboratory

Catheterization Laboratory (Cath Lab) is a specialized procedural room in hospitals, equipped with medical imaging instruments to visualize heart chambers and vessels. Figure 1.1 shows a photo taken inside a Catheterization Laboratory. Cath Lab is essential for the diagnosis and treatment of cardiovascular diseases. For example, Diagnostic Cardiac Catheterization requires a cardiologist to insert a catheter through an artery and finally into the heart via the guidance of a medical imaging instrument to find blockages or narrowings [2]. Percutaneous Coronary Interventions place a tiny balloon to alleviate blockages in vessels through minimally invasive surgery [2]. In our study, we used procedure recordings from the Cath Labs at two organizations: Reinier de Graaf Hospital, located in Delft, the Netherlands, and Philips Best Campus, located

Figure 1.1: An image of Catheterization Laboratory, taken from [1].

in Eindhoven, the Netherlands. Inside the Cath Labs, advanced C-Arm X-ray machines are used to perform imaging on the hearts and vessels of patients during various procedures. Their special design features a maneuverable C-shaped arm connected to the X-ray detector and X-ray source, allowing imaging of the patient from almost any angle. These machines are capable of providing real-time and high-resolution imaging with improved radiation exposure control.

Various threats and risks for both medical personnel and patients are associated with Cath Labs. Chronic radiation exposure can pose health concerns for interventional physicians, despite protective measures like lead aprons [3]. A study suggests there is still room for optimization on radiation exposure to patients and medical staff [4]. Besides radiation, health conditions caused by orthopedic strain are related to working inside Cath Labs [3]. The risks associated with chronic radiation exposure, fatigue, and beyond motivate improvement in protective measures during procedures, procedure management, and policies. For those purposes, workflow analysis can provide valuable insights into efficiency optimization and reducing the effects of potential risks.

1.2.2 Workflow Analysis and Data Acquisition

Workflow refers to the sequence of processes across space and time performed to accomplish a task [5]. Through monitoring and analyzing the sequence of actions from professionals in procedures, workflow helps to improve operational efficiency and ensure the quality of medical services [6]. Insights obtained from workflow analysis can be applied to Quality Improvement (QI) and process redesign [5].

Medical diagnosis and treatment are highly complicated and detailed processes [7]. Therefore, efficient data acquisition is required for complex modeling. Traditional data acquisition methods rely on comprehensive and extensive documentation from medical professionals. For example, detailed standards have been formulated for reporting documents of cardiac catheterization procedures [8]. Despite efforts to refine the documentation methods, they are subject to human factors. Additionally, these methods require excessive effort from healthcare professionals.

Therefore, monitoring systems have been introduced to record and analyze workflow in a non-intrusive and automated manner. For example, real-time location systems are actively used, which mainly rely on signal feeds from wireless sensors such as radio frequency technologies for localization and accelerometer along with gyroscope sensors for activity recognition [9]. Computer vision, on the other hand, is more versatile, as video can provide richer information. Extensive research has been conducted regarding human action recognition in the healthcare field [10]. Pose estimation, applied to patients and healthcare staff in clinical environments, can help researchers better understand their behavior and further improve the guidelines for treatment or recovery procedures [11] [12]. Vision-based event detection systems are also capable of event detection such as alerting for falls of the elderly [13]. Additionally, computer vision has been a very helpful tool for providing location information [14].

Object detection, as a key component in many vision applications mentioned above, is of great importance in most vision-based workflow analysis systems. It is the basis of many other computer vision tasks, including object tracking, semantic segmentation, and image captioning [15].

1.2.3 Object Detection and Its Challenges

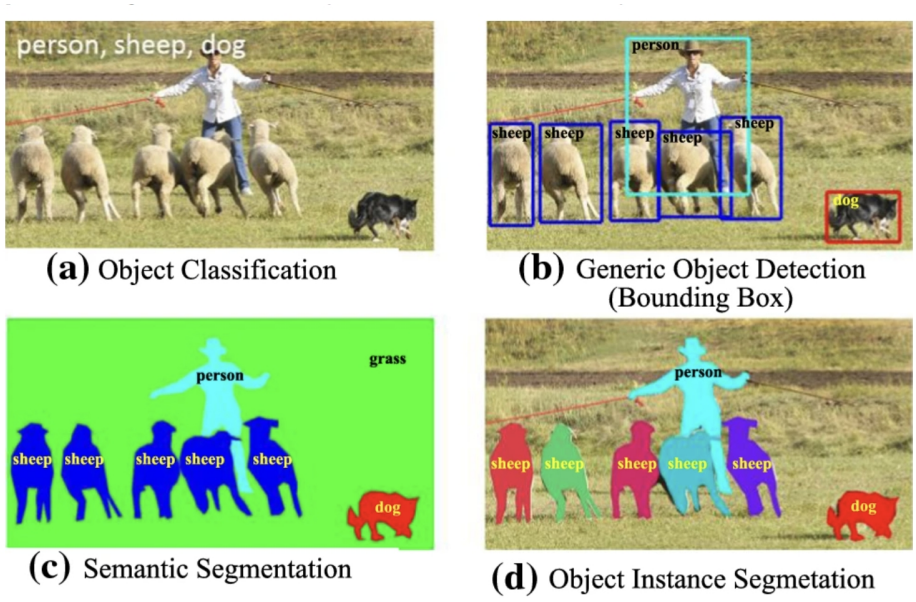


Figure 1.2: Generic object detection and related computer vision tasks, taken from [16].

Object detection is one of the most fundamental tasks in the field of computer vision and is crucial to image or video understanding, despite but also because of its highly challenging nature [15]. The most common form of object detection is generic object detection. The task is shown in Figure 1.2. Given an image, the model decides if objects from predefined categories are present in this image, and if so, the model should give a bounding box capturing the spatial location of each instance [16]. Before 2014, the field of object detection was dominated by handcrafted models [16]. The transition happened

when researchers noticed the superior performance of AlexNet, a deep learning-based image classification algorithm [17]. Since then, intensive research on deep-learning-based object detectors has been conducted. RCNN [18], Faster-RCNN [19], YOLO [20], and various other deep-learning object detectors have been developed.

Deep learning algorithms, featuring automatic feature extraction and utilization, benefit greatly from large quantities of high-variety data. Large-scale datasets, such as PASCAL VOC [21], ImageNet [22], and MS COCO [23], have been constructed and set as benchmarks to facilitate the research of object detection. Modern object detectors are often developed with those datasets as performance indicators and have shown strong performance in them. For example, Co-DINO-Deformable-DETR++ achieved 0.785 AP@0.5 in the COCO val dataset [24]. Those datasets have training and testing data sampled from the same (or very similar) data distribution. Additionally, individual images are taken from different environments, which largely increases the variety of the training and testing data. The COCO dataset itself contains 80 object categories, and 1.5 million object instances in various contexts [23].

For object detection in Cath Labs and other medical applications, the object detectors often need to operate in environments that differ from the setting where the training data was collected. Therefore, their performance can suffer consequently. The main issues are the lack of large datasets with high data variety plus the training and inference data following different distributions. Due to privacy concerns and related administrative issues, it is very hard to transfer medical data (procedure videos, in our case) outside the medical institutes. Therefore, it is usually impractical to have training and testing data covering various environments from different hospitals. Additionally, restricted data access makes it expensive to annotate new data and re-train the object detector for new deployments. Notably, public large-scale datasets usually adopt a crowd-sourcing approach for speed and cost-effectiveness, distributing data annotation tasks to a large number of participants on the Internet. However, this option is impractical in our case, as data annotation and model training usually require medical experts within the hospital. Without annotating new data and re-training the model in a new environment, the training data and inference data may follow very different distributions. This divergence can occur due to differences in room settings, instrument types, camera settings, and other medical-specific factors in different medical institutes. Numerous domain-related factors can affect the data distribution of the images and threaten the robustness of the object detector. Those factors highlight the importance of the data aspect when developing a robust object detector to deploy in Cath Labs.

1.3 Outline

This report is structured as follows:

1. In Chapter 2, we summarize the relevant literature. It includes deep-learning-based object detection models, relevant research on domain shifts, and applications of multi-camera systems. It lays the foundation for understanding performance degradation related to data distribution shift and using multi-camera systems for workflow analysis in Cath Labs.

2. In Chapter 3, we cover the methods we have applied in this thesis. Specifically, we describe the YOLOv8 object detector we have used and the visualization techniques that reveal the data distribution characteristics of our dataset in the feature space.

3. In Chapter 4, we illustrate our experiment results and our main findings. In particular, we expand on our datasets, evaluation metrics, experiment design, results, and analysis. Notably, we elaborate on the evaluation method for the multi-camera system. In the results section, we highlight the performance gap due to varying data distributions and show the capability of the multi-camera system to detect objects more reliably.

4. Finally, in Chapter 5, we provide discussions of the results, a summary of this project, and suggestions for future research directions based on our findings and limitations of current methods.

2 Related Works

2.1 Deep-learning-based Object Detector

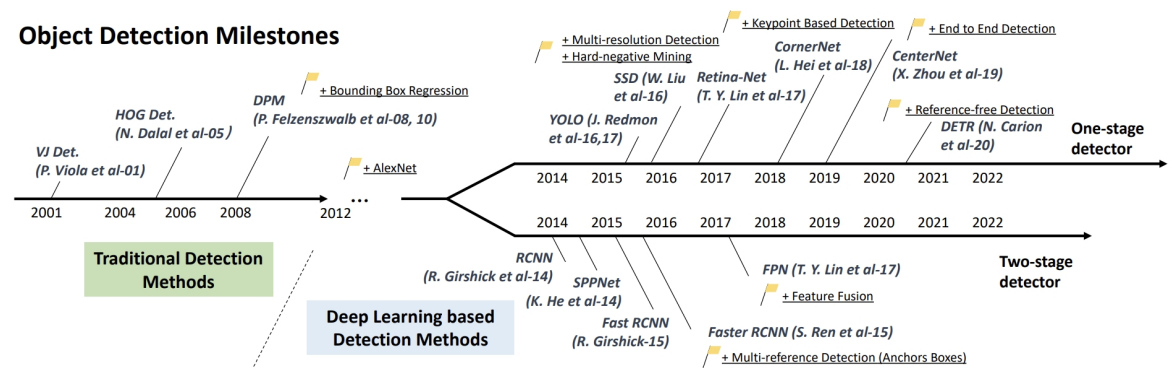


Figure 2.1: The developments of object detection algorithms, taken from [15].

Since the advent of AlexNet, research on object detection algorithms has been focused on deep-learning-based models and can be broadly divided into two directions, two-stage detector and one-stage detector [15]. As shown in Figure 2.1, each direction has numerous detectors featuring their unique designs. In this chapter, the first work of each direction is covered in detail, because they set the foundational design paradigms, which are crucial for comprehending the underlying mechanisms distinguishing the two directions. Their following works are introduced briefly with their major improvements. Specifically, we start with R-CNN, the first deep-learning-based object detector delivering groundbreaking performance, as well as other two-stage detectors. Then, we move to YOLO, whose design is more unified, and extend our discussion to subsequent developments in one-stage detectors.

2.1.1 Two-stage Object Detector

The advent of the two-stage object detector represents a milestone in the development of object detection algorithms. It is the first deep-learning object detector that showed ground-breaking performance compared to traditional algorithms. In 2012, AlexNet showed that Convolutional Neural Networks (CNNs) can effectively learn high-level features from images and utilize them to achieve record-breaking performance in image classification tasks [17]. However, the object detection task is more complex than image classification, as the number of object instances is not fixed and the detectors need to localize them apart from classification. Therefore, AlexNet can not be directly applied to object detection. Region-based Convolutional Neural Networks (R-CNN)

was introduced in 2014. Through a special two-stage design, R-CNN transfers the success of CNNs from image classification to object detection [18].

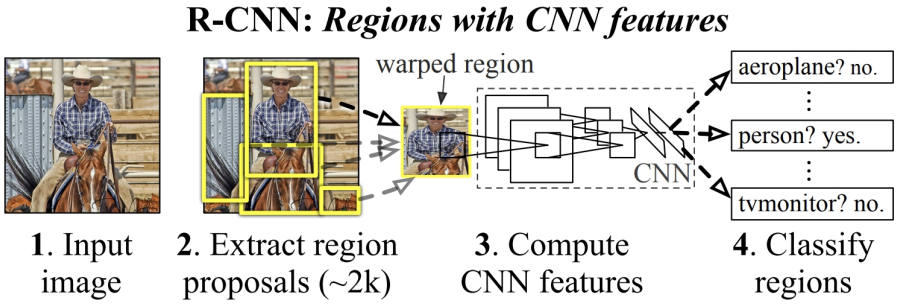


Figure 2.2: The data processing pipeline of R-CNN, taken from [18].

The design of R-CNN is shown in Figure 2.2. The pipeline of R-CNN can be roughly divided into two stages, the region proposal stage and the object detection stage [18]. In the first stage, it identifies potential regions of interest, which are agnostic to the class of objects. R-CNN chose the selective search method for this purpose, which employs a CNN to give 2000 region proposals [18]. After this, a simple preprocessing step will crop them individually into 227×227 RGB bounding boxes for the following feature extraction. Compared to the naive method of using a sliding window for region choosing, proposing 2000 regions can significantly reduce computational costs. The second stage performs the classification of that region. First, each cropped region will go through a CNN to obtain a 4096-dimension feature vector. The CNN is made up of 5 convolutional layers and 2 fully connected layers. It is pre-trained on ImageNet and fine-tuned on the proposed regions of the target dataset. Finally, a trained linear SVM predicts the class of the proposed regions based on their feature vector. An additional post-processing step, Non-Maximum Suppression (NMS) is applied to refine the result. It removes redundant bounding boxes by greedily removing extra overlapping bounding boxes.

As R-CNN established the design paradigm of the two-stage object detectors, the following works continued to make improvements based on R-CNN. SPPNet was introduced in 2015, which is a significant advancement from R-CNN [25]. Its Spatial Pyramid Pooling can process images of varying sizes without cropping or rescaling. Additionally, the whole image is processed only once to create feature maps, and the proposed feature vectors of regions can be generated from it. As a result, SPPNet is more than 20 times faster than R-CNN [25]. Fast R-CNN is both faster and more accurate than R-CNN [26]. It only processes the entire image once with the Region of Interest (RoI) pooling layer and can be trained in an end-to-end manner. Faster R-CNN speeds up the region proposal stage with Region Proposal Network (RPN), achieving real-time object detection [19]. Feature Pyramid Network (FPN) utilizes features from different levels of CNN to create a rich, multi-scale feature pyramid [27]. This design improves the object detection performance, especially for objects at different scales, and is thus used as a standard structure in many later works.

2.1.2 One-stage Object Detector

You Only Look Once (YOLO) is the first one-stage object detector [15]. Contrary to the two-stage detectors, which utilize deep-learning classifiers, YOLO treats object detection as a regression problem that predicts bounding box and corresponding class probabilities [20]. Due to the unified and simple architecture of YOLO, it enables end-end training, can run very fast, and has better generalization ability than two-stage detectors at its time [20].

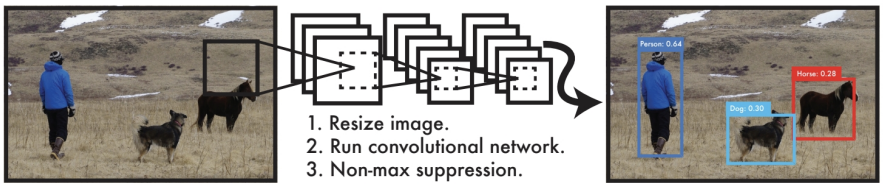


Figure 2.3: The data processing pipeline of YOLO, taken from [20].

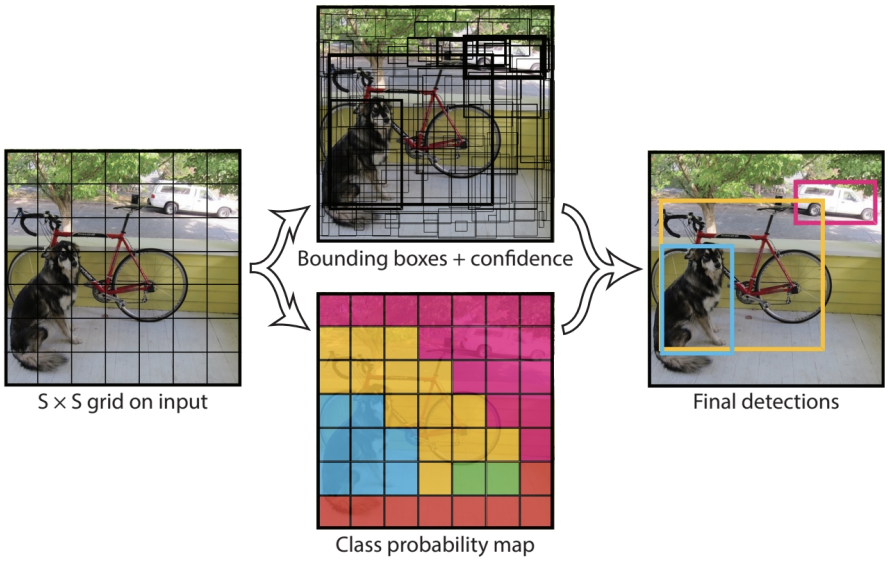


Figure 2.4: The design mechanism of YOLO, taken from [20].

The image processing pipeline and overall design of YOLO are shown in Figure 2.3 and Figure 2.4, respectively. When running YOLO, the image is first resized, then fed into the neural network, and finally run through a post-processing step. YOLO divides the image into an s×s grid, and each cell is responsible for predicting bounding boxes whose centers are inside the cell. For each bounding box, YOLO will predict 4 coordinates (x,y,w,h) and a confidence score associated with objectness and Intersection over Union (IoU). Objectness is the probability that an object exists in this cell, regardless of the class of the object. IoU is calculated as the ratio of the area of overlap between

the predicted and ground truth bounding boxes to the area encompassing both. This design helps to balance the detection and localization accuracy. A separate map for class probability is calculated. It predicts conditional class probability given there is an object existing $P(\text{class}|\text{object})$. The final output confidence score is the class probability times the bounding box confidence score, which is Confidence $= P(\text{class})\text{IoU}$. Note that $P(\text{class})$ and IoU are estimated values that the model tries to fit. As each cell will predict a fixed number of bounding boxes and their scores, there will be a large number of predicted bounding boxes. In the post-processing step, NMS will remove redundant bounding boxes of the same object, and we will also set a confidence threshold to filter out low-confidence detections.

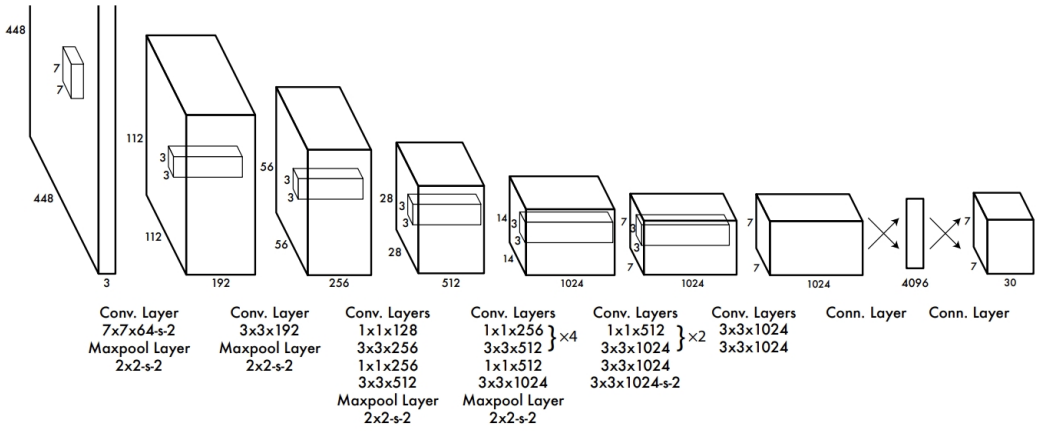


Figure 2.5: The architecture of the neural network in YOLO, taken from [20].

The architecture of the neural network is shown in Figure 2.5. It has 24 convolutional layers, as an adapted version of GoogLeNet, followed by 2 fully connected layers. This network design became a design paradigm in later works. The convolutional layers are referred to as the backbone, responsible for feature extraction, and the fully connected layers are called the head, responsible for performing regression from the extracted features [28]. Training of the networks involved pretraining the convolutional layers on the ImageNet image classification task, and then the whole model is trained for the detection task in the desired dataset.

Compared to two-stage detectors, the localization ability and detection performance of YOLO for small objects suffer. Those issues were addressed in the following works of the one-stage detector [15]. Some of the important following works are Single Shot MultiBox Detector (SSD), which improves detection accuracy while maintaining fast speed via integrating region proposal network (RPN) from faster R-CNN [29], and RetinaNet, which introduces focal loss to address the class imbalance of the foreground and background [30].

Starting from YOLO, a series of one-stage object detectors developed by different research groups, have gained popularity for object detection applications due to their fast speed and easy deployment. YOLO9000 (YOLOv2) uses a hierarchical method, Wordtree, achieving a very large detectable class category number of 9418 [31]. YOLOv3 improves the detection performance of small-scale objects with a network structure

similar to the Feature Pyramid Network [32]. Starting from YOLOv4, the YOLO families tend to use a combination of engineering techniques to further push the detection performance. YOLOv4 incorporates improved backbone CSPDarknet53, anchor boxes optimization, the Mish activation function, and data augmentation techniques such as Mosaic [33]. At the start of the project, YOLOv8 was the latest version of YOLO families [34]. It utilizes a large combination of techniques compared to its predecessor and has achieved a higher mAP on the COCO benchmark. Though the official paper is not available, we can find its design details in its documentation. Due to its fast speed and decent performance, we chose YOLOv8 as the detector and will introduce it in detail in Chapter 3.

2.2 Domain Shift and Its Solutions

2.2.1 Domain Shift

Domain shifts refer to the domain-related data distribution difference, which can damage the performance of machine learning methods [35]. Most machine learning methods naturally assume that the training data and testing data are independently sampled from the same distribution [36]. The object detectors mentioned in the previous sections are no exception. In reality, this ideal assumption usually does not hold. The training data (from the source domain) and testing data (from the target domain) can have a distribution shift.

Mathematically speaking, there are different kinds of domain shifts: covariate shift, label shift, and concept shift (note that there exist more general data shifts) [37]. Our case fits the covariate shift assumption. A machine learning model focuses on using input features X to predict target variables Y, which can be achieved by estimating the conditional probability $P(Y|X)$. Different types of domain shifts can be depicted by the change in decomposed components of joint distribution $P(X, Y) = P(X|Y)P(Y) = Pr(Y|X)P(X)$.

1. **Covariate shift assumes $P_{\textbf{train}}(X) \neq P_{\textbf{test}}(X)$, $P_{\textbf{train}}(Y|X) = P_{\textbf{test}}(Y|X)$:** Covariate shift happens when the relationship between the input and the output $(Y|X)$ is not changed, but the distribution of input $P(X)$ is changed. One example can be a model that predicts the happiness level of people from the weather. Assume the relationship between weather and happiness remains consistent. The model has been trained on data from the summertime in Australia, and now it is going to be deployed in the Netherlands in the wintertime. Our case fits the covariate shift assumption, because object appearances from different Cath Labs are only tiny subsets of the entire spectrum of possible images, while the concept of the object class (the relationship between the RGB image and its object class) is not changed. Even though $P(Y|X)$ is unchanged, covariate shift will make it difficult to estimate $P(Y|X)$ in regions where the data points are sparse or absent.

2. **Label shift assumes $P_{\textbf{train}}(Y) \neq P_{\textbf{test}}(Y)$, $P_{\textbf{train}}(X|Y) = P_{\textbf{test}}(X|Y)$:** It is the case that the prior of the label $P(Y)$ changes while its conditional probability given the input $P(X|Y)$ is not changed. An example can be disease diagnosis

in different periods when the prior of having the disease is different. Observed symptoms given having the disease stay the same. When a person shows the symptoms, the chance it having the disease is higher at the time when the disease prevails.

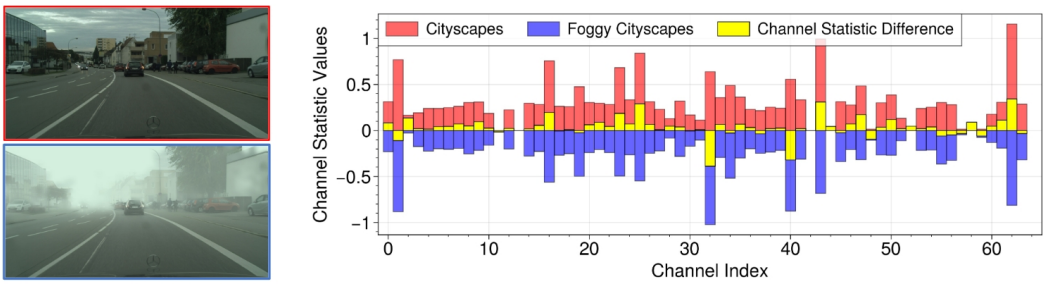
3. **Concept shift assumes $P_{\textbf{train}}(Y|X) \neq P_{\textbf{test}}(Y|X)$, $P_{\textbf{train}}(X) = P_{\textbf{test}}(X)$:** Concept shift assumes the internal relationship between the input and the output $P(Y|X)$ is changed. Therefore, concept shift is more difficult to tackle. One example can be having an aged and inaccurate thermometer, given the same input temperature its output readings changed due to the aging process.

Domain shift can happen due to numerous factors and pose serious threats to the deployed machine learning systems. Researchers have found its presence in images taken by different types of cameras [38]. Deploying an image segmentation system in a new city for autonomous driving can seriously hurt its performance [39]. In the medical field, different imaging devices can also impair performance for automatic polyp detection in the digestive system [40]. The list of factors that cause domain shift can be nearly unlimited. Most commonly, they are changes in lighting, camera angles, or backgrounds [41]. The ubiquitous nature of domain shift makes it hard to avoid. For less critical systems such as imaging searching systems, the errors introduced by domain shift are undesirable but can be tolerated. However, for systems deployed in high-stakes fields, such as autonomous driving and healthcare, the consequences of such errors can be lethal, thus research on domain shift has attracted growing interest [42].

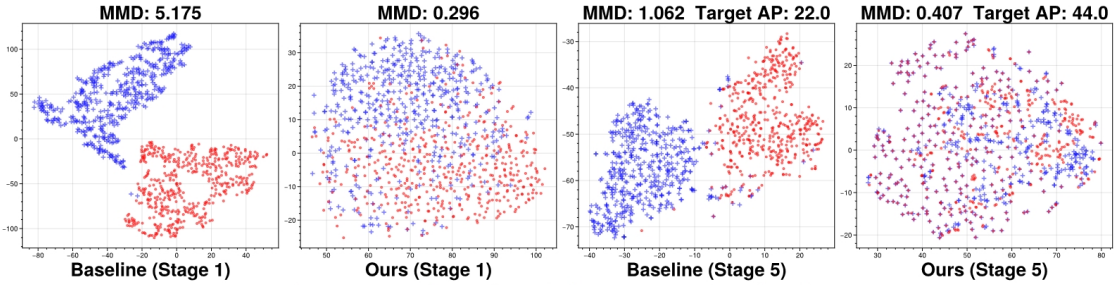
2.2.2 Domain Adaptation and Domain Generalization

Making the object detector work in a novel domain is a very challenging yet rewarding task, which attracts researchers working in various directions. That effort can generally be categorized into two directions: domain adaptation and domain generalization [43]. Domain adaptation adjusts the machine learning model based on the data distribution of the target domain to increase its performance when the model is deployed in the target environment. It requires access to the target dataset and adjusts models to perform well in the target domain without considering other environments. Though it is a valuable direction, we did not pursue it in this project because domain generalization is more practical in our specific medical setting. As a promising alternative, domain generalization intends to make the model maintain good performance in various unseen domains. Research in this field includes representation learning, training strategy, and data augmentation [43]. In this approach, the model is encouraged to extract and utilize features that are robust or even invariant to domain changes so that the model can have consistent performance in various domains.

The following is a very relevant example in the field of autonomous driving, where cars need to operate in various weather conditions, which can alter the data distribution of the appearance of objects of interest. Researchers have found that images from car cameras in a foggy environment have a different data distribution than in clear weather, which will hurt the performance of the detector in foggy weather as shown in Figure 2.6 [44]. The researchers showed the domain shift by calculating the mean



(a) Feature channel statistics (mean) difference at **Stage 1**



(b) Feature channel statistics (mean) visualization

Figure 2.6: The feature channel statistics (mean) difference in the CityScapes and Foggy CityScapes datasets. The distribution difference is alleviated by introducing perturbation in the backbone of the model. The image is taken from [44].

of each feature channel. They found that, by injecting their proposed noise named "Normalization Perturbation" in the backbone of the model, the data distribution of two domains can be closer, which alleviates the performance degradation when training in clear weather and testing in foggy weather.

2.3 Applications of Multi-camera Systems

The multi-camera systems are applied to overcome the limitations inherent in a single-camera setup. Primarily, it can provide wider coverage, better reliability, and improved depth and 3D spatial perception. Its potential can go even further because its multi-perspective views can provide more comprehensive information about the scene for sophisticated applications. Therefore, multi-camera systems are widely used in robotics, traffic monitoring, and indoor localization systems. Moreover, such systems have been drawing increasing research interest in medical applications.

Expanding the visual coverage is one major reason for deploying a multi-camera system. Monitoring systems need to track an object or multiple objects for an extended range, which is hard to accomplish with a single camera. For traffic monitoring systems, the research interests have been focused on the re-identification problem, which is linking objects from different camera views. Current state-of-the-art algorithms for traffic tracking can link cars despite drastic viewpoint changes and the absence of overlapping views [45]. This capability allows traffic analyzing systems to track traffic flow in great

detail, covering each vehicle across multiple blocks. For indoor applications, the cameras are more clustered and deployed to provide better coverage of the rooms. Such systems can provide the location of the object inside a room even at every corner [46]. Research on camera layout has been conducted to optimize camera coverage for specific tasks while balancing the number of cameras used [47]. In healthcare, expanded coverage is an important reason for multi-camera systems. Fall detection systems can analyze real-time video feeds and automatically alert for sudden fall accidents of patients and elderlies [48] [49]. To run this system reliably, multiple cameras are needed to eliminate blind spots. Otherwise, there exists the possibility that a patient fell outside the covered area by the camera, and the healthcare staff stayed unalerted for the accident.

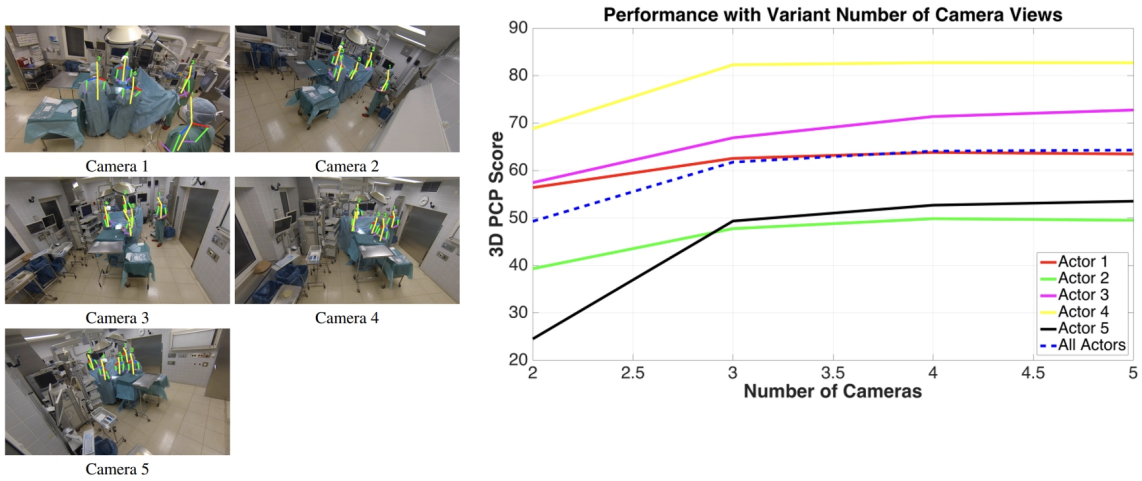


Figure 2.7: 3D pose estimation results inside an operating room from the 3DPS model, which merging 2D pose estimation results. The performance of the model against camera coverage is shown in the line plot. The image is taken from [50].

The capability of providing 3D spatial information is another advantage of multi-camera systems. The applications of multi-camera systems have a long history in robotics. Starting from fixed binocular cameras for depth estimation, the capability now has been extended to using uncalibrated multiple cameras to guide robotic arms to operate in 3D space [51]. For medical applications, 3D information from multi-camera systems is also highly valuable. One popular application is 3D human pose estimation. For example, researchers can track the 3D pose of surgeons (played by actors in the example) inside an operating room [50] as shown in Figure 2.7. It is done by initially tracking 2D poses in each view, and then merging them via the 3DPS model. The model can generate 3D poses when the person is tracked by at least 2 cameras, and the performance increases with more cameras. The pose information acquired by this system can be used in vision-based workflow analysis systems for analyzing the behaviors of medical staff [52]. It can be very helpful for increasing efficiency and warning for fatigue. The detector in this project can advance beyond merely human activities, as it can provide information about the interaction between humans and

instruments, providing more insights such as X-ray exposure. 2D to 3D information conversion of medical instruments can be accomplished by triangulation-based methods or more sophisticated models.

The redundancy provided by multi-camera systems can ensure reliability, which is crucial for medical applications. As shown in Figure 2.7, the camera setup for the 3D pose estimator uses 5 cameras facing the operating tables, providing overlapping views. As the operating room is a highly crowded environment, the extra cameras can work when a person is occluded from certain viewpoints. Additionally, the viewpoint variety can also ensure reliability. As object detection is a highly challenging task, object detectors can have difficulties in dealing with viewpoint changes. The appearance of objects can be very different from different angles. Researchers have attempted to compensate for this problem with a two-camera setup, and they demonstrated a higher detection performance can be obtained by combining detection results from two cameras [53]. This idea can also be helpful for our project, as when the training data has limited viewpoint variety, our trained detectors can also be unstable with regard to viewpoint changes.

3 Methods

3.1 YOLOv8 Object Detector

YOLOv8 was the latest object detector in the YOLO family at the start of this project. It was developed by Ultralytics, who also developed YOLOv5. Its software provides a well-designed library for easy implementation of both training and inference. The algorithm achieved state-of-the-art object detection performance, with its x version (extra large) achieving 53.9 mAP@0.5-0.95 on the 2017 COCO val dataset [34].

YOLOv8 follows the same design paradigm as the YOLO families. It incorporates an anchor-free design and provides better generalization ability than two-stage detectors due to its simple and unified model design. Additionally, the built-in data augmentation module in the training part can also help to improve its generalization ability when deployed in the real world.

3.1.1 Architecture

The model architecture of YOLOv8 is shown in Figure 3.1. Like previous YOLO models, it has a backbone for feature extraction, a neck for feature multi-scale and multi-level feature integration, and a head for the final bounding boxes prediction.

The backbone of YOLOv8 is a modified version of CSPDarkNet, which is made up of a series of convolutional layers and the newly introduced Cross Stage Partial Bottleneck with two convolutions fast (C2f) layers [54]. The convolutional layers perform standard 2D convolutions to extract features. The C2f module is inspired by the ELAN module [55]. Its design features bottlenecks with skip-connections. The C2f module can make the model more trainable, learn multi-scale features more effectively and expand its receptive field [56]. At the end of the backbone is the Spatial Pyramid Pooling-Fast (SPPF), modified from Spatial Pyramid Pooling (SPPF) for greater inference speed [25] [54]. It serves the same role as SPP, which is dandling varied input sizes and preserving better spatial information for improved accuracy.

The neck is designed to efficiently merge feature maps from 3 different scales in the backbone. It helps to improve the localization accuracy and detection performance of smaller objects. The design of the neck in YOLOv8 follows the Path Aggregation networks (PANet) [57], with its newly introduced C2f module used in it.

The head will finally predict the bounding boxes from the feature maps from the neck. The design of YOLOv8 features an anchor-free and decoupled head. Anchor-free is first introduced by CornerNet [58]. Rather than predicting the bounding box coordinates directly, the anchor-based model calculates their offsets to a set of anchors predefined at the training stage. This design makes the networks easier to train but will impose a strong prior of the bounding box. By adopting an anchor-free design, YOLOv8 gained better generalization ability. The decoupled detection head design

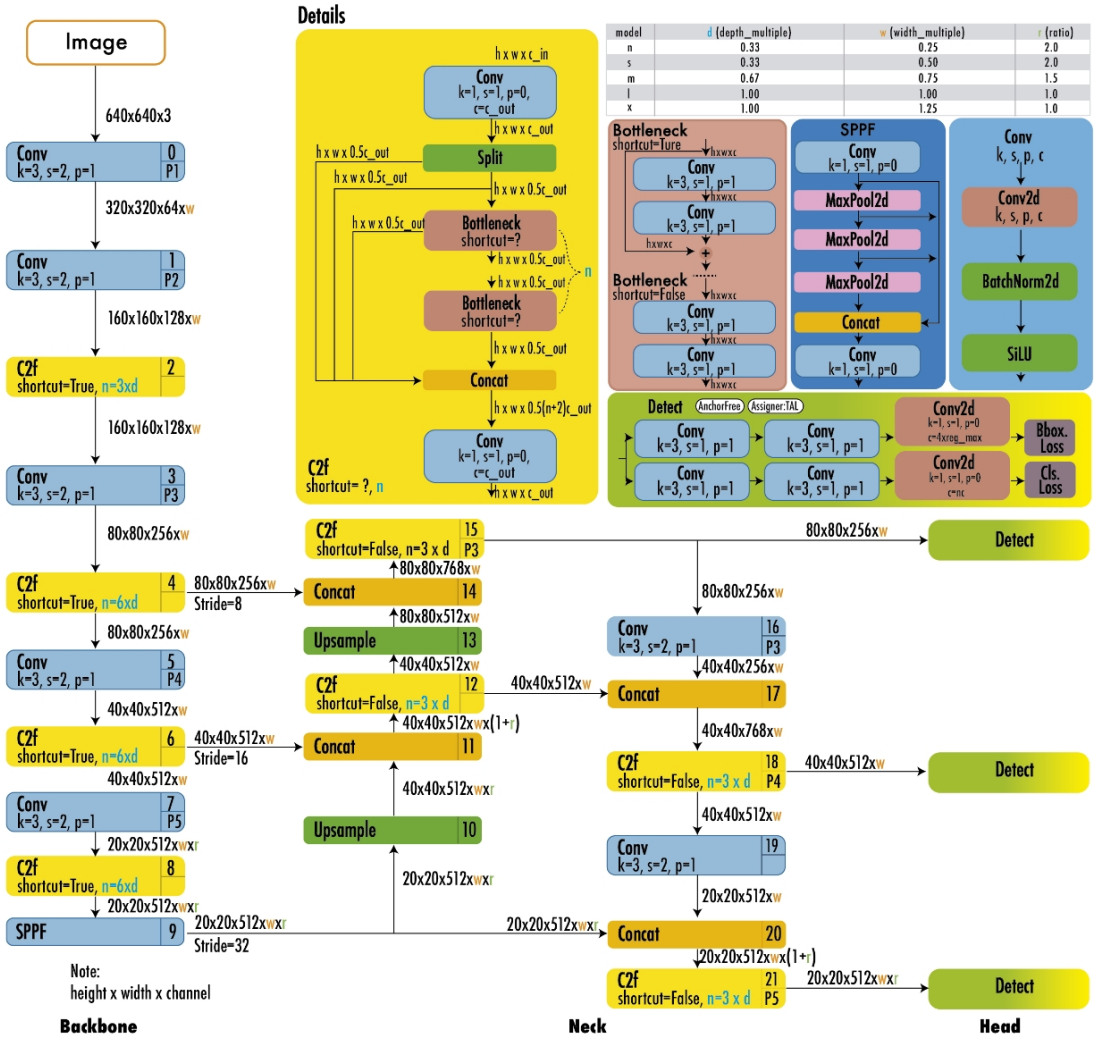


Figure 3.1: The architecture of YOLOv8, taken from [34].

is first proposed in the Decoupled Classification Refinement (DCR) network, which improves the precision of object detection greatly [59]. In the original YOLO model, the task of predicting object class and bounding box coordinates is accomplished by a single unit. The researchers of DCR found the tasks of classification and regression are fundamentally different. The loss functions derived from those two tasks can lead to gradient conflicts, resulting in performance drops [59]. The YOLOv8 instead has two branches in its head, the classification branch for class probabilities prediction and the regression branch for bounding box coordinates prediction.

3.1.2 Loss Function

The loss function of YOLOv8 is a weighted sum of multiple loss functions responsible for predicting the class probability and bounding box coordinates. Though the exact formula is not given by its authors, information can be found in its GitHub repository.

The losses are Binary Cross-Entropy (BCE) loss in the classification branch, Complete Intersection over Union (CIoU) loss, and Distribution Focal Loss (DFL) loss in the regression branch.

BCE loss is the most commonly used loss for the classification task:

$$\text{BCE Loss} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \cdot \log(\hat{y}_i) + (1-y_i)\cdot \log(1-\hat{y}_i)\right], \tag{3.1}$$

where y_i is the actual label $\{0, 1\}$; $\hat{y}_i$ is the predicted probability.

CIoU Loss is an advanced version of the IoU loss used for bounding box regression in object detection. It accounts for the overlap, central point distance, and aspect ratio between the predicted and ground truth boxes [60]. Its formula is:

$$\text{CIoU Loss} = 1 - \text{IoU} + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha \cdot v, \tag{3.2}$$

$$v = \frac{4}{\pi^2}\left(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w^b}{h^b}\right)^2, \tag{3.3}$$

$$\alpha = \begin{cases} 0 & \text{if IoU} < 0.5 \\ \frac{v}{(1-\text{IoU})+v} & \text{if IoU} \geq 0.5 \end{cases}, \tag{3.4}$$

where $\rho(b, b^{gt})$ is the Euclidean distance between the center points of the predicted bounding box and the ground truth bounding box; c is the diagonal length of the smallest enclosing box covering both the predicted bounding box and the ground truth; v measures the consistency of the aspect ratio; α is a trade-off parameter.

DFL helps to provide more accurate localization by considering the distribution of bounding box predictions for a better localization quality representation [61]. For this loss, bounding box coordinates are given in the form of a probability distribution. Its expression is:

$$\text{DFL}(S_i, S_{i+1}) = -(y_{i+1} - y)\log(S_i) + (y - y_i)\log(S_{i+1}), \tag{3.5}$$

where y_i and y_{i+1} are boundaries of the discretized bins that are closest to the target coordinate y; (S_i) and S_{i+1} are the predicted probabilities for these bins.

3.1.3 Training

The YOLOv8 library offers convenient built-in functions for controlling the training process. We can adjust transfer learning, data augmentation, and training strategies by changing them. Careful utilization of those functions can help to improve the model performance and accelerate the training process.

In YOLOv8, transfer learning can be achieved by initializing the model with pre-trained weight before training. Those weights are usually obtained by training on

large-scale datasets. YOLOv8 provides pretrained weights from the MS COCO dataset, which is a very large and diverse dataset. Transfer learning can significantly accelerate the training process, lowering the requirement for the quantity of training data, and providing benefits for the model performance. Its benefits come from feature reuse. In the pretraining process, the model can learn useful features from the pretraining dataset. And because pretraining datasets are large and diverse, the learned features are usually robust. As the images in our datasets and in the COCO datasets are all captured by RGB cameras, we can expect they share common visual features. Therefore, the object detector can reuse those features to improve its performance in detecting the objects inside Cath Labs.

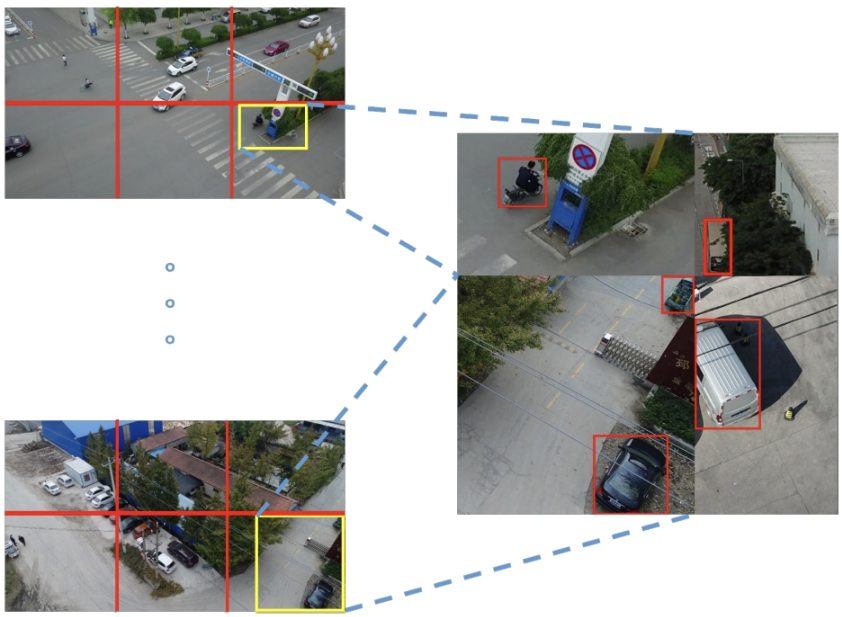


Figure 3.2: The Mosaic data augmentation technique, taken from [62].

To improve its generalization ability and robustness in real-world deployment, YOLOv8 provides built-in data augmentation functions. The data augmentation techniques include Mosiac, augment HSV, random affine, and other random linear transformations. Mosaic augmentation stitches cropped part of four random training images as shown in Figure 3.2 [62]. By doing so, the model can be more robust to different object scales and background information. It also moderately helps with occlusion issues, since parts of objects might be cropped out. Augment HSV (hue, saturation, value) randomly adjusts the HSV values of the images to create variations in color and brightness. This augmentation can make the model more robust to changing lighting conditions or camera color shifts in real life. Random affine transformation including scaling, translation, and rotation. It helps the model to detect objects regardless of their size and orientation.

YOLOv8 also offers a wide range of parameters to control training strategies. They

help to accelerate the training process and improve the detection performance. The basic training setting of YOLOv8 includes changing the optimizer, batch size, learning rate, weight decay, etc. More advanced training strategies include the warm-up phase and Exponential Moving Average (EMA). In the warm-up phase, the learning rate gradually increases from a lower value to the target learning rate. It helps to mitigate instability in the early training phase. EMA maintains a moving average of model weights during the training process. It smooths the noise in the training data, which helps to stabilize the training process and improve the model performance [63].

3.2 Visualization of Data Distributions

As relevant literature on domain shift suggests the relationship between data distribution shifts and the performance degradation of machine learning models, we want to investigate whether images in our datasets follow different distributions in the feature space. In this project, we have applied visualization methods to show the feature distribution of the images in our datasets.

The impact of data distribution shifts on the performance of the object detector can be understood from an optimization perspective. At the training stage, the weight of the model is adjusted to optimize the empirical loss in training:

$$\hat{J}_A(\theta) = \frac{1}{N}\sum_{i=1}^{N} L(x_i, y_i; \theta), (x_i, y_i) \sim A, \tag{3.6}$$

where $\hat{J}_A(\theta)$ is the empirical loss in training; θ is the weight of the model; (x_i, y_i) are the data points sampled from distribution A.

Due to different testing data, the empirical loss in the testing process is:

$$\hat{J}_B(\theta) = \frac{1}{N}\sum_{i=1}^{N} L(x_i, y_i; \theta), (x_i, y_i) \sim B. \tag{3.7}$$

If the data in the training and testing process fits the independent and identically distributed (i.i.d.) assumption, the model can deliver good performance as the distribution A and B is the same. But when the domain shifts happen, the loss function can be very different due to the difference in the data. Weight optimized in minimizing $\hat{J}_A(\theta)$ can not guarantee to minimize $\hat{J}_B(\theta)$, resulting in a bad performance in the test set.

Despite the seemingly complex architectural design of YOLOv8, the purpose of the network design is first feature extraction and then regression as shown in Figure 3.3. We aimed to show the distribution shift in the regression part. For this purpose, We chose the feature maps just before the detection head. Though there are three multi-scale feature maps for bounding box prediction, we chose the deepest one for its richer semantic information and also the simplicity of our experiments. The high dimensionality of feature maps (20×12×512) makes it hard to show the data distribution. Therefore,

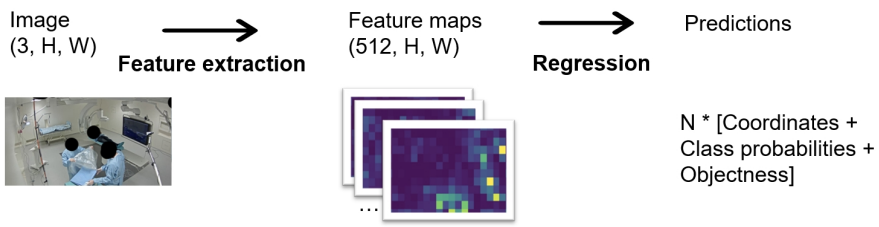


Figure 3.3: The simplified data processing pipeline of the neural network in YOLOv8.

we reduce its dimensionality in two steps: calculation of feature channel statistics and dimensionality reduction. Similar techniques can be found in works regarding domain shift [44].

3.2.1 Feature Channel Statistics

The feature channel statistics capture the activation patterns of the neural networks. In this project, we chose mean and variance as the feature channel statistics, while the work on domain shift in autonomous driving visualizes mean as previously shown in Figure 2.6. The mean provides a measure of the average intensity of the activations, and variance measures how much the activations vary across the feature map. The significance of the mean and variance of the channel can be seen in many important works in deep learning, such as Batch Normalization calculates them for stabilizing the training process of neural networks [64]. For model compression, the mean and variance of the activations can also be used as the criteria to prune unimportant neurons [65].

As there are 512 layers in our feature maps and we calculate the mean and variance of each layer, we have a 512-dimension mean vector and a 512-dimension variance vector. They are concatenated to get the final feature vector of 1024 dimensions. This feature vector captures the neural activation pattern of the whole image.

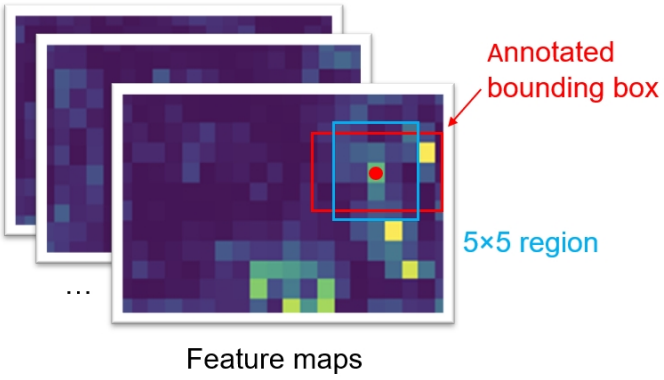


Figure 3.4: The 5×5 region for calculating object-level feature vector for visualization.

Additionally, we also calculate the feature vector on the object level. YOLOv8 is designed in a way that each cell is responsible for predicting the bounding boxes centered

at it. We can read the position of an object from the annotation, and discretize it to get the responsible cell in the feature maps. Additionally, in the head of YOLOv8, two convolutional layers have a kernel size of 3 as shown in Figure 3.1. The receptive field is 5×5. Therefore, we can find the 5×5 region and calculate its feature channel statistics instead of the whole image as shown previously in Figure 3.4. The feature vector will have the same dimension because we calculate it with the same method only in a smaller region instead of the whole image. This method allows us to gain a better understanding of how objects influence the data distribution.

3.2.2 Dimensionality Reduction Methods

The dimensionality reduction step aims to further reduce the channel feature statistics vector to 2 dimensions for visualization. In this project, we tested three techniques: Principal Component Analysis (PCA), T-distributed Stochastic Neighbor Embedding (T-SNE), and Uniform Manifold Approximation and Projection (UMAP).

In summary, PCA can model global data structure well but is unable to capture non-linearities. By contrast, both T-SNE and UMAP model non-linearity. While T-SNE is good at modeling local data structure, it falls short at global data structure. On the other hand, UMAP can balance both local and global data structures [66].

3.2.2.1 PCA

PCA is one of the most commonly used dimensionality reduction tools. It linearly projects vectors into lower dimensional space while maximizing the variance after projection.

PCA performs linear projection with a matrix multiplication:

$$\mathbf{Y} = \mathbf{U'^T X}, \tag{3.8}$$

where $\mathbf{Y}$ is the projected low dimensional data; $\mathbf{U'^T}$ is the projection matrix; $\mathbf{X}$ is the original data.

$\mathbf{U'^T}$ is obtained by computing the eigenvectors and eigenvalues of the covariance matrix to identify the principal components. This is done by solving the equation:

$$\text{Cov}(\mathbf{X}, \mathbf{X})\mathbf{U} = \mathbf{U}\mathbf{\Lambda}, \tag{3.9}$$

where $\text{Cov}(\mathbf{X}, \mathbf{X})$ is the covariance matrix of $\mathbf{X}$; $\mathbf{U}$ is the matrix of eigenvectors; $\mathbf{\Lambda}$ is the diagonal matrix of eigenvalues.

Projection matrix $\mathbf{U'}$ is obtained by taking top k eigenvectors (columns) of $\mathbf{U}$. In our case, we chose the top 2 columns for visualization.

3.2.2.2 T-SNE

T-SNE is a widely used statistical method in the field of machine learning for visualizing data from high dimensional space [67]. It works by finding low-dimensional (typically two or three-dimensional) representations of high-dimensional data points while keeping

the data structure. However, studies have found that T-SNE pays more attention to local data structure while ignoring global data structure [68].

The calculation process of T-SNE is done by minimizing the difference between the distributions in the original high-dimensional space and the projected low-dimensional space. It starts by computing asymmetric distributions. Given a data point, the probability of picking another data point as a neighbor is under a Gaussian distribution center around it. The probability that x_i will pick x_j as the neighbour is:

$$p_{j|i} = \frac{\exp(-||x_i - x_j||^2/2\sigma_i^2)}{\sum_{k \neq i} \exp(-||x_i - x_k||^2/2\sigma_i^2)}. \tag{3.10}$$

The symmetric pair-wise similarity is defined as:

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2N}, \tag{3.11}$$

where N is the number of data points

In the low dimensional space, the corresponding data points of x_i and x_j are y_i and y_j. A heavy-tailed student's t-distribution is used for more consistent modeling of the distance [67]. The pair-wise similarity in low dimensional space is calculated as:

$$q_{ij} = \frac{(1 + ||y_i - y_j||^2)^{-1}}{\sum_{k \neq l}(1 + ||y_k - y_l||^2)^{-1}}. \tag{3.12}$$

The difference between the two distributions is modeled by Kullback–Leibler divergence:

$$KL(P||Q) = \sum_i \sum_j p_{ij} \log\left(\frac{p_{ij}}{q_{ij}}\right). \tag{3.13}$$

Finding the low-dimensional points is done by minimizing Kullback–Leibler divergence. As it is non-convex, the gradient descent method is used here.

3.2.2.3 UMAP

UMAP is a dimensionality reduction technique that serves a similar purpose as T-SNE [66]. It is widely applied in the field of data science like bioinformatics, because it preserves both the global and local structure of the high dimensional data [66].

UMAP is a manifold learning method. The idea behind it is that the high-dimensional data can be modeled as a manifold embedded in high-dimensional space. UMAP tries to learn the manifold structure and project it into a lower-dimensional space. The modeling of manifold structure is based on the theory of Riemannian geometry and topological data analysis.

The implementation of UMAP is shown as the following. The first step is constructing a weighted k-neighbour graph to describe the relations of points in high-dimensional space. The weight is calculated by:

$$w((x_i, x_{i_j})) = \exp\left(-\frac{\max(0, d(x_i, x_{i_j}) - \rho_i)}{\sigma_i}\right), \tag{3.14}$$

$$\rho_i = \min\{d(x_i, x_{ij}) \mid 1 \leq j \leq k, d(x_i, x_{ij}) > 0\}, \tag{3.15}$$

$$\sum_{j=1}^{k} \exp\left(-\frac{\max(0, d(x_i, x_{i_j}) - \rho_i)}{\sigma_i}\right) = \log_2(k), \tag{3.16}$$

where $d(x_i, x_{i_j})$ is the Euclidean distance between the two points; k is the number chosen for k-nearest neighbors. This equation is solved for calculating ρ_i.

The symmetric weight $w((x_i, x_j))$ is further acquired by symmetrization:

$$\mathbf{B} = \mathbf{A} + \mathbf{A}^{\mathbf{T}} - \mathbf{A} \circ \mathbf{A}^{\mathbf{T}}, \tag{3.17}$$

where $\mathbf{A}$ is the weighted adjacency matrix calculated by $w((x_i, x_{i_j}))$;$\circ$ is the point-wise product. $\mathbf{B}$ is the matrix of symmetric weights.

The second step is to formulate the graph layout of points in low-dimensional space. It is done via a force directed graph layout algorithm, which applies attractive forces to vertices and repulsive forces to edges. The attractive force and repulsive force are calculated as:

$$F_{\text{Attractive}} = -\frac{2ab\|\mathbf{y}_i - \mathbf{y}_j\|_2^{2(b-1)}}{1 + \|\mathbf{y}_i - \mathbf{y}_j\|_2^2} w((x_i, x_j))(\mathbf{y}_i - \mathbf{y}_j), \tag{3.18}$$

$$F_{\text{Repulsive}} = \frac{2b}{(\epsilon + \|\mathbf{y}_i - \mathbf{y}_j\|_2^2)\left(1 + a\|\mathbf{y}_i - \mathbf{y}_j\|_2^{2b}\right)}(1 - w((x_i, x_j)))(\mathbf{y}_i - \mathbf{y}_j), \tag{3.19}$$

where a and b are parameters; ϵ is a very small constant to prevent division by zero.

4 Experiments and Results

4.1 Datasets

The datasets involved in this project are from one real clinical procedure performed in a Cath Lab inside Reinier de Graaf Hospital, two mock procedures performed inside a Cath Lab in Philips Best Campus, and online images that are collected from Google Images and Bing Images.

Those datasets have their unique characteristics. The RdGG and Philips Best datasets have highly limited backgrounds and object appearance variety, since the images of each dataset are captured inside a single Cath Lab. However, the videos from multiple cameras can cover the objects from different angles, and their appearance changes over time. Additionally, the cameras in the Cath Labs have a wider field of view for better coverage, which can introduce a greater degree of distortion than hand-held cameras, especially at the edge of the image.

On the other hand, the collected online images have a wide variety of objects and background appearances as each photo is taken inside different Cath Labs. However, the viewpoint variety of these images is limited, because most photos were taken by hand-held cameras from a limited range of angles. Additionally, most images were captured without operations being performed. Those still images do not contain appearance changes of objects across multiple frames.

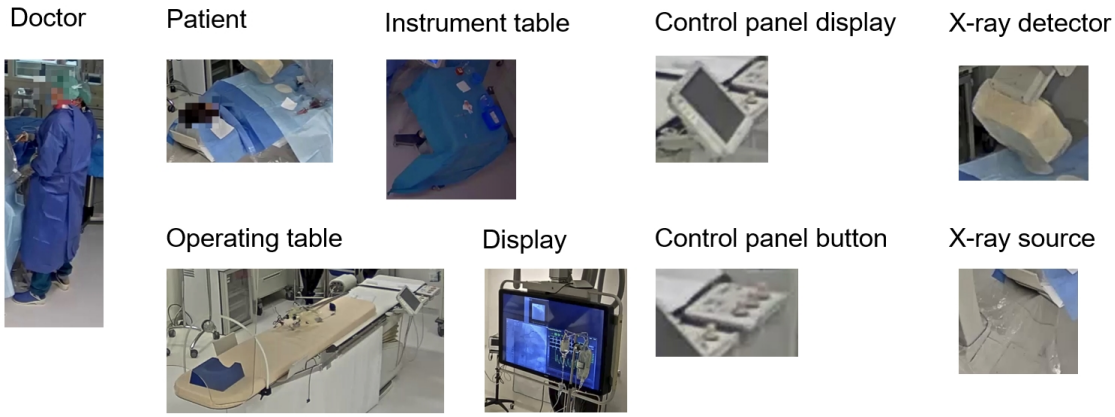


Figure 4.1: The object Class for detection in the RdGG 20211007 dataset.

The object classes for detection in this project are: [doctor, patient, operating table, instrument table, control panel display, control panel button, x-ray detector, x-ray source, display]. Figure 4.1 shows the object classes for detection inside the RdGG 20211007 dataset. The RdGG dataset and online image dataset also have class 'lead

shield', but it is not reported in this project because the Cath Lab in Philips Best does not have a lead shield. All the object instances were annotated manually. The annotation process was carried out using the Computer Vision Annotation Tool (CVAT), an open-source, web-based tool designed for the annotation of images and videos [69]. The work was done by the MSc student of the project, who visually recognized objects of interest and annotated them inside bounding boxes. The annotation results are plotted and re-examined by the student to avoid errors. The annotation process adhered to the guidelines that when severe occlusion happens (over 70% of an object is occluded), we do not annotate the object unless it is a person with more than 50% of the face visible.

4.1.1 RdGG Dataset

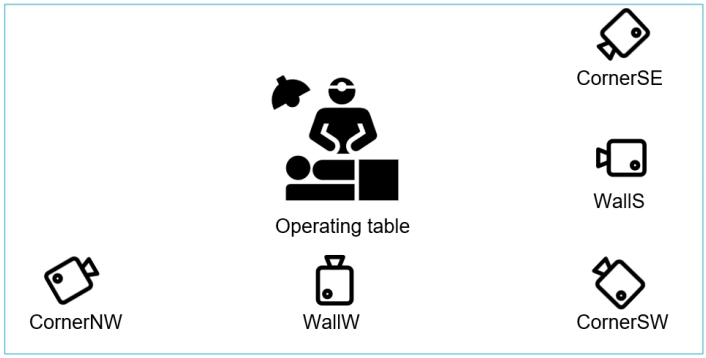


Figure 4.2: The multi-camera system setup in the Cath Lab of Reinier de Graaf Hospital.

Figure 4.3: Example images from the multi-camera system in the Cath Lab of Reinier de Graaf Hospital.

The RdGG 20211007 dataset was captured from five fixed cameras mounted inside the Cath Lab in Reinier de Graaf Hospital. All five cameras are facing the operating table to provide an overlapping view of the operation as the camera setup is shown in Figure 4.2. All five cameras are synchronized and have a frame rate of 25 Frames Per Second (FPS). The images in the dataset were taken every 5 seconds from the

recording of procedure 20211007. The example image from each camera is shown in Figure 4.3. For privacy reasons, the faces of humans are blurred in all images shown in this report, but for the training and inference, images are directly from the recording without changes.

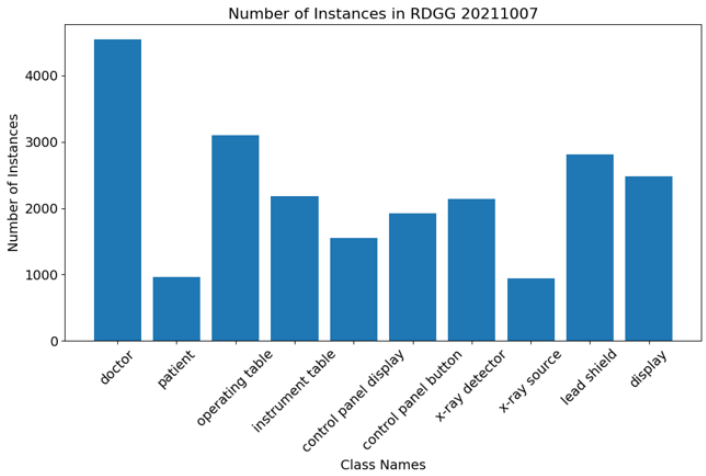


Figure 4.4: The number of object instances of the RdGG 20211007 dataset.

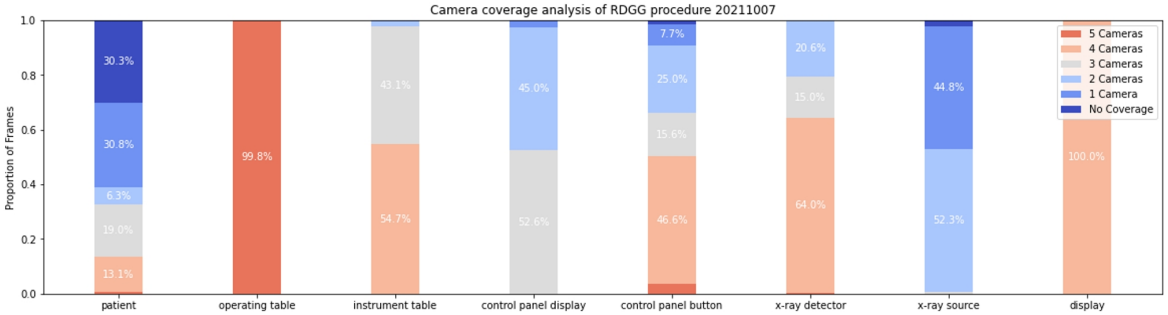


Figure 4.5: Camera coverage analysis of the RdGG 20211007 dataset.

The RdGG 20211007 dataset has 620 images for each camera and thus 3100 images in total. The dataset covers a short period before the patient entered the room, a long procedure period, and a relatively short period when medical staff did some cleaning and organizing after the patient left. The number of object instances is shown in Figure 4.4. The camera coverage analysis of each single-object class is done based on the number of frames where the annotation exists. in Figure 4.5, we can see that most objects have good camera coverage, while the patient and the X-ray source have relatively poor camera coverage. After examining the videos, we found that if the patient is inside the Cath Lab, at least one camera will capture it. The reason for no coverage of the patient is the patient leaving the room. The X-ray source, often placed under the operating table, can be occluded by doctors, the table, and the surgical drape covering the patient.

4.1.2 Philips Best Datasets

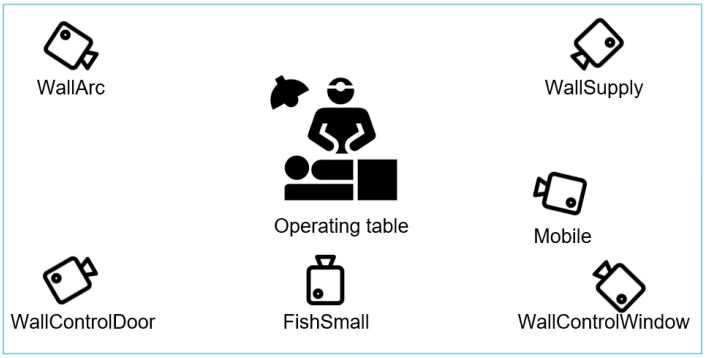


Figure 4.6: The multi-camera system setup in the Cath Lab in Philips Best Campus.

Figure 4.7: Example images from the multi-camera system in the Cath Lab in Philips Best Campus.

The Philips Best 105340 and 100000 are two mock procedures performed in a Cath Lab at the Philips Best campus. The two mock procedures are relatively short compared to the real procedure. In the videos, healthcare professionals are performing actions that simulate real procedures. The reason for using mock procedures for the second Cath Lab is for data access convenience, as the real procedures are only meant to be accessed inside their respective medical institutes. The mock procedures provide high visual fidelity, though future studies may replace them with actual procedures for greater rigor.

The Cath Lab in Philips Best has 6 cameras rather than 5 with a slightly different setup. However, they both serve the same purpose, mounted at the edge of the room, facing the operating table to provide a more comprehensive view and better coverage of the operation. The example images and camera setup can be found in Figure 4.6 and Figure 4.7, respectively. Although some cameras have slightly different frame rates, all are close to 25 FPS. Images in the datasets are also taken every 5 seconds from the videos. The Philips Best 105340 dataset has 492 images, and the Philips Best 100000 dataset has 792 images. The number of object instances of Philips Best 105340 and Philips Best 100000 are shown in Figure 4.8. Figure 4.9 shows the result of the

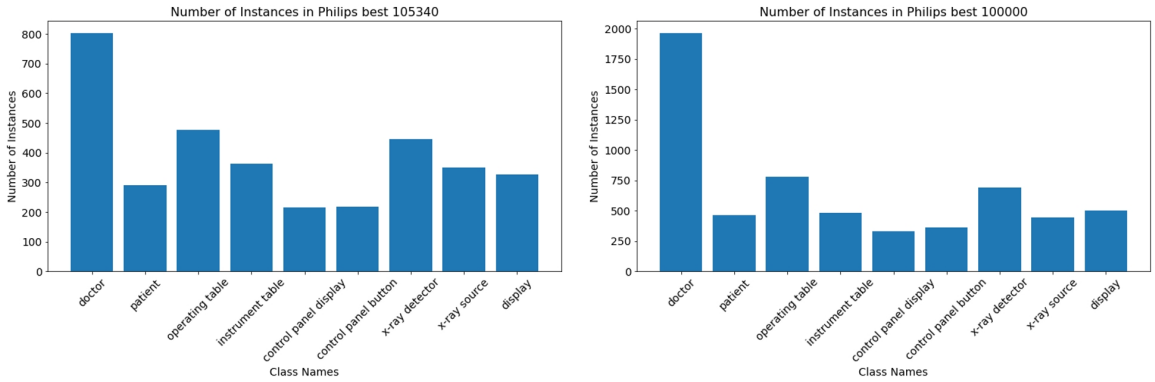


Figure 4.8: The number of object instances of the Philips Best 105340 and 100000 datasets.

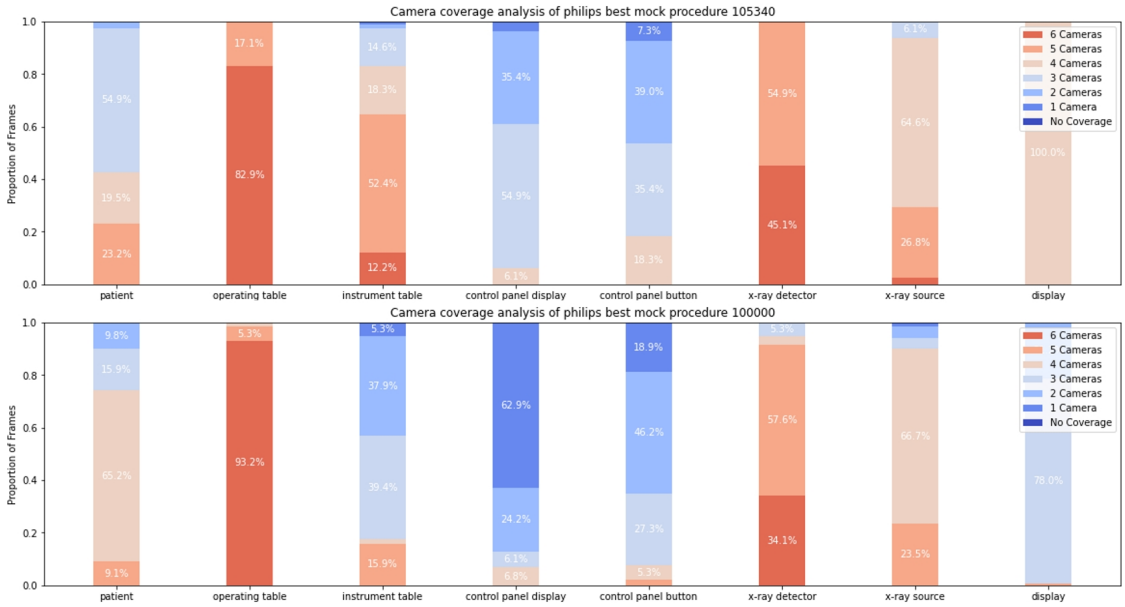


Figure 4.9: Camera coverage analysis of the Philips Best 105340 and 100000 datasets.

camera coverage analysis. The control panel button and the control panel display have relatively bad camera coverage, as they are attached to the operating table at positions that can be easily occluded by healthcare professionals.

4.1.3 Online Image Dataset

The objects inside a single Cath Lab have highly limited appearance diversity, because they are namely same objects moving and performing actions in the same environment. The online images dataset was collected to improve the object and background appearance diversity in our training data. In a manner similar to large-scale datasets like ImageNet and MS COCO, we too collected online images from search engines. This approach is based on the idea that diverse images will have a broader data distribution

in the feature space, which improves the robustness of the trained model.

Figure 4.10: Example images of the online image dataset.

The collection process is simple but labor-intensive. We turned to existing image repositories available in Google Images and Bing Images by searching for the keyword 'Catheterization Laboratory'. Images with a resolution lower than 640×480 were automatically filtered out. All collected images were visually examined. The irrelevant, blurry, or damaged images were manually deleted. The example images are shown in Figure 4.10.

We observed that most images were taken without procedures being performed. Therefore, the patient and surgical drapes are not present. Additionally, because most images are taken by hand-held cameras with aesthetically pleasing angles for advertisement purposes, the viewpoints of those images are highly limited.

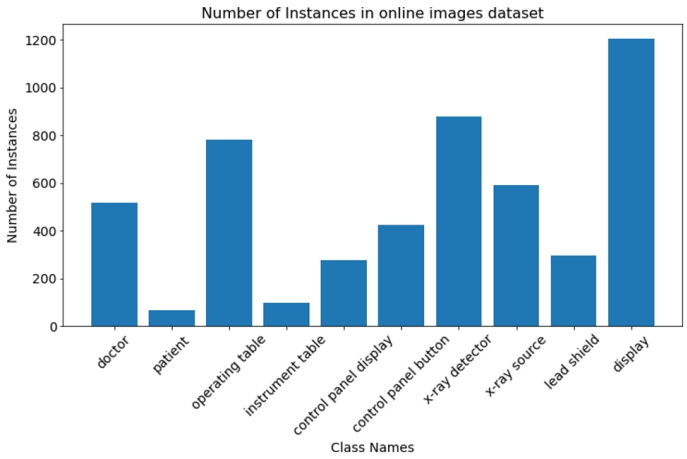


Figure 4.11: The number of object instances of the online image dataset.

The number of object instances of the online images is shown in Figure 4.11. We can see the numbers of patients and the instrument table are much fewer than in other classes. The reason is that most online images are taken when no operation is performed

because of privacy results. The extremely low number of patient and operating table instances will have a negative impact on their detection performance when the online images are used as training data. In future studies, it is possible to add more images of instrument tables inside other types of operating rooms. The lack of patient instances can be solved by detecting humans and then classifying them into medical staff and patients, as there are abundant publicly available images of people and pretrained human detectors.

4.2 Metrics

Object detection is a complex task involving a range of metrics, each of which might have specific implementations depending on the benchmark. In this project, we have chosen an implementation consistent with the widely recognized COCO benchmark. This section first covers those metrics. Then, we move to their aggregated version we created for evaluating the detection performance of the multi-camera system.

4.2.1 Average Precision

Precision and recall are two of the most important metrics in predictive systems especially in machine learning, as they provide insights into practical model performance and have significant business implications

Precision and recall are more naturally defined for classification tasks, such as binary predictions of "Yes" or "No". Precision measures the reliability of the positive predictions (predictions of "Yes") of a model. It is calculated as the proportion of correct positive predictions out of all positive predictions made. Recall tells the ability of a model to find all relevant instances (cases with a "Yes" ground truth). It is the ratio of positive ground truth cases found by the model to all positive ground truth cases. Typically, prediction outcomes are categorized into four categories, which as True Positive (TP), False Positive (FP), True Negative (FN), and False Negative (FN). Positive/negative describes the result of prediction ("Yes" or "No"), and true/false describes the correctness of detection. The detection and recall can be expressed as:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{4.1}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{4.2}$$

Most predictive systems provide a confidence score along with the prediction or a probability, we can set a confidence threshold to decide whether a prediction is positive. Adjusting this threshold allows us to manage the trade-off between precision and recall. Generally, a higher threshold provides a higher precision but a lower recall, and vice versa. By changing the threshold, we can get a precision-recall curve as shown in Figure 4.12 for a more comprehensive understanding of the system's performance.

In object detection tasks, an annotated bounding box of an object instance is considered as a "relevant case" and contributes to the denominator of recall. For a correct

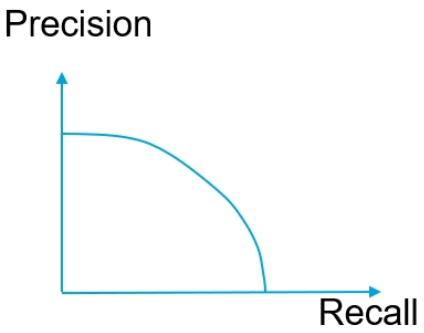


Figure 4.12: An illustration of the precision-recall curve.

detection that successfully found the object instance, it needs a correct class prediction, a confidence score higher than the threshold, and enough localization accuracy. Localization accuracy is quantified by Intersection over Union (IoU) [70]. As its name suggests, IoU is the ratio of intersection to union between the ground truth bounding box and the prediction bounding box. Its value ranges from 0 to 1, with 0 indicating no overlap between the ground truth and predicted bounding boxes, and 1 indicating a perfect match between them.

Additionally, the COCO benchmark has a best-match requirement to discourage redundant bounding boxes of the same object instance. When multiple predicted bounding boxes match an object instance (their IoU and confidence are higher than the thresholds), only the one with the highest IoU is considered as a true positive while the rest will be seen as false positives.

Average precision (AP) and mean average precision (mAP) are commonly utilized metrics for comparing detection performance and providing a basic assessment of the capabilities of a detector [70]. They can be commonly found in the algorithm rankings section of most object detection benchmarks such as the COCO benchmark. In the following sections, we assess AP and mAP under an IoU threshold of 0.5, denoted as AP@0.5 and mAP@0.5, respectively. The selection of an IoU threshold of 0.5 is because it is commonly reported in object detection benchmarks, where it is recognized as a threshold for reliable localization accuracy. While the precision-recall curve provides more comprehensive evaluation results of object detectors, single-number metrics like AP and mAP are preferred for more direct comparison. By its definition, AP is the average precision value across all recall values.

$$\mathrm{AP} = \int_0^1 p(r)\, dr, \tag{4.3}$$

where AP is average precision; p is precision; r is recall.

In practice, the calculation of average AP is performed by summing interpolated precision times its recall interval over given recall thresholds. Interpolated precision is the maximum precision in any recall level greater than or equal to the given recall threshold. It is used to guarantee monotonically decreasing the precision-recall curve, which fits the concept of the precision-recall trade-off. Different benchmarks

have different implementations, such as Pascal-VOC uses 11-point interpolation average precision. We followed the implementation of the COCO benchmark, 101-point interpolated precision, where recall thresholds are $\{0, 0.01, 0.02, ..., 1\}$.

$$\text{AP} = \sum_{k=1}^{n} (r_k - r_{k-1}) \cdot p_{\text{interp}}(r_k), \tag{4.4}$$

$$p_{\text{interp}}(r) = \max_{\tilde{r} \geq r} p(\tilde{r}), \tag{4.5}$$

where AP is average precision; p_{interp} is interpolated precision; r is recall.

Mean Average Precision (mAP) is another widely-used metric, which takes object classes into consideration. It is the mean value of the average precision over each object class, as shown in the equation:

$$\text{mAP} = \frac{1}{N} \sum_{i=1}^{N} \text{AP}_i, \tag{4.6}$$

where AP_i is the average precision of the object class i.

Compared to AP, which assigns equal weight to every bounding box, mAP assigns equal weight to the AP of each class, without being influenced by its object instance number. mAP can provide a more balanced performance measure in case of class imbalance, where classes with higher object instance numbers can be over-represented in AP. For example, in Cath Labs, where there tend to be more medical staff than patients. However, it does not necessarily mean the detection performance of medical staff is more important than patients. Due to the higher number of object instances of medical staff, AP tends to be influenced more by the detection performance of staff detection, but mAP treats the performance of these two classes equally.

4.2.2 Aggregated Metrics for Multi-camera System Evaluation

The goal of multi-camera evaluation is to demonstrate how well multi-camera systems can enhance detection performance by mitigating issues related to changes in viewpoint and occlusion. During our examination of the object detector trained on the online images, we found its performance can be sensitive to viewpoint changes and occlusion. This sensitivity can be attributed to the characteristics of the online images. Because the online images are taken from limited angles for commercial purposes and outside procedural timings, viewpoint changes and occlusion are rare in the online image dataset. This issue can be mitigated by switching to the camera views with less viewpoint change and occlusion at each frame.

Another important reason for multi-camera evaluation is to study the impact of the detection performance on the downstream tasks that utilize multiple cameras. Given that object detectors are a vital component of video analysis systems, the performance of the whole system in downstream tasks becomes a critical area of interest.

Our approach is developing multi-camera evaluation metrics designed to assess the impact of the detector on downstream tasks which vary in their requirement for the

number of cameras. The key advantage of this method is its focus solely on the influence of 2D object detection performance, effectively excluding any effects attributable to other algorithms within the video analysis pipeline. In Chapter 2, we have introduced various applications oe multi-camera systems such as fall detection and 3D human pose estimation. However, conducting tests on all these applications would be beyond the time constraints of our project. More importantly, our detector detects medical instruments while the methods introduced only analyze human behavior. Nevertheless, the integration of medical instrument detection in data analysis is expected to follow a similar methodology. Specifically, we cover applications requiring 1 or 2 camera(s), though it can be extended to more cameras for more sophisticated applications.

1. **Applications require 1 camera:**
   Applications that are only interested in the state of the objects. In this case, object detectors only need to propose the region of the object and other algorithms then analyze this region. Examples can be detecting if a person falls over or if the control panel button has been touched by medical staff.

2. **Applications require 2 cameras:**
   Applications that are interested in the 3D position of objects. In this case, object detectors need to propose the region of the object in at least 2 cameras. The keypoint detection and matching algorithm then detects keypoints (such as the joints of people) in the region and matches them across views. Finally, triangulation or a more sophisticated model merges them into 3D locations. Examples can be analyzing the activities of medical staff and calculating the distance from the X-ray source to people for studying X-ray exposure.

Given the number of cameras (N) needed for different applications, we evaluate the performance of multi-camera systems based on the number of frames. We choose the suitable cameras at each frame for each object based on the detection confidence score of the object, selecting the top-N camera views with the highest confidence scores. For simplicity in our evaluation approach, we only evaluate classes with a single object inside the Cath Lab to avoid the matching problem (Note: This evaluation is applied to all classes except doctors). By adopting this method, in each camera view, we select the most confident detection of each class for the object. Matching objects of the same class from multiple camera views can be reserved for future research.

The selected N detections from N cameras for each object are consolidated into a single, aggregated detection. This evaluation method simulates whether this aggregated detection can produce an accurate output for the video analysis pipeline. For each object, each frame is classified into one of three categories based on the visibility of the object and the correctness of the aggregated detection. We calculate the numbers of these three types of frames as follows:

1. **The number of frames that an object can be detected (detectable frames)**
   It means the frame when an object is visible in at least N cameras. It can be

calculated based on the number of annotations that exist in the current frame.

$$N_{\text{detectable}} = \sum_{i=1}^{M} (\text{Visibility}_i \geq N), \tag{4.7}$$

where i is the frame index; M is the total frame number; Visibility_i is the number of annotations of this object in all cameras at frame i.

2. **The number of frames that the detector gives an aggregated detection of the object (detected frames)**
   If all the selected N cameras have detection confidence higher than the confidence threshold. We will consider it as a valid aggregated detection. Otherwise, we will reject it as the detector is not confident enough.

$$N_{\text{detected}} = \sum_{i=1}^{M} \left[\prod_{j=1}^{N} \mathbb{1} \left(\text{Conf}_{ij} \geq \text{Conf}_{\text{thresh}} \right) \right], \tag{4.8}$$

   where N is the number of selected cameras; $\mathbb{1}()$ is the indicator function, which returns 1 when the input is true and 0 when the input is false; Conf_{ij} is the detection confidence of the object in frame j and camera i; $\text{Conf}_{\text{thresh}}$ is the detection confidence threshold.

3. **The number of frames that the aggregated detection is correct (correctly detected frames)**
   A correct aggregated detection means all selected cameras have a detection confidence score that is high enough and localize the object accurately enough. If all the selected N cameras have detection confidence higher than the confidence threshold and IoU with the ground truth higher than the IoU threshold, we consider that the object is correctly detected in this frame. Only in this case, 3D localization can yeild the correct location of the object.

$$N_{\text{correctly detected}} = \sum_{i=1}^{M} \left[\prod_{j=1}^{N} \mathbb{1} \left(\text{Conf}_{ij} \geq \text{Conf}_{\text{thresh}} \wedge \text{IoU}_{ij} \geq \text{IoU}_{\text{thresh}} \right) \right], \tag{4.9}$$

   where $\text{IoU}_{\text{thresh}}$ is the IoU between detection between the ground truth in frame j and camera i; $\text{IoU}_{\text{thresh}}$ is the IoU threshold.

The concept of these three frame types directly relates to the components used in calculating precision and recall. The number of detectable frames corresponds to all ground truth instances. The number of detected frames corresponds to all positive detections. Lastly, the number of correctly detected frames corresponds to true positive detections. Therefore, we substitute the components in the precision and recall to create their aggregated version as follows:

$$\text{Precision}_{\text{aggregated}} = \frac{\sum_{i=1}^{M} \left[\prod_{j=1}^{N} \mathbb{1} \left(\text{Conf}_{ij} \geq \text{Conf}_{\text{thresh}} \wedge \text{IoU}_{ij} \geq \text{IoU}_{\text{thresh}} \right) \right]}{\sum_{i=1}^{M} \left[\prod_{j=1}^{N} \mathbb{1} \left(\text{Conf}_{ij} \geq \text{Conf}_{\text{thresh}} \right) \right]}, \tag{4.10}$$

$$\text{Recall}_{\text{aggregated}} = \frac{\sum_{i=1}^{M} \left[\prod_{j=1}^{N} \mathbb{1} \left(\text{Conf}_{ij} \geq \text{Conf}_{\text{thresh}} \wedge \text{IoU}_{ij} \geq \text{IoU}_{\text{thresh}} \right) \right]}{\sum_{i=1}^{M} \mathbb{1} \left(\text{Visibility}_i \geq N \right)}. \tag{4.11}$$

In the context of object detection metrics like AP and mAP, which are fundamentally derived from precision and recall, we propose an aggregated version by substituting the standard precision and recall with their aggregated forms.

Those aggregated metrics account for the redundancy of the multi-camera system, which is a factor not considered in the object detection metrics discussed in Section 4.2.1, where each bounding box is treated independently. For instance, consider a scenario within a 5-camera system involving two frames. In one case, the object is successfully detected in 2 camera views in each frame. In another case, the object is detected in 4 camera views in the first frame but is missed in the second frame. Traditional AP would yield identical results for both scenarios. However, a 3D localization system would be able to track the object in both frames in the first scenario, whereas it would lose track of the object in one frame in the second scenario. The aggregated version of AP distinguishes between these two cases, offering a more realistic evaluation of the object detection performance in multi-camera systems.

4.3 Experiment design

Experiments in this project are divided into three parts, which are "performance gap and distribution shift", "generalization to unseen Cath Labs", and "multi-camera evaluation". They are designed to answer the following questions:

1. How large is the performance gap when the model is deployed in the same Cath Lab and in a different Cath Lab?
2. Is the performance drop caused by data distribution shifts?
3. Can the object detector trained on online images generalize to unseen Cath Labs?
4. Can the multi-camera system improve the detection performance?

4.3.1 Performance Gap and Distribution Shift

For the experiment that shows the performance gap, we aimed to train the model on different training sets and compare their performance on the same testing set to simulate different scenarios. The testing set we chose is Philips Best 105340. The training sets we chose are:

1. For performance in the same Cath Lab:
   The Philips Best 100000 dataset
2. For performance in different Cath Labs:
   The RdGG 20211007 dataset

3. For performance from diverse non-sensitive images:
   The online image dataset

In addition to showing the performance gap, we also want to show that the performance gap is caused by the different distribution of our datasets in the feature space. For comparison, we will also show the data distribution of all our four datasets (feature extraction is from the pretrained model weight on the MS COCO dataset). The dataset visualization results serve as the following comparisons:

1. Data distribution in the same Cath Lab:
   Philips Best 105340 vs. Philips Best 100000

2. Data distribution from two different Cath Labs:
   Philips Best 105340 (100000) vs. RdGG 20211007

3. Data distribution of online images and procedure recording:
   Online images vs. Philips Best 105340 (100000) and RdGG 20211007

4.3.2 Generalization to Unseen Cath Labs

The second experiment is to verify whether the model trained purely on online images can generalize to previously unseen Cath Labs. We trained our model on online images without using any procedure recordings and tested the trained model on the three procedure recording datasets, which are Philips Best 105340, Philips Best 100000, and RdGG 20211007.

We are curious about if the detection performance of each class is related to the object-level data distribution. Therefore, after obtaining the detection performance in the unseen Cath Labs, we also visualized the object-level data distribution of the best-performing class and the worst-performing class.

4.3.3 Multi-camera System Evaluation

The final experiment is to verify if the multi-camera system can further improve detection performance in unseen Cath Labs compared to a single camera. So we perform multi-camera evaluation via the aggregated metrics. It is done by testing the detector trained on online images in three procedure recording datasets. which are Philips Best 105340, Philips Best 10000, and RdGG 20171007. And we will choose the number of selected cameras N=1 and 2, for simulating applications:

1. N=1: Applications that need to know the state of the object.

2. N=2: Applications that need to know the 3D space of the object.

4.4 Implementation Details

This section gives implementation details about the training and evaluation processes.

When using our datasets for training, we further divide each of them into a training set and validation set. The weight of the model is first initiated with the COCO pretrained weight, then updated based on the empirical loss on the training set, while the loss on the validation set is used for hyperparameter tuning. The early stopping function also depends on the validation set, it evaluates validation loss after every epoch and stops training when the validation loss begins to increase to prevent over-fitting. We splitted 25% of each dataset as their validation set and the rest was used as the training set. For procedure recordings, the validation set is extracted at the end of the video rather than random selection, because we want to make the validation set more independent of the training set. Because the RdGG 20211007 dataset has a considerable period when the patient has left the Cath Lab, the validation set is chosen at the end of the period when the patient was in the Cath Lab and the period it left the Cath Lab. We adjusted the default training hyperparameter of YOLOv8 to achieve the best performance on the validation set. The adjusted parameters of different training data are shown in Table 4.1. During training and inference, images are resized to 640 in width while keeping their aspect ratio.

Table 4.1: Adjusted hyperparameters (training function arguments) when using the following datasets for training the YOLOv8 object detector.

Function argument	Trained on RdGG 20211007	Trained on Philips Best 100000	Trained on Online Images
epochs	100	100	50
lr0	0.0005	0.0005	0.0005
batch	-1(auto)	-1(auto)	-1(auto)
patient	10	10	10
close_mosaic	30	30	30
imgsz	640	640	640
workers	0	0	0

For object detection performance evaluation, we followed the implementation of COCO datasets. We used the official APIs from the COCO benchmark, a Python library called pycocotools. The detection results and annotations are fed to pycocotools to obtain the precision array, which are precision values under the 101 recall thresholds. Then we calculate the average precision based on this precision array. A detail to note is that when the detector can not reach a recall of 1, the precision under the recall threshold that it can not reach will be set to 0. For the aggregated metrics, we wrote our own implementation. It involves calculating the IoU of each detection with its corresponding ground truth and calculating the aggregated precision array. The calculation of precision and recall followed the same method of pycocotools, except we substituted the nominator and denominator of precision and recall with our own definition previously discussed.

4.5 Results and Analysis

In this section, results are grouped according to the experiment design. Additionally, a failure analysis section provides a summary of the reasons for failure, along with images of the failure cases.

4.5.1 Performance Gap and Distribution Shift

For the detectors trained on the online image, Philips Best 100000, or RdGG 20211007 dataset, their performance evaluated on the Philips Best 105340 dataset is shown in Table 4.2.

Table 4.2: AP@0.5 of the detector tested on Philips best 105340 dataset, when the model is trained on different datasets.

Class	Philips Best 100000	RdGG 20211007	Online images
Doctor	0.942	0.778	0.831
Patient	0.928	0.118	0.516
Operating table	0.929	0.404	0.581
Instrument table	0.830	0.240	0.670
Control panel display	0.849	0.183	0.214
Control panel button	0.930	0.059	0.054
X-ray detector	0.976	0.012	0.766
X-ray source	0.853	0.047	0.630
Display	1.000	0.775	1.000
Mean	0.915	0.291	0.585

The results suggest that training the detector in the same Cath Lab significantly outperforms training in a different Cath Lab. The detector trained on the Philips Best 100000 dataset achieved an mAP@0.5 of 0.915, compared to an mAP@0.5 of 0.291 for the detector trained on the RdGG 20211007 dataset. A closer examination of each class reveals that, for the detector trained in the same Cath Lab, the AP@0.5 for all classes is higher than 0.8. In contrast, the model trained in a different Cath Lab failed badly in detecting the X-ray detector and the X-ray source. However, it maintained moderately effective detection of doctors and displays, with AP@0.5 scores of 0.778 and 0.775, respectively. This can be attributed to the high visual similarities of these two classes in both Cath Labs. In both Cath Labs, doctors wear blue coats with hats and face masks, and the displays are highly similar, likely being of the same type.

The performance of the model trained on online images falls in between, achieving 0.585 mAP@0.5. It outperforms the model trained in a different Cath Lab, yet it does not match the effectiveness of the detector trained in the same Cath Lab. This model also shows potential weaknesses in detecting certain classes. For instance, the control panel button, with an AP@0.5 of only 0.054, highlights concerns about the inconsistent performance of the object detector across different classes.

Another key point is that merely increasing the quantity of images from a Cath Lab may not enhance the cross-room detection capability. The online image dataset and the Philips Best 100000 dataset contain similar numbers of images, 800 and 720

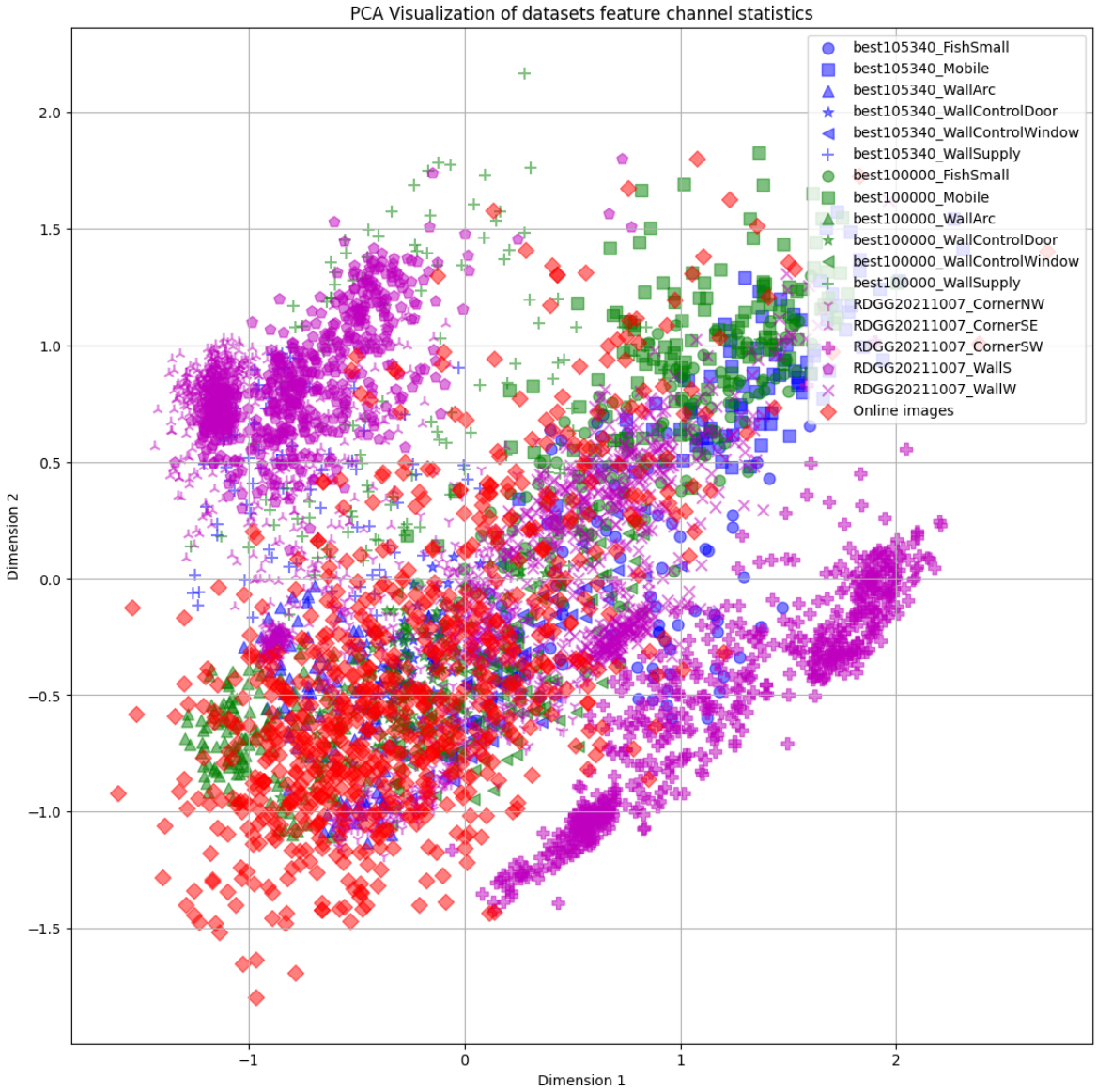


Figure 4.13: PCA data distribution visualization of our datasets.

respectively. In contrast, the RdGG 20211007 dataset has a significantly larger dataset size of 3100 images. Nevertheless, when used as training data, the RdGG 20211007 dataset provides the poorest performance on the Philips Best 105340 dataset. This outcome is likely due to the high similarity of images collected within the same Cath Lab.

To explain the performance difference, the experiment is complemented by visualizing the data distribution of our four datasets in the feature space. We noticed our datasets have their unique data distribution characteristics, which relate to the detection performance in the experiments. The visualization results utilizing PCA, T-SNE, and UMAP are shown in Figure 4.13, Figure 4.14, and Figure 4.15, respectively. Ev-

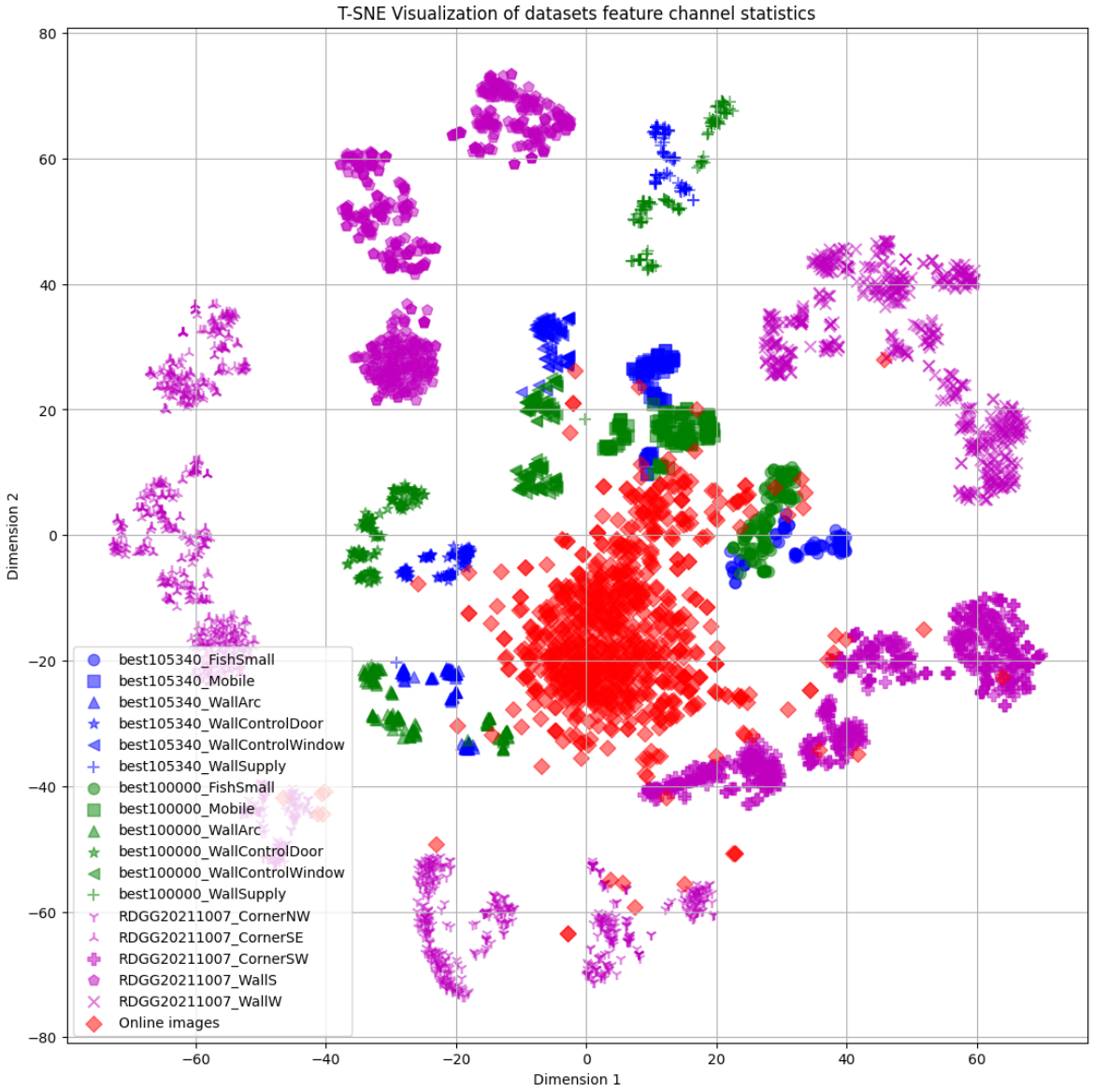


Figure 4.14: T-SNE data distribution visualization of our datasets.

ery point in those figures is the reduced feature vector of an image in our datasets. Their colors denote the datasets and their marks denote the camera view. We can see that the visualization result from PCA is highly overlapping, as it does not model non-linearities, while T-SNE and UMAP give good and consistent visualization results.

Our datasets exhibit their unique characteristic of data distribution. For procedure recordings in Cath Labs, their data points are narrowly clustered around each camera view. Images from the same Cath Labs have a very close data distribution. However, images in different Cath Labs have distinct data distributions. The online images, compared to procedure recordings, have a wider data distribution, though the dataset size is relatively small, so some data points scatter sparsely in certain regions.

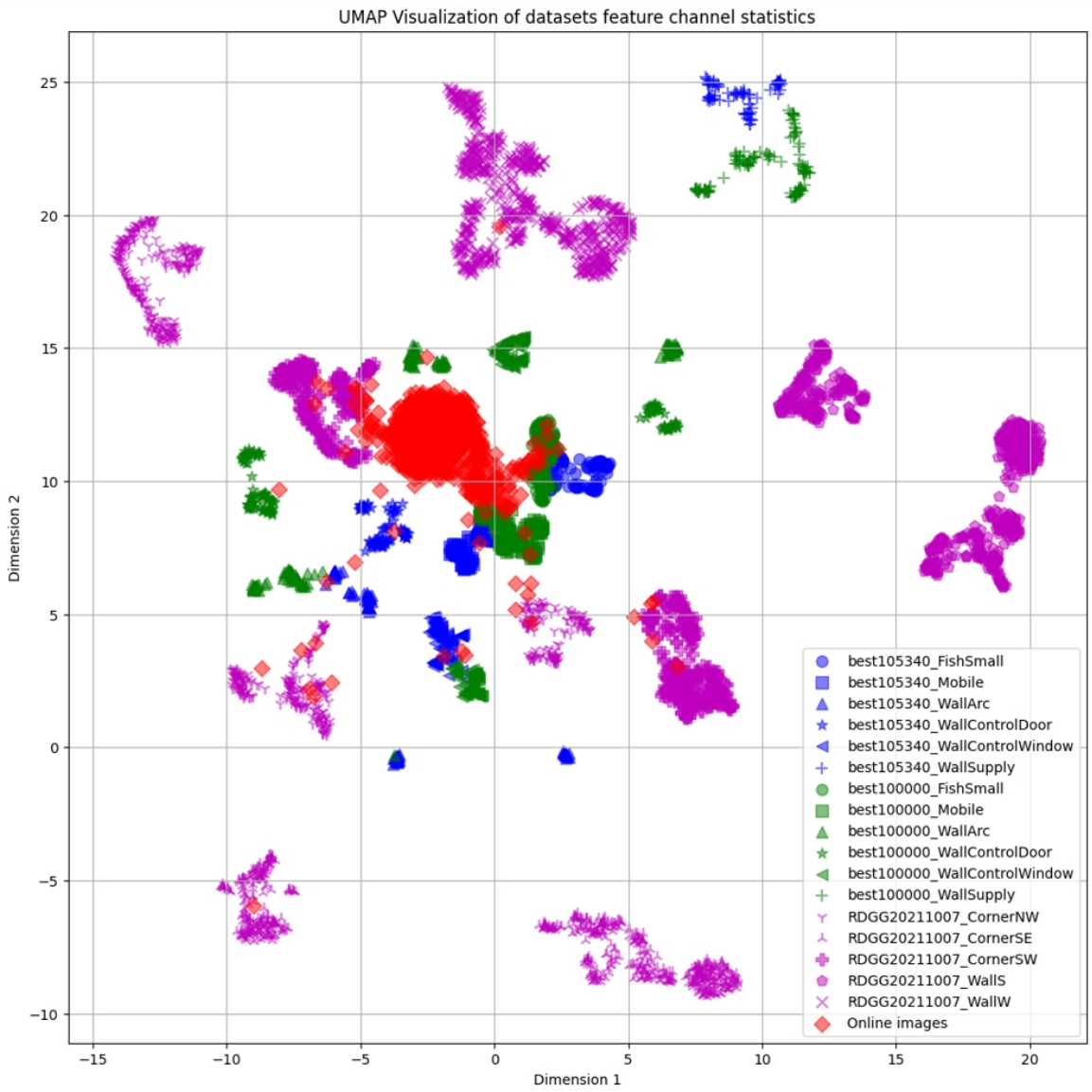


Figure 4.15: UMAP data distribution visualization of our datasets.

The distinct characteristics of data distribution in our datasets provide insights into the varied detection performance observed in the experiment. Images originating from the same Cath Lab exhibit similar data distributions, which explains why the detector trained and tested in the Philips Best Cath Lab shows the most impressive performance. The online images have a wider yet different data distribution to the procedure recordings. It partially covered the data distribution of the procedure datasets. Therefore, a model trained on online images has a moderately good performance, but it is inferior to the one trained on images from the same Cath Lab. Lastly, the data distribution of images from RdGG and Philips Best are highly different, which explains the worst performance of the model trained and tested on images from different Cath Labs.

4.5.2 Generalization to Unseen Cath Labs

Table 4.3: Evaluation results (AP@0.5) of the detector in different Cath Labs, when the detector is trained purely on the online images (Note: The result is consistent when the relative difference $\frac{|A-B|}{|A|+|B|} \times 2 \leq 0.2$ for each pair).

Class	Philips Best 105340	Philips Best 100000	RdGG 20211007	Consistent?
Doctor	0.831	0.870	0.856	Yes
Patient	0.516	0.485	0.274	No
Operating table	0.581	0.649	0.704	Yes
Instrument table	0.670	0.616	0.266	No
Control panel display	0.214	0.152	0.738	No
Control panel button	0.054	0.115	0.311	No
X-ray detector	0.766	0.684	0.709	Yes
X-ray source	0.630	0.681	0.071	No
Display	1.000	0.974	0.727	No
Mean	0.585	0.581	0.517	Yes

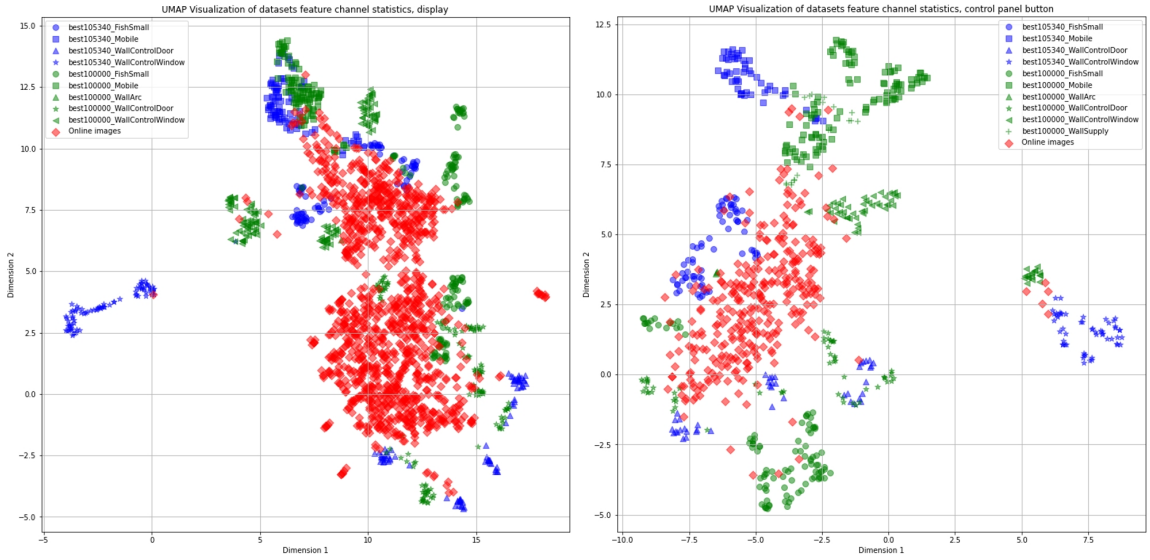
We were curious how well the model trained on online images can generalize to unseen Cath Labs. Accordingly, we tested it on our procedure recording datasets, with the results detailed in Table 4.3.

The results suggest that the detector trained purely on online images can generalize to previously unseen Cath Labs with a moderately good performance. However, certain classes can have poor detection performance, such as the control panel button and the X-ray detector in the RdGG dataset. We added an extra column to the table to indicate the performance consistency of each class, based on the relative differences in their performance across different datasets. Only three out of nine object classes, namely doctors, operating tables, and X-ray detectors, exhibit consistent detection performance. This performance inconsistency might be attributable to the narrow yet distinct data distribution of procedure recordings, potentially leading to significant variance in detection performance across different Cath Labs. This situation poses challenges to the safety of deploying the detector in unseen Cath Labs, where the performance of the object detector can be unpredictable.

To gain a deeper understanding of the relationship between the detection performance variability and data distribution, we further conducted a detailed visualization of the object-level data distribution for the best-performing and worst-performing classes.

Figure 4.16 shows visualization results for the Philips Best datasets. We can see that the well-detected display has closer data distributions of the training and testing sets, while the badly-detected control panel display has more divergent data distributions. This phenomenon is visible but not very obvious in this case because we calculated the feature vector from a 5×5 region in the feature map and the control panel display is placed close to the operating table. Therefore, the higher visual similarity of the operating table can create interference, thereby weakening this effect.

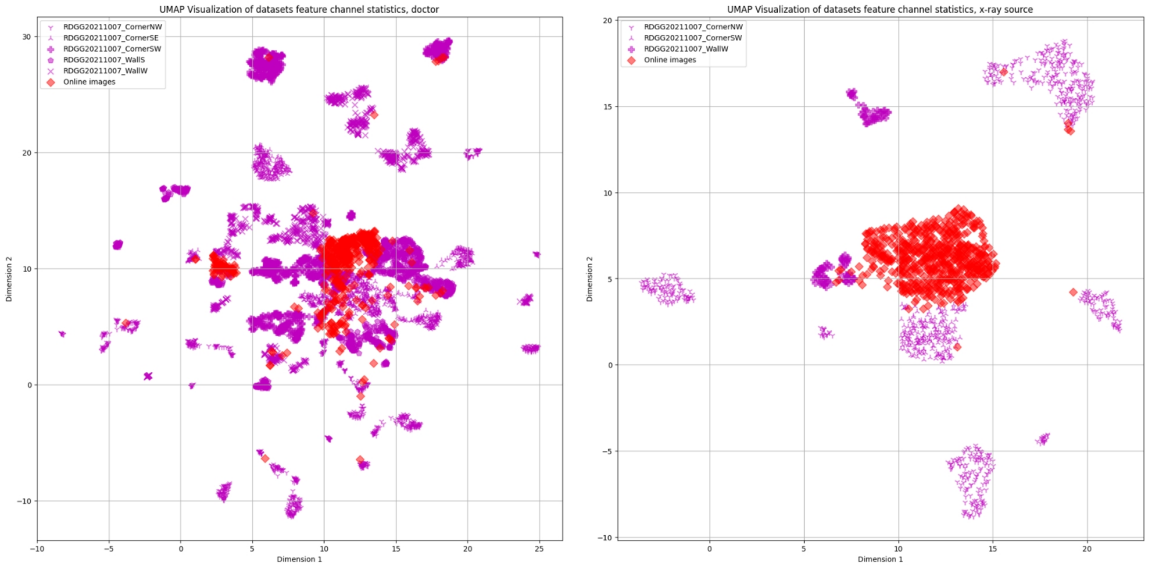
The visualization result of the RdGG dataset is shown in Figure 4.17. The result can better reflect the influence of data distribution on the detection performance. For the well-detected doctor class, the data distributions in the training and testing sets



Display
AP@0.5=1.000 in **Philips Best 105340**
AP@0.5=0.974 in **Philips Best 100000**

Control panel button
AP@0.5=0.054 in **Philips Best 105340**
AP@0.5=0.115 in **Philips Best 100000**

Figure 4.16: UMAP object-level visualization result of the best and worst performing class in the Philips Best datasets.



Doctor
AP@0.5=0.856 in **RdGG20211007**

X-ray source
AP@0.5=0.071 in **RdGG20211007**

Figure 4.17: UMAP object-level visualization result of the best and worst performing class in the RdGG dataset.

are observed to be similar. However, the poorly-detected X-ray source class has very distinct data distributions in the training and testing sets.

4.5.3 Multi-camera Evaluation

Table 4.4: Aggregated AP@0.5 of model trained on the online images and evaluated in different Cath Labs when using the most confident 1 or 2 camera(s) (Numbers higher than 0.7 are highlighted in bold and green).

	Aggregated 1-camera AP@0.5			Aggregated 2-camera AP@0.5		
	Philips Best 105340	Philips Best 100000	RdGG 20211007	Philips Best 105340	Philips Best 100000	RdGG 20211007
Patient	**0.947**	**0.832**	0.505	0.570	0.459	0.053
Operating table	**0.884**	**0.934**	**0.958**	**0.764**	**0.827**	**0.803**
Instrument table	**0.896**	**0.793**	0.585	**0.809**	0.496	0.076
Control panel display	0.440	0.334	**0.897**	0.105	0.031	0.596
Control panel button	0.041	0.211	0.635	0.012	0.026	0.227
X-ray detector	**0.956**	**0.986**	**0.798**	**0.918**	**0.946**	0.605
X-ray source	**1.000**	**1.000**	0.053	**0.908**	**0.790**	0
Display	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
Mean	**0.771**	**0.761**	0.679	0.636	0.572	0.420

Having demonstrated that the model trained on online images can adapt to unseen Cath Labs with reasonably good performance, our next objective was to enhance its effectiveness using multi-camera systems. The outcomes of this multi-camera system evaluation are presented in Table 4.4. Within the table, values exceeding 0.7 are highlighted to indicate good performance.

The multi-camera system shows promising performance in detecting operating tables, X-ray detectors, and displays for applications that require one camera. Additionally, it shows promising performance in detecting operating tables and displays for applications that require two cameras. However, given that our evaluation was conducted using data from only two Cath Labs, coupled with the significant variation in data distribution across different Cath Labs, additional safety measures should be implemented before deploying this system for the detection of these objects.

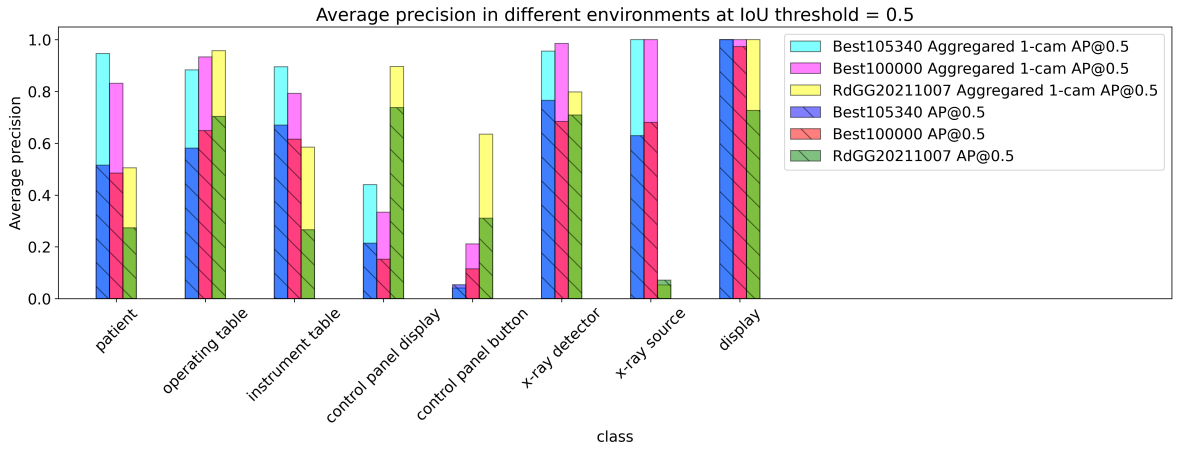


Figure 4.18: AP@0.5 and aggregated 1-camera AP@0.5 when the detector is evaluated on the three procedure datasets.

To illustrate the performance improvement from multi-camera systems, we conducted a comparison between the standard AP@0.5 and the aggregated 1-camera AP@0.5, as depicted in Figure 4.18. The result suggests the multi-camera system can significantly improve the performance in most instances. However, two exceptions exist, which are detecting the control panel button in the Philips Best 105340 dataset and detecting the X-ray source in the RdGG 20211007 dataset. In these two cases, the detector fails badly with very low AP@0.5, and the extremely low confidence scores given by the detector are not informative, resulting in ineffective camera selection.

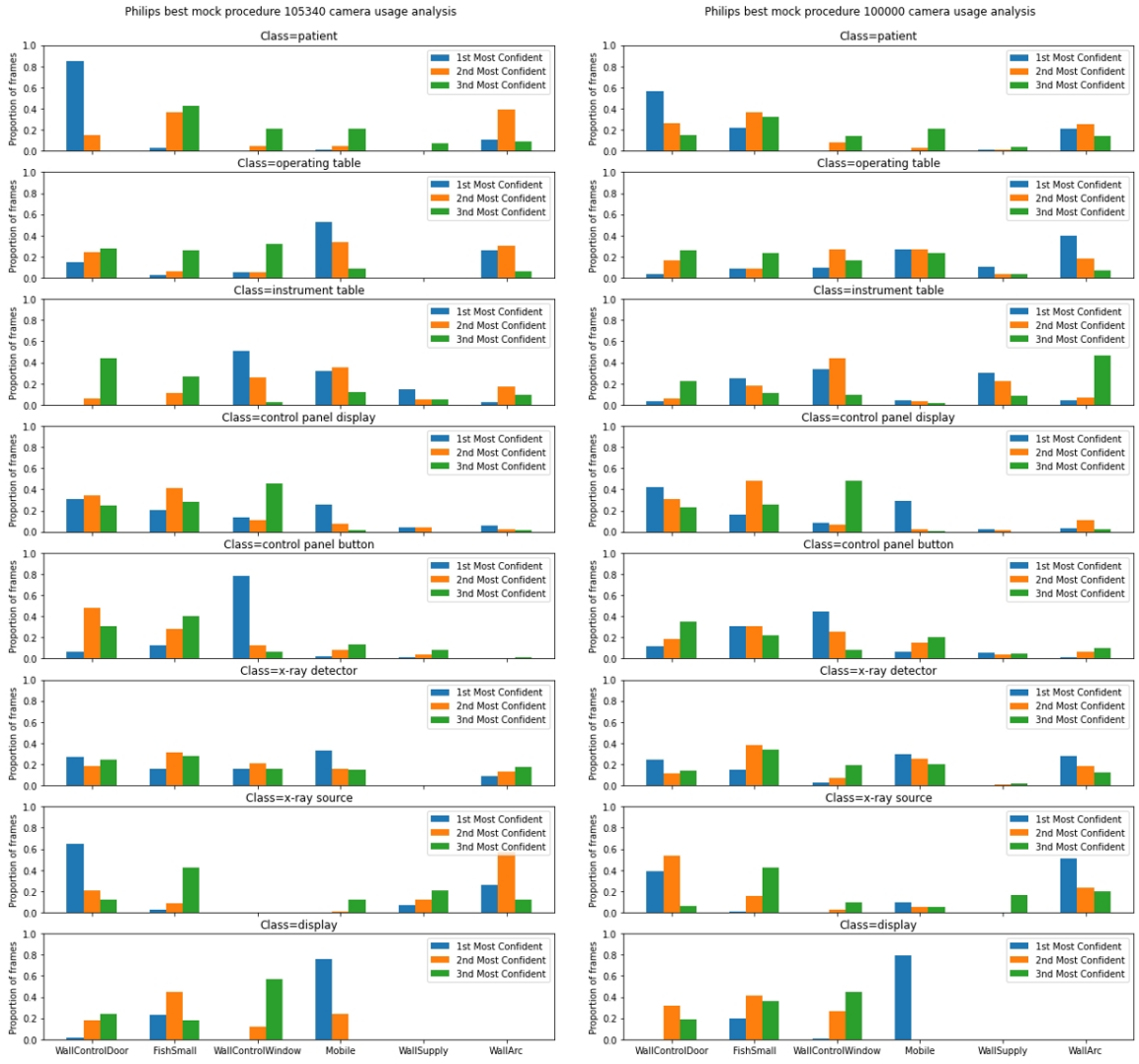


Figure 4.19: Camera usage analysis on the Philips Best 105340 and 100000 datasets.

The analysis of camera usage in both Cath Labs is depicted in Figure 4.19 and Figure 4.20. This analysis validates the camera setups in each Cath Lab, demonstrating that every camera has functioned as the most confident choice at some point during the procedures. However, it's noteworthy that certain classes exhibit a preference for one specific camera over others.

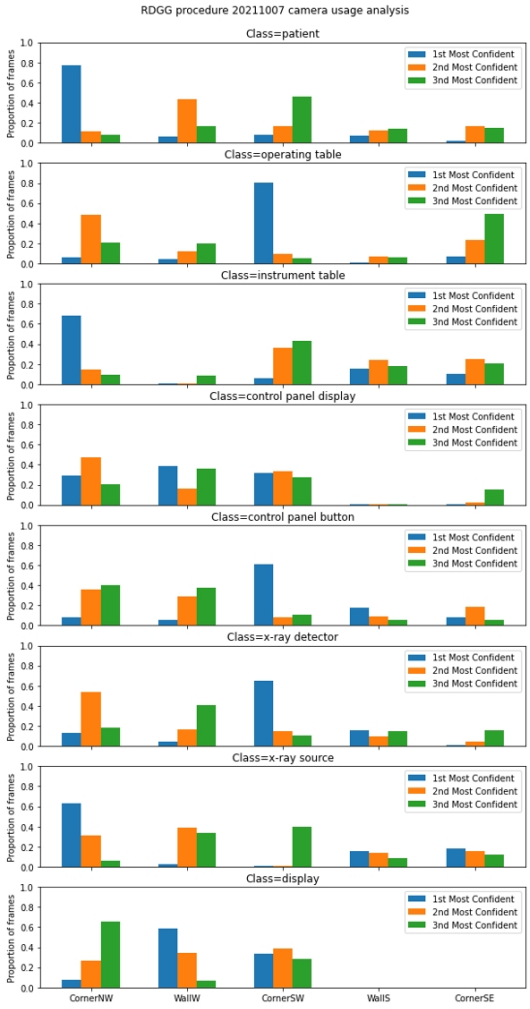


Figure 4.20: Camera usage analysis on the RdGG 20211007 dataset.

4.5.4 Failure Cases

Table 4.5: Summary of major failure reasons in different Cath Labs when the model is trained on the online images.

Class	Philips Best 105340+100000	RdGG 20211007
Patient	Viewpoint, unseen action, occlusion	Viewpoint, unseen action, occlusion
Operating table		Viewpoint(when undraped)
Instrument table	Occlusion	Viewpoint, absence of sterile sheet
Control panel display	Occlusion, viewpoint, plastic film coverage	Occlusion, viewpoint, plastic film coverage
Control panel button	Occlusion, viewpoint, plastic film coverage	Viewpoint, occlusion, plastic film coverage
X-ray detector	Viewpoint	Viewpoint
X-ray source	Viewpoint, occlusion	Plastic film coverage
Display		Viewpoint

For the detector trained on online images and tested on procedure recordings, we visually examined failure cases in our testing set and summarized the major failure

reasons for each class in Table 4.5. The major reasons for detection failure are viewpoint changes, occlusion, and coverage from plastic film.

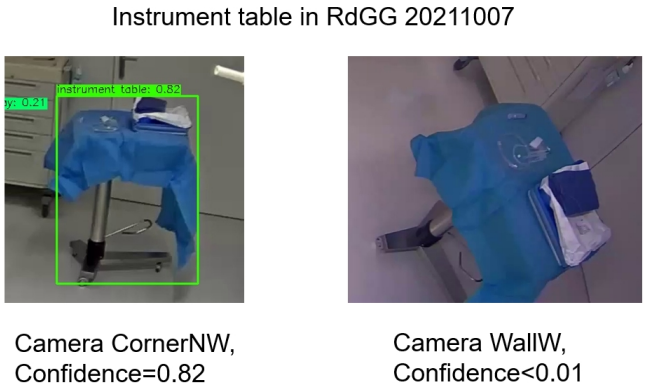


Figure 4.21: Example of failed detection due to viewpoint change from camera setup.

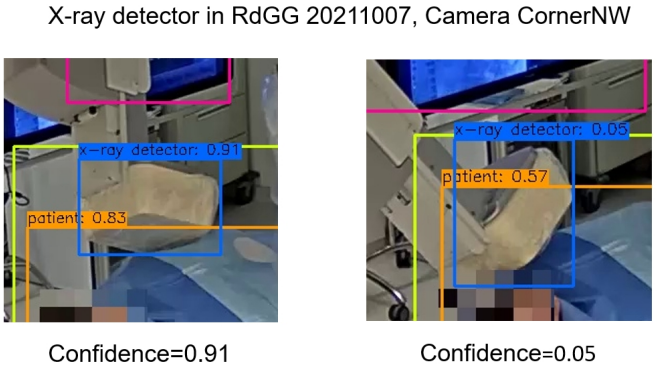


Figure 4.22: Example of failed detection due to viewpoint change from object movement.

Viewpoint changes, which affect detection performance, can arise from both the camera setup and the movement of objects. Its impact on detection performance can be coupled with the state of the object, such as whether the operating table is covered by the surgical drape. The example images for viewpoint-related detection failures are shown in Figure 4.21, Figure 4.22, and Figure 4.23. In these cases, we can see the instrument table is detected in camera CornerNW but missed in camera WallW. The detection confidence score of the X-ray detector dropped sharply because its movement changed the viewpoint. And for the operating table, its detection from camera WallW failed only when it was not covered by the drape. The reason could be the drape has high visual similarity from different viewpoints.

The issue of occlusion is a common challenge for most object detectors. When parts of an object are covered, features corresponding to the covered parts will have a weaker response inside the object detection model, resulting in a drop in the detection confidence score. However, in this project, the issue of occlusion can be particularly

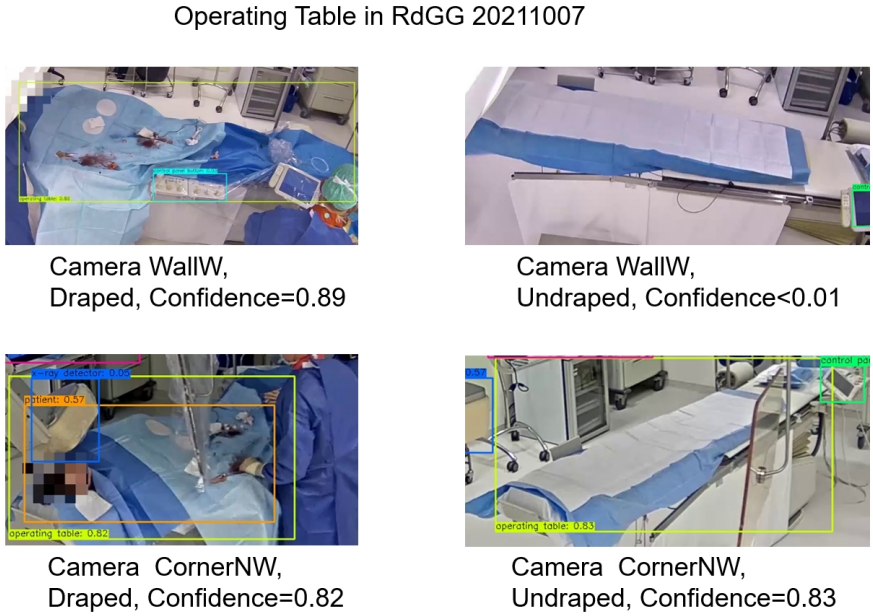


Figure 4.23: Example of failed detection due to viewpoint only when the operating table is undraped.

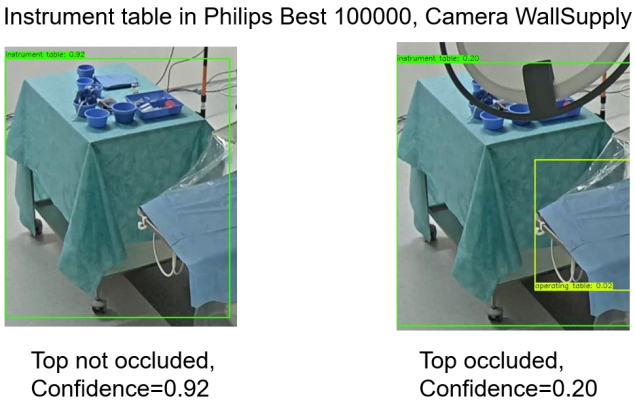


Figure 4.24: Example of failed detection due to occlusion

severe for certain classes. This situation stems from the nature of our training data. The online images are mostly taken when no operation is performed in the Cath Labs, resulting in less occlusion of objects. Consequently, our trained detector becomes more sensitive to occlusion. Figure 4.24 shows an example where the instrument table is partially occluded. Even though only a small part of the instrument table's top is occluded, the detection confidence score drops from 0.92 to 0.2. This example suggests that the features used to identify instrument tables seem to predominantly rely on the blue surgical bowls placed on top of them.

Plastic film coverage is a unique problem in operating rooms, including Cath Labs.

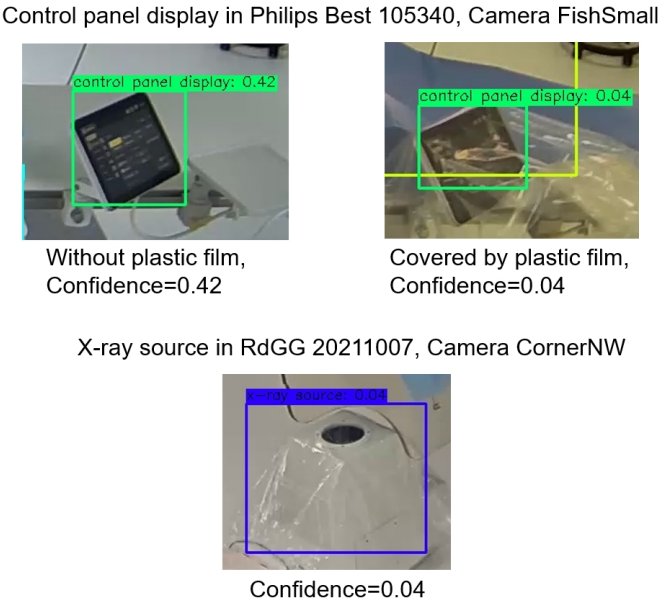


Figure 4.25: Example of failed detection due to plastic film coverage.

The plastic film is usually used to cover medical instruments for hygiene, and it affects the control panel displays, the control panel buttons, and the x-ray sources in our datasets. Figure 4.25 shows the impact of plastic film coverage on the detection confidence. We can see a drastic drop in the detection confidence of the control panel display in the Philist Best 105340 dataset. In the RdGG 20211007 dataset, the detection of the X-ray source constantly failed, highly likely due to plastic film coverage. As the online images are mostly taken with no operation being performed, the plastic film is often absent in the online image dataset. Therefore, the object detector trained on online images may have difficulties in recognizing objects covered by plastic film.

5 Conclusion

5.1 Discussions

5.1.1 Implications of the Research

In summary, the experiment results show a promising aspect: moderately good detection performance in unseen Cath Labs can be achieved with only publicly available images as the training data. The detection performance can be further enhanced with a multi-camera system. This research lays a promising foundation for developing robust, vision-based workflow analysis systems for deployment in unseen Cath Labs. However, the results also highlight concerns about the uncertainty in object detection performance inside an unseen Cath Lab, which calls for more safety measures.

In Section 4.5.1, we have observed that the YOLOv8 model, when trained on diverse online images, significantly outperforms the same model trained on data from a different Cath Lab. Further experiments in Section 4.5.2 show the model trained on online images can generalize to our datasets from two Cath Labs with moderately good detection performance. Additionally, the data visualization results shed light on the connection between the data distribution in the feature space and the observed variations in detection performance. The online image dataset has a wider data distribution. Therefore, the loss function of YOLOv8 can be optimized in large regions in the feature space. Images from new Cath Labs have a higher chance of falling into the proximity of the optimized regions, obtaining better detection performance. Moreover, the object detection performance can be further improved with the multi-camera system. As online images have less occlusion and viewpoint diversity, the detector trained on them can be sensitive to occlusion and viewpoint changes, often resulting in low detection confidence when encountering them. By switching to the camera with the highest confidence in the multi-camera system, the detector can operate on views with less occlusion and fewer viewpoint changes, thereby achieving improved detection performance.

The issue of performance inconsistency is a significant finding in our project, carrying serious implications. We observed this inconsistency in the detection performance when testing the detector with images from different Cath Labs. Notably, certain objects, such as the control panel button in the Philips Best dataset and the X-ray source in the RdGG dataset, yielded particularly poor detection results. The visualization of object-level data distribution provides insight into the cause of these poor detection outcomes: these objects have significantly different data distributions compared to the training data. Additionally, a concerning observation from our study is the narrow and distinct data distribution of images from different Cath Labs, coupled with the fact that only a limited number (two) of Cath Labs have been analyzed in this project. Consequently, when encountering a new Cath Lab, it can be uncertain how their data

distributions will vary from our existing datasets, and as a result, poor detection performance could potentially occur. On the bright side, since we have found that data distribution in feature space is related to the detection performance, it is possible to infer detection performance by comparing the features of the inference image with the data distribution of the training images. This approach can be leveraged to develop an out-of-distribution detector, which alerts for potential detection inaccuracies when encountering unfamiliar objects. In contrast, the current detector YOLOv8 tends to categorize unfamiliar objects as part of the background. Further detailed research into the factors influencing data distribution could be instrumental. This could guide the collection of training data to enhance the robustness and reliability of object detection performance.

5.1.2 Factors Related to Data Distribution Shifts

Additionally, we aim to provide more insights into the causes of varying data distributions and their associated impact on detection performance. We believe the data distribution shifts are largely influenced by differences in object appearances. In Section 4.5.1, we have noticed that the model trained on the RdGG 20211007 dataset and tested on the Philips best 105340 dataset has good detection performance of doctors and the display, while struggling with most other classes. The good detection performance can be attributed to the similarities in the wearings of the doctors in the two Cath Labs, as well as the high visual resemblance between the two displays, which appear to be the same equipment model. In contrast, other medical instruments, like the X-ray machines, have quite different appearances, which can result in different neural activation patterns inside the deep-learning-based object detector. In an ideal situation, we would have a large-scale image dataset of medical instruments of diverse equipment models, varying in factors such as viewpoint and lighting conditions. By training on such a diverse dataset, the object detector would learn to use the common features across different equipment models and achieve robust performance in various environments.

Furthermore, differences in viewpoints play a significant role in contributing to the divergent data distributions. As observed in the data visualization results presented in Section 4.5.1, data points are found to cluster narrowly around each camera viewpoint. The occurrence of such clustering within images from the same Cath Lab highlights the impact that viewpoint differences have on the data distribution. The differences in viewpoint also account for the divergent data distributions between the online images and the procedure recording datasets. The cameras within the Cath Labs are mounted in high positions, observing the procedures from angled downward views. In contrast, online images are predominantly taken by hand-held cameras with limited viewpoint variety for commercial purposes. While the online image dataset includes a wide array of medical instrument models, the images are captured from a relatively limited range of viewpoints. Consequently, even if the specific equipment model used in Cath Labs is represented in the training data, the object detector may still struggle to recognize the equipment from an unfamiliar viewpoint, especially when the viewpoint change is drastic. This issue underscores the importance of incorporating a multi-camera system.

Finally, the interactions between objects and their environments can significantly

impact both data distribution and detection performance. In instances where the detection was unsuccessful, we observed that certain instruments in the Cath Labs, such as the control panel button and the X-ray source, were covered with plastic films. The optical effects of these coverings, like reflections, cause the challenges in object detection. Occlusion is another factor that significantly impacts detection performance by diminishing the response of features within the object detector. This is particularly prevalent in Cath Labs during procedures, where medical staff often obstruct the view of various objects, leading to frequent occlusions. In contrast, the online images, typically captured outside of active procedures, exhibit fewer instances of occlusion compared to procedure recordings. Additionally, factors like lighting conditions may also influence data distribution. These factors, which are not covered in this thesis, suggest areas for more detailed and comprehensive future research. Such studies are essential for the robustness of vision-based systems, especially in dynamic and complex environments like Cath Labs.

5.2 Conclusion

This work emphasizes the generalization issues of deep-learning-based object detectors for the applications of workflow analysis inside Catheterization Laboratories (Cath Labs), where object detectors are required to deliver reliable detection results of people and medical instruments. However, the performance of the object detector significantly degrades when it is trained on procedure recordings from one Cath Lab and then deployed in a new, unseen Cath Lab. The implications of this performance degradation are exacerbated by the sensitive nature of medical videos, which are largely inaccessible outside the hospitals where clinical procedures occur. Consequently, our motivation is to ensure that object detectors function effectively in previously unseen Cath Labs.

In summary, the primary goal of this project was to identify the causes of performance degradation in unseen Cath Labs and to explore solutions to mitigate this issue. We demonstrated that such performance degradation is primarily due to differences in data distribution within the feature space of images. Furthermore, we showed that using diverse, non-sensitive online images as training data enables the object detector to generalize effectively to previously unseen Cath Labs. Additionally, the multi-camera systems, by switching to the most confident camera, can further improve object detection performance.

Our efforts focus on the data and camera systems aspects. We chose YOLOv8 as the object detector. The datasets are collected from one real clinical procedure recorded at Reinier de Graaf Hospital, two mock procedures recorded at Philips Best Campus, in addition to publicly available online images. The first experiment compared the detection performance when the training data is from the same Cath Lab, a different Cath Lab, or online images. The results suggest training data from the same Cath Lab can provide excellent detection performance, while the data from a different Cath Lab can cause large performance degradation. The model trained on the online images lies in the middle, providing moderately good performance. The data distribution visualization revealed that the reason for the performance degradation is data distribution shifts. Images from the same Cath Labs have narrow and similar data distribution,

while the ones from different Cath Labs are distinct. Online images have shifted but wider data distributions compared to procedure recordings. Then, we tested the object detector trained on online images. It can generalize to all three procedures with moderately good performance, but the performance of certain classes can be inconsistent across different Cath Labs. Data visualization results further showed that the poor performance stems from data distribution shifts at the object level. The final experiment explored the capability of the multi-camera systems when the detector is trained on online images. We created an aggregated version of object detection metrics, and showed that the multi-camera system can further improve detection performance by switching to suitable cameras.

This study formed a promising foundation for developing vision-based workflow analysis systems that are robust for deployment in unseen Cath Labs. It emphasized the importance of diverse training data and multi-camera systems in improving the generalization ability of object detectors in medical settings. However, its limitations, including the inconsistent performance of certain classes across different Cath Labs, highlight the need for further research.

5.3 Future Directions

5.3.1 Performance Improvement

1. **Data augmentation that creates synthetic occlusion and plastic film coverage:**
   The online images used in our training, primarily captured when no operations are being performed, exhibit significantly fewer instances of occlusion and plastic film coverage compared to actual procedure recordings. Introducing data augmentation techniques that mimic these real-world conditions in the online images could train the object detector to rely on features that are robust to both occlusion and plastic film coverage. There are already existing data augmentation techniques designed to introduce occlusion, as referenced in [71]. For simulating plastic film coverage, approaches can range from naive methods, such as overlaying a transparent image layer, to more sophisticated techniques. Employing these methods can help to bridge the gap between the online images and clinical procedure recordings, thereby enhancing the performance and reliability of the object detector.

2. **Modeling temporal information in the detection system:**
   In our research, we utilized the capabilities of multi-camera systems while treating each frame independently. However, integrating frame dependency through a tracking mechanism could significantly enhance the system's effectiveness. Our observations indicate that our object detector is sensitive to occlusion and viewpoint changes, leading to potential failures over certain time periods. Implementing a tracking system that maintains object identification across consecutive frames would allow the detector to momentarily fail without losing track of the objects. Given that the cameras in this project operate at a high frame rate of 25 FPS, incorporating a tracking mechanism is feasible from a hardware standpoint.

If we can jointly utilize the 3D spatial information provided by the multi-camera setup and the temporal information from video sequences, we can anticipate a considerable improvement in detection performance. This combined approach would offer a more comprehensive solution, enhancing the robustness and accuracy of object detection in dynamic environments like Cath Labs.

3. **Scaling up the online image dataset:**
   In this project, we have demonstrated that using online images as training data can lead to improved performance in unseen Cath Labs, primarily due to the high diversity these images offer. However, it's important to note that the scope of our image collection was limited in this research. We managed to collect and annotate only 800 images. As we expand our dataset with a greater number of diverse training images, we can anticipate a corresponding improvement in the detection performance.

5.3.2 Safety Measure

1. **Developing an out-of-distribution detector to alert for unfamiliar objects and bad performance:**
   In this thesis, we have successfully demonstrated that a model trained on online images can generalize to unseen Cath Labs. However, a notable observation is the inconsistency in the detection performance of certain classes across different Cath Labs. This issue becomes more concerning when considering the narrow and distinct data distributions characteristic of images from different Cath Labs.

   When the detector encounters an object situated in an unfamiliar region of the feature space, it would be more beneficial for it to provide a score indicating its level of familiarity with the object, rather than simply classifying it as part of the background. This mechanism could serve as an important safety measure, predicting the risk associated with the detector overlooking unfamiliar objects. This is particularly crucial in medical applications where high safety standards are essential.

   Implementing such an out-of-distribution detector is vital not only in new, unseen Cath Labs but also in managing variability within previously encountered Cath Labs. For instance, if unseen factors like uneven lighting lead to object detection failures in a Cath Lab that has been seen before, an out-of-distribution detector could provide a valuable warning.

   The concept of out-of-distribution detectors has already been explored in high-stake fields like autonomous driving, where they are used to alert for poor segmentation performance, as detailed in [72]. Employing similar methodologies in medical applications, particularly in complex environments like Cath Labs, could significantly enhance the safety and reliability of detection systems.

2. **Conducting object detection performance evaluation and feature analysis in more Cath Labs:**
   Despite having gained valuable insights from clinical procedure recordings in different Cath Labs, the scope of our study is limited by the number of Cath Labs

examined, which is currently only two. Expanding our research to include experiments in a greater number of Cath Labs would provide a more comprehensive understanding of the various factors that pose challenges to object detection in these environments. Additionally, by analyzing data from a wider array of Cath Labs, we can uncover more general trends in image data distribution.

Bibliography

[1] TBRHSC, *Getting to know the Cardiac Catheterization Lab*, en, Dec. 2023. [Online]. Available: https://tbrhsc.net/getting-to-know-the-cardiac-catheterization-lab/ (visited on 12/17/2023).

[2] *Cardiac Catheterization Lab Ventura County — Cardiac Catheterization Procedures*, en. [Online]. Available: https://www.mycmh.org/programs-services/heart-vascular/cardiac-catheterization-lab/ (visited on 12/16/2023).

[3] E. Picano, M. G. Andreassi, E. Piccaluga, A. Cremonesi, and G. Guagliumi, "Occupational risks of chronic low dose radiation exposure in cardiac catheterisation laboratory: The italian healthy cath lab study," *EMJ Int Cardiol*, vol. 1, no. 1, pp. 50–8, 2013.

[4] L. Venneri, F. Rossi, N. Botto, *et al.*, "Cancer risk from professional exposure in staff working in cardiac catheterization laboratory: Insights from the national research council's biological effects of ionizing radiation vii report," *American heart journal*, vol. 157, no. 1, pp. 118–124, 2009.

[5] M. Ozkaynak, K. Unertl, S. Johnson, J. Brixey, and S. N. Haque, "Clinical workflow analysis, process redesign, and quality improvement," in *Clinical informatics study guide: Text and review*, Springer, 2022, pp. 103–118.

[6] E. Rojas, J. Munoz-Gama, M. Sepúlveda, and D. Capurro, "Process mining in healthcare: A literature review," *Journal of biomedical informatics*, vol. 61, pp. 224–236, 2016.

[7] C. P. Nemeth, R. I. Cook, and D. D. Woods, "The messy details: Insights from the study of technical work in healthcare," *IEEE Transactions on Systems Man and Cybernetics- Part A Systems and Humans*, vol. 34, no. 6, pp. 689–692, 2004.

[8] T. A. Sanborn, J. E. Tcheng, H. V. Anderson, *et al.*, "Acc/aha/scai 2014 health policy statement on structured reporting for the cardiac catheterization laboratory: A report of the american college of cardiology clinical quality committee," *Circulation*, vol. 129, no. 24, pp. 2578–2609, 2014.

[9] R. S. Antunes, L. A. Seewald, V. F. Rodrigues, *et al.*, "A survey of sensors in healthcare workflow monitoring," *ACM Computing Surveys (CSUR)*, vol. 51, no. 2, pp. 1–37, 2018.

[10] M. Al-Faris, J. Chiverton, D. Ndzi, and A. I. Ahmed, "A review on computer vision-based methods for human action recognition," *Journal of imaging*, vol. 6, no. 6, p. 46, 2020.

[11] K. Chen, P. Gabriel, A. Alasfour, *et al.*, "Patient-specific pose estimation in clinical environments," *IEEE journal of translational engineering in health and medicine*, vol. 6, pp. 1–11, 2018.

[12] K. E. Weiss, M. Kolbe, Q. Lohmeyer, and M. Meboldt, "Measuring teamwork for training in healthcare using eye tracking and pose estimation," *Frontiers in Psychology*, vol. 14, p. 1 169 940, 2023.

[13] G. Diraco, A. Leone, and P. Siciliano, "An active vision system for fall detection and posture recognition in elderly healthcare," in *2010 Design, Automation &*

Test in Europe Conference & Exhibition (DATE 2010), IEEE, 2010, pp. 1536–1541.

[14] R. Kojcev, B. Fuerst, O. Zettinig, *et al.*, "Dual-robot ultrasound-guided needle placement: Closing the planning-imaging-action loop," *International journal of computer assisted radiology and surgery*, vol. 11, pp. 1173–1181, 2016.

[15] Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," *Proceedings of the IEEE*, 2023.

[16] L. Liu, W. Ouyang, X. Wang, *et al.*, "Deep learning for generic object detection: A survey," *International journal of computer vision*, vol. 128, pp. 261–318, 2020.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[18] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[19] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, 2015.

[20] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "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.

[21] D. Hoiem, S. K. Divvala, and J. H. Hays, "Pascal voc 2008 challenge," *World Literature Today*, vol. 24, no. 1, 2009.

[22] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*, Ieee, 2009, pp. 248–255.

[23] T.-Y. Lin, M. Maire, S. Belongie, *et al.*, "Microsoft coco: Common objects in context," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, Springer, 2014, pp. 740–755.

[24] Z. Zong, G. Song, and Y. Liu, "Detrs with collaborative hybrid assignments training," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 6748–6758.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[26] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[27] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.

[28] T. Diwan, G. Anirudh, and J. V. Tembhurne, "Object detection using yolo: Challenges, architectural successors, datasets and applications," *multimedia Tools and Applications*, vol. 82, no. 6, pp. 9243–9275, 2023.

[29] W. Liu, D. Anguelov, D. Erhan, *et al.*, "Ssd: Single shot multibox detector," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, Springer, 2016, pp. 21–37.

[30] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[31] J. Redmon and A. Farhadi, "Yolo9000: Better, faster, stronger," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7263–7271.

[32] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[33] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.

[34] G. Jocher, A. Chaurasia, and J. Qiu, *YOLO by Ultralytics*, version 8.0.0, Jan. 2023. [Online]. Available: https://github.com/ultralytics/ultralytics.

[35] K. Zhou, Z. Liu, Y. Qiao, T. Xiang, and C. C. Loy, "Domain generalization: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[36] J. Quinonero-Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence, *Dataset shift in machine learning*. Mit Press, 2008.

[37] W. M. Kouw and M. Loog, "An introduction to domain adaptation and transfer learning," *arXiv preprint arXiv:1812.11806*, 2018.

[38] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part IV 11*, Springer, 2010, pp. 213–226.

[39] Y.-H. Chen, W.-Y. Chen, Y.-T. Chen, B.-C. Tsai, Y.-C. Frank Wang, and M. Sun, "No more discrimination: Cross city adaptation of road scene segmenters," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 1992–2001.

[40] X. Liu and Y. Yuan, "A source-free domain adaptive polyp detection framework with style diversification flow," *IEEE Transactions on Medical Imaging*, vol. 41, no. 7, pp. 1897–1908, 2022.

[41] A. Torralba and A. A. Efros, "Unbiased look at dataset bias," in *CVPR 2011*, IEEE, 2011, pp. 1521–1528.

[42] Z. Shen, J. Liu, Y. He, *et al.*, "Towards out-of-distribution generalization: A survey," *arXiv preprint arXiv:2108.13624*, 2021.

[43] X. Zhang, Z. Xu, R. Xu, *et al.*, "Towards domain generalization in object detection," *arXiv preprint arXiv:2203.14387*, 2022.

[44] Q. Fan, M. Segu, Y.-W. Tai, *et al.*, "Towards robust object detection invariant to real-world domain shifts," in *The Eleventh International Conference on Learning Representations*, 2022.

[45] H.-M. Hsu, T.-W. Huang, G. Wang, J. Cai, Z. Lei, and J.-N. Hwang, "Multi-camera tracking of vehicles based on deep features re-id and trajectory-based camera link models.," in *CVPR workshops*, 2019, pp. 416–424.

[46] M. Rampinelli, V. B. Covre, F. M. De Queiroz, R. F. Vassallo, T. F. Bastos-Filho, and M. Mazo, "An intelligent space for mobile robot localization using a multi-camera system," *Sensors*, vol. 14, no. 8, pp. 15 039–15 064, 2014.

[47] U. M. Erdem and S. Sclaroff, "Automated camera layout to satisfy task-specific and floor plan-specific coverage requirements," *Computer Vision and Image Understanding*, vol. 103, no. 3, pp. 156–169, 2006.

[48] S. Ezatzadeh, M. R. Keyvanpour, and S. V. Shojaedini, "A human fall detection framework based on multi-camera fusion," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 34, no. 6, pp. 905–924, 2022.

[49] M. Taufeeque, S. Koita, N. Spicher, and T. M. Deserno, "Multi-camera, multi-person, and real-time fall detection using long short term memory," in *Medical Imaging 2021: Imaging Informatics for Healthcare, Research, and Applications*, SPIE, vol. 11601, 2021, pp. 35–42.

[50] V. Belagiannis, X. Wang, H. B. B. Shitrit, *et al.*, "Parsing human skeletons in an operating room," *Machine Vision and Applications*, vol. 27, pp. 1035–1046, 2016.

[51] E. J. González-Galván, S. R. Cruz-Ramırez, M. J. Seelinger, and J. J. Cervantes-Sánchez, "An efficient multi-camera, multi-target scheme for the three-dimensional control of robots using uncalibrated vision," *Robotics and Computer-Integrated Manufacturing*, vol. 19, no. 5, pp. 387–400, 2003.

[52] W. Xu and A. Huang, "Multi-camera operating room activity analysis for workflow analysis," in *Medical Imaging 2022: Imaging Informatics for Healthcare, Research, and Applications*, SPIE, vol. 12037, 2022, pp. 72–76.

[53] A. Coates and A. Y. Ng, "Multi-camera object detection for robotics," in *2010 IEEE International Conference on Robotics and Automation*, IEEE, 2010, pp. 412–419.

[54] R.-Y. Ju and W. Cai, "Fracture detection in pediatric wrist trauma x-ray images using yolov8 algorithm," *arXiv preprint arXiv:2304.05071*, 2023.

[55] C.-Y. Wang, H.-Y. M. Liao, and I.-H. Yeh, "Designing network design strategies through gradient path analysis," *arXiv preprint arXiv:2211.04800*, 2022.

[56] Z. Zhang, "Drone-yolo: An efficient neural network method for target detection in drone images," *Drones*, vol. 7, no. 8, 2023.

[57] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8759–8768.

[58] H. Law and J. Deng, "Cornernet: Detecting objects as paired keypoints," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 734–750.

[59] B. Cheng, Y. Wei, R. Feris, *et al.*, "Decoupled classification refinement: Hard false positive suppression for object detection," *arXiv preprint arXiv:1810.04002*, 2018.

[60] Z. Zheng, P. Wang, D. Ren, *et al.*, "Enhancing geometric factors in model learning and inference for object detection and instance segmentation," *IEEE transactions on cybernetics*, vol. 52, no. 8, pp. 8574–8586, 2021.

[61] X. Li, W. Wang, L. Wu, *et al.*, "Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection," *Advances in Neural Information Processing Systems*, vol. 33, pp. 21 002–21 012, 2020.

[62] Z. Wei, C. Duan, X. Song, Y. Tian, and H. Wang, "Amrnet: Chips augmentation in aerial images object detection," *arXiv preprint arXiv:2009.07168*, 2020.

[63] M. D. Awheda and H. M. Schwartz, "Exponential moving average q-learning algorithm," in *2013 IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning (ADPRL)*, IEEE, 2013, pp. 31–38.

[64] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*, pmlr, 2015, pp. 448–456.

[65] P. Molchanov, S. Tyree, T. Karras, T. Aila, and J. Kautz, "Pruning convolutional neural networks for resource efficient transfer learning," *arXiv preprint arXiv:1611.06440*, vol. 3, 2016.

[66] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction. arxiv 2018," *arXiv preprint arXiv:1802.03426*, 1802.

[67] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne.," *Journal of machine learning research*, vol. 9, no. 11, 2008.

[68] T. T. Cai and R. Ma, "Theoretical foundations of t-sne for visualizing high-dimensional clustered data," *The Journal of Machine Learning Research*, vol. 23, no. 1, pp. 13 581–13 634, 2022.

[69] CVAT.ai Corporation, *Computer Vision Annotation Tool (CVAT)*, version 2.8.2, Nov. 2023. [Online]. Available: https://github.com/opencv/cvat.

[70] R. Padilla, S. L. Netto, and E. A. Da Silva, "A survey on performance metrics for object-detection algorithms," in *2020 international conference on systems, signals and image processing (IWSSIP)*, IEEE, 2020, pp. 237–242.

[71] A. Wang, Y. Sun, A. Kortylewski, and A. L. Yuille, "Robust object detection under occlusion with context-aware compositionalnets," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 12 645–12 654.

[72] P. Oberdiek, M. Rottmann, and G. A. Fink, "Detection and retrieval of out-of-distribution objects in semantic segmentation. 2020 ieee," in *CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2020, pp. 1331–1340.