Comparative Analysis of YOLO Variants for Object Detection in Thermal Images

MSc Research Project
Msc Data Analytics

Shajo Varghese
Student ID: 22115943

School of Computing
National College of Ireland

Supervisor: Bharat Agarwal

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Shajo Varghese
Student ID:	22115943
Programme:	Msc Data Analytics
Year:	2023
Module:	MSc Research Project
Supervisor:	Bharath Agarwal
Submission Due Date:	14/12/2023
Project Title:	Comparative Analysis of YOLO Variants for Object Detection in Thermal Images
Word Count:	5608
Page Count:	18

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Shajo Varghese Eramplackal
Date:	14th December 2023

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Comparative Analysis of YOLO Variants for Object Detection in Thermal Images

Shajo Varghese
22115943

Abstract

This study delves into the application of the state-of-the-art object detection models inside the YOLO (You Only Look Once) family,with a special focus on YOLOv5, YOLOv7, and YOLOv8, in the context of thermal imaging for autonomous vehicles and which can also be extended to security applications. Focusing on the limitations of traditional vision systems under challenging and extreme conditions such as low visibility and extreme weather conditions, it advocates for thermal imaging as a viable alternative This approach seeks to overcome the weaknesses of Camera sensors,RADAR (Radio Detection and Ranging) and LIDAR(Light Detection and Ranging). The study involves the acquisition and preprocessing of a diverse set of thermal images (TIR - Thermal Infrared), consisting of various real-world scenarios to ensure compatibility with the YOLO architecture. A detailed comparative analysis of YOLOv5, YOLOv7, and YOLOv8 is conducted, focusing on evaluation metrics like Precision, Recall, and mAP (Mean Average Precision). This analysis aims to capture the model's performance in the thermal imaging domain for detecting car and humans under real-world conditions. YOLOv5 and YOLOv8 had similar mAP, but the latter had slightly higher precision and recall, making it the best performing model in this study. The findings from this research will contribute significantly to autonomous driving systems and security surveillance,offering valuable insights for the selection of appropriate object detection model considering application-specific requirements.

1 Introduction

In the realm of autonomous vehicles, surveillance and security, the ability to detect vehicles and other objects under extreme weather conditions, such as low visibility and adverse weather, is of paramount importance. Accurate perception of surrounding environment still remains a critical challenge Balasubramaniam and Pasricha (2022). However, computer vision has witnessed significant advancements in recent years. Lightweight sensors are used to perceive objects around the vehicles like Camera Sensors, RADAR Sensors, etc.Kurfiss and Jackson (2023) argues that Camera Sensors are less reliable in extreme weather conditions and Radar sensors have limited vision of range. Thermal Imaging data has emerged as one of the best sensors to be incorporated in such scenarios. Thermal imaging offers a unique perspective, enabling the detection and re-cognition of objects in challenging conditions, such as low visibility and adverse weather. Thermal cameras detect heat emitted by objects and convert it into a visual representation and they are not vulnerable to extreme weather conditions. Miethig et al. (2019) strongly supports the integration of Thermal imaging sensors into autonomous driving systems. According to the research by Rivera Velázquez et al. (2022), current vision systems that integrate visible imaging, LIDAR, and Radar technologies face challenges when operating in tough environmental conditions. They also suggest that thermal imaging offers a potential solution, as it has the ability to perceive and understand the surroundings even in extremely foggy situations.

The application of deep learning models in identifying and locating vehicles and other objects has shown promising results Krišto et al. (2020). YOLO algorithms are known for their real time object detection capabilities, which makes them an attractive candidate for tasks involving thermal images Lianqiao et al. (2019). Rivera Velázquez et al. (2022) have highlighted the challenges faced by existing systems and suggested that thermal imaging data emerges as a promising solution. Their research highlighted the challenges faced by the existing systems and advocates the use of the thermal images as a promising solution. Their research demonstrates the efficiency of a UAV object detection framework based on YOLO models in detecting Thermal Infrared (TIR) images and videos, particularly in scenarios with multiple objects. This project aims to explore the capabilities of state-of-the-art object detection models within the YOLO (You Only Look Once) family. The primary focus of this project lies in enhancing the accuracy and robustness of car and human detection in thermal images under challenging real-world conditions. On this end, we conduct a comprehensive comparative analysis of three distinct YOLO variants: YOLOv5, YOLOv7, and YOLOv8. In this study, we first acquire a diverse dataset of thermal images capturing various real-world scenarios involving vehicles. These thermal images are pre-processed to ensure compatibility with the YOLO architecture, followed by rigorous training and fine-tuning of each YOLO version. To evaluate their performance, we conduct a range of experiments encompassing various metrics like Precision, Recall and Mean Average precision(mAP).

The aim of this study is to evaluate the performance of these YOLO variants when experimented with the domain of thermal imaging, which poses unique challenges when compared with traditional visual object detection. We seek to answer critical questions about the suitability of different YOLO versions for car and human detection in thermal images and their compatibility with hardware constraints. The insights gained from this comparative analysis will inform decision making for choosing the most appropriate model for specific applications, considering both efficiency and resource constraints.

2 Research Question

How do the YOLO variants compare in terms of computational resource requirements and efficiency for thermal image object detection?

3 Motivation

The purpose of conducting a comparative analysis serves several practical uses in real-world applications. Many individuals employ comparative analysis to make informed choices among multiple machine learning models. It becomes crucial to strike a balance between model accuracy and efficiency in applications involving edge devices and hardware with constrained memory and processing power. The field of thermal imaging can also add additional complexity due to the nature of the data and the specific challenges

it represents.
Consider a scenario where we require object detection on an end device, such as hardware. We have a variety of advanced and highly accurate models at our disposal. However, these algorithms are not lightweight and can consume memory unpredictably. Using the YOLOv8 model on a Raspberry Pi, for instance, may be challenging due to limited RAM capacity, unlike previous YOLO variants that integrated seamlessly with this hardware. If, in our specific task, the earlier YOLO variants provide superior results, there is no need to invest time and resources in attempting to use YOLOv8. In real-time settings, many organizations face difficulties in adopting advanced ML techniques due to such constraints.
While the evolution of YOLO models, from YOLOv5 to YOLOv8, represents a progression towards increasingly sophisticated architectures, the study by Shokri et al. (2023) sheds light on a crucial aspect: higher YOLO versions may not always outperform their predecessors. YOLOv7 outperformed YOLOv8 in this study. This emphasizes the indispensability of conducting comparative analyses to discern the nuances of each model's performance in diverse environments. Such analyses help in making informed decisions regarding model selection, striking a balance between accuracy and computational efficiency. The findings of this study underscore the significance of scrutinizing various YOLO variants, as they hold the key to optimizing vehicle detection in real-time applications for the ever-expanding domain of intelligent transportation systems.
In this comparative analysis, by assessing these models against our dataset, we can swiftly determine whether it is necessary to opt for large and resource-intensive models to improve accuracy or if there are alternative models that can yield superior results with our dataset.

4 Literature Review

Object detection in thermal images forms a critical component of numerous applications, particularly in autonomous navigation, security and surveillance. As the demand for such robust and efficient object detection systems continues to rise, the You Only Look Once algorithms have managed to gain significant attention because of their real time processing capabilities. These algorithms come in different versions and variants such as Tiny-YOLO and Complex-YOLO. However, there is still insufficient research, hence a growing need to understand how these Variants perform in the specific context of thermal image object detection. This section aims to explore and study the existing research focused on comparing the efficiency of YOLO variants concerning Thermal image object detection. The paper proposed by Balasubramaniam and Pasricha (2022) explores the role of object detection in computer vision , advocating its significance in a diverse range of applications , including autonomous vehicles. In perception systems, object detectors play a very important role in identifying and localizing objects in real- time. The article also addresses the challenges and opportunities that exist while integrating object detectors into AVs. Some of the noteworthy points stressed by the authors include using Neural Architecture Search (NAS) for optimizing object detection models, the need for real time processing to reduce latency in video-based detection. The paper discusses the challenges faced by supervised learning methods, incorporation of time-series information and advocates using semi supervised approaches. The paper also stresses the importance of open datasets that includes various weather and environmental conditions, which would in turn provide a more robust training ground for object detectors. Resource constraints problems have also been addressed, highlighting efforts to reduce computational power using pruning, quantization and knowledge distillation. The paper stresses that all these challenges must be overcome to establish safe and reliable AV transportation models. The paper proposed by Rivera Velázquez et al. (2022) states the limitations of commonly employed sensors like Camera, LIDAR and Thermal. Camera and Lidar sensors perform very poorly in adverse weather conditions. Although Radar sensors perform good in extreme weather conditions, obstacle edge detection is very difficult using them. The paper also discusses calibrating sensors before using them also remains an arduous task, since wireless networks face difficulties in achieving reliable calibrations. Environmental impact on sensor functionality, especially in extreme weather conditions still poses a major challenge. Sensors like Radar, may experience decreased efficacy in challenging weather conditions.

In light of these challenges, Thermal imaging have emerged as the best alternative. Thermal imaging captures data provides temperature related insights. Additionally, it is immune to colour information absence and coarse resolution, which generally affects LIDAR and Radar, positioning thermal imaging as a robust solution.The paper proposed by Ignatious et al. (2022) advocates the use of thermal imaging as a promising solution, which is extremely competent in perceiving environment , even in extremely foggy conditions. After conducting various experiments, they concluded that (Angle of View) AOV is one of the primary factors for ensuring effective object detection. Thermal cameras with AOV of 18° and 30° were found to be most suitable, achieving a detection rate for more than 90 percent under thick fog conditions. Their research also proved that thermal images are compatible with various detection algorithms based on neural networks, further strengthening the case for incorporating thermal sensors in object detection systems. The authors also emphasized the need for further testing and evaluation of such systems in diverse real world scenarios consisting static/dynamic objects for gaining reliable conclusions of the sensor's resilience in extremely foggy conditions. Thermal sensors are also the optimum choice for sensor fusion. Miethig et al. (2019) strongly supports the use of thermal imaging as supplementary sensor in autonomous driving systems. It states that it can offer key information about the vehicle's environment in challenging conditions where other sensors fail to operate at optimum level.

Along with choosing appropriate sensor, choosing the object detection model that gives the best performance is also of paramount importance. Krišto et al. (2020) investigates the performance of the state-of-the-art object detection models, which includes R-CNN, Faster R-CNN, SSD, Cascade R-CNN, and YOLOv3 on thermal images. A custom dataset consisting of images of various weather conditions was used. The study dealt with issues associated with detecting persons engaging in normal evasive scenarios, especially in scenarios simulating illegal border crossings. Preliminary experiments using different object detectors such as Faster R-CNN, SSD, FCOS, Cascade R-CNN, and YOLOv3, revealed that while R-CNN, Cascade R-CNN, and YOLOv3 achieve comparable detection results in thermal images, YOLOv3 had significantly higher processing speed, which prompted authors to choose YOLOv3 for further experiments. The study established a baseline using an original YOLO-V3 model trained on the COCO RGB dataset. However, it performed sub-optimally on thermal images, prompting authors to carry out additional training specifically for thermal images. The model was trained on approximately 3000 thermal images and demonstrated substantial improvement, achieving an average precision score of 97.93 % for all weather conditions. There was a stark difference between the original YOLOv3 (trained on COCO RGB) with the model specifically trained on the thermal dataset, the latter outperformed the formal, indicating the need for thermal-specific training for optimal results in thermal object detection. The study also demonstrates YOLOv3's model's good generalization properties, showcasing reliable results when tested on external image sets. Lianqiao et al. (2019) investigates the application of YOLO neural network for thermal image recognition in power facilities, with the primary emphasis on dataset construction specific to power equipment and preprocessing infrared images to reduce photo noise. Bounding boxes were employed to identify and name electrical equipment in infrared thermal images. The effectiveness of the system was tested on various power devices like Porcelain Sleeve, Isolation Switch Combine Filter. The results indicated that the system achieved accurate and stable identification of power facilities. The least squares curve method was successful in precisely locating the highest temperature of each device. The paper also highlights the significance of YOLO model by comparing it with the traditional neural network R-CNN. Compared to R-CNN, proposed system based on YOLO demonstrated 80 % more recognition accuracy and 48.08 % improvement in recognition speed at a similar correct rate.

The YOLO models promises high speed and accuracy in object detection systems at the cost of high computational power. There are numerous YOLO variants available, each with its strength and weaknesses. Hence, it becomes important to compare the performance of different YOLO variants in order to select the best model suitable to our application. Suppose a company develops a camera trap system to monitor wildlife in remote areas. These traps are powered by solar energy and have limited battery life. The company wants to employ an object detection algorithm that can run on the camera traps' low-power CPUs to identify and track wildlife. The company initially tried to use the YOLOv8 for object detection but was found to be too computationally demanding. The camera traps used to run of battery quickly if YOLOv8 was used. The company then experimented with YOLOv5 and YOLOv7 and found that both models were able to run on the camera traps' CPUs without draining the batteries. This leads to lower versions of YOLO used in edge devices attain similar object detection capabilities at lower computational costs. Olorunshola et al. (2023) conducts a comparative analysis between YOLOv5 and YOLOv7 for object detection in field of Remote Weapon Station imagery. The experimentation involved training custom models with both YOLOv5 and YOLOv7 independently, with a dataset that consisted of 9,779 images with 21,561 annotations across four classes: Persons, Handguns, Rifles, and Knives. The comparative analysis evaluated precision, recall, mAP@0.5, and mAP@0.5:0.95 for both YOLOv5 and YOLOv7.It was found that YOLOv5 consistently outperformed YOLOv7 in terms of precision, achieving higher scores across all classes. YOLOv7 demonstrates a slightly higher recall but YOLOv5 excels in accuracy, with an overall improvement of 4.0 %.In terms of mAP@0.5 and mAP@0.5:0.95, YOLOv5 provided better results than YOLOv7, recording a 4 % difference in mAP@0.5 and a 2.7 % difference in mAP@0.5:0.95, indicating its higher accuracy and better performance in object detection.

Shokri et al. (2023)conducted experiments on a nine challenging and diverse datasets covering various environmental conditions, camera angles and vehicle types. The evaluation metrics used were vehicle detection rate, computation time, classification and localization. In their research, they found out that YOLO versions, particularly YOLOv7, demonstrated superior performance in both, vehicle detection and localization. While RCNN and SSD exhibited weaker performance, YOLO versions outperformed them and YOLOv7 has shown better detection and classification among all YOLO versions (v8,v7,v5 and v5).

YOLOv7 also registered lower response in computational time making it the best performing model. It displayed commendable adaptability to nighttime conditions, achieving an accuracy exceeding 99 percent. It achieved high accuracy without additional training data and demonstrated real-time capabilities, particularly when using GPU and RAM in addition to a CPU processor. The paper introduces a novel approach for comparing vehicle detection algorithms using challenging highway video datasets, thus addressing the limitations in previous studies. The proposed method relies on the primary model of each algorithm and eliminates the need for extra training data, unlike some previous works that customizes weights or used additional training data. The authors also suggest further evaluation of deep learners using UAVs to cover larger road areas and emphasize the need for improved accuracy in classification of heavy vehicles and more accurate localization.Gašparović et al. (2023) also focuses on comparing various YOLO versions , but in harsh underwater environments using a dataset collected by a Remote operated vehicle (ROV). The YOLO versions evaluated in this paper were YOLOv5, YOLOv6, YOLOv7. The aim of the study was to determine the superiority of newer YOLO version over the older ones in context of object detection performance, specifically in challenging underwater conditions. Among them, YOLOv5 achieved the highest mean Average Precision (mAP) score, followed by YOLOv7 and YOLOv6. The mAP metric results showed that YOLOv5 and YOLOv7 were most effective versions for detecting objects in underwater conditions on their dataset, with YOLOv8 and YOLOv6 also showing excellent performance. The difference in mAP between different YOLO versions were also relatively small, indicating high levels of object detection accuracy for all YOLO versions. The Precision-recall curves for YOLOv5 and YOLOv7 showed high precision and recall values with the former having a slightly higher recall for their three target classes. The F1 confidence curves indicated that YOLOv7 and YOLOv5 are the most accurate detectors for object detection in harsh underwater conditions.

5 Methodology

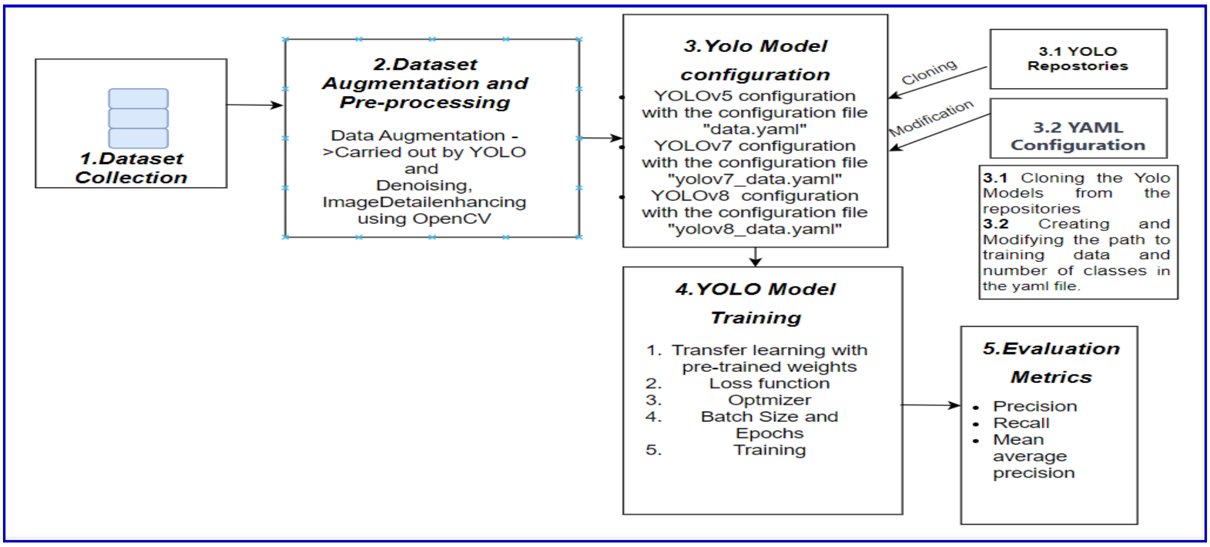

Figure 1: Model Architecture

5.1 Data collection

A labelled dataset was collected from the universe-roboflow dataset[1] with bounding box annotations for the objects, car, and person. The dataset had 10037 images for training and 2237 for validation. Some pre-processing steps were already applied, like resizing (stretched to 640x480), grayscale, and filter null, which require at least 66 percent of the images to be annotated. The folder structure of the dataset is visible in Figure 2. The images folder has images of persons and car objects, and the labels file has details of the location of the objects and the class of the objects. There are two classes specified in the dataset, i.e., person and car.

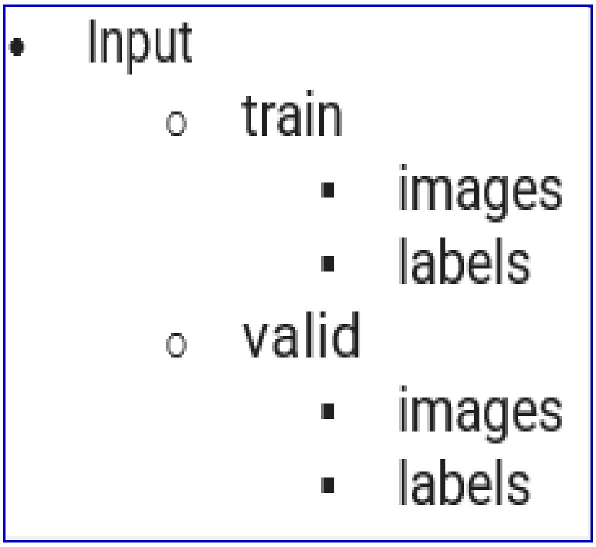

Figure 2: Dataset Structure

1

5.2 Data Augmentation and pre-processing

Data augmentation is an essential technique in machine learning, especially for deep neural networks for tasks such as image classification and object detection Shorten and Khoshgoftaar (2019). The YOLO architecture efficiently handles the augmentation process, employing various techniques like albumentations, including Blur (with a probability of 0.01 and a blur limit between 3 and 7), MedianBlur (with a probability of 0.01 and a blur limit between 3 and 7), ToGray (with a probability of 0.01), and CLAHE (with a probability of 0.01, a clip limit between 1 and 4.0, and a tile grid size of 8x8). Various preprocessing techniques will also be applied using the OpenCv library in Python.

5.3 Model configuration

This step consists of the configuration of all the YOLO models. To initiate the use of YOLO models, we start by cloning the relevant GitHub repository associated with the YOLO version, in this instance, YOLOv5, YOLOv7, and YOLOv8.These repositories are equipped with the essential configuration files necessary for training the YOLO Model.

[1]https://universe.roboflow.com/thermal-imaging-0hwfw/flir-data-set/dataset/13

The act of cloning the repository ensures that you have access to the model, including all the requisite files and configurations tailored for efficient training. By leveraging these configuration files, users can seamlessly tailor the YOLOv5 model to their specific datasets and objectives.

Configuration File: Creating or modifying an existing YAML configuration file ('yaml' file) to specify the dataset paths and targets.

5.4 Model Training

5.4.1 Transfer Learning

The YOLO models will be initialized with weights pretrained on a large dataset (e.g., COCO). This helps the model converge faster and achieve better performance.

5.4.2 Loss Function

YOLO models use a combination of localization loss, confidence loss, and class loss. The default loss function is generally suitable.

5.4.3 Optimizer

The common optimizer used includes stochastic gradient descent (SGD) or Adam.

5.4.4 Batch Size and Epochs

The YOLO model training commences by gathering fundamental user inputs, including the number of epochs, image resolution, batch size, as well as the paths to the model weights and data.yaml file. Once these essential data is provided, the model loads and validates the data, initiating the training process. Throughout the training, the model automatically saves necessary graphs and the trained models. This streamlined approach ensures an efficient and user-friendly experience in configuring, training, and saving YOLO models.

5.5 Evaluation

Metrics: Evaluate the model's performance using metrics such as precision, recall, and mAP (mean Average Precision) on a validation set. This helps us to understand how well the model will generalize to unseen data.

6 Implementation

6.1 Pre-processing

A series of pre-processing steps were applied to the dataset in order to enhance the quality and information content of the input images. The aim is to prepare the data for optimal performance during the subsequent stages of training. The following pre-processing techniques were employed:

6.1.1 Data Loading

The process is initiated by loading the raw images from the dataset using the DataLoading function. The OpenCV library is used to read the images from the file path.

6.1.2 Denoising

A denoising technique is applied using the fastNlMeansDenoisingColored () function from OpenCV. It is carried out to reduce noise and artifacts in the images and is crucial for improving the overall clarity of images.

6.1.3 Image Smoothening

Smoothening is carried out through Gaussian blur using the GaussianBlur () function. It was carried out to reduce high-frequency noise and detail in images.

6.1.4 Image Sharpening

Images are sharpened by applying a sharpening kernel using the filter2D function to enhance the edges and fine details in the images.

6.1.5 Image Detail Enhancement

The detailEnhance () function is used to carry out Detail enhancement, adjusting the spatial and tonal factors.

6.1.6 Image Texture Enhancement

Texture enhancement is accomplished through applying a convolutional filter using the filter2D function contributing to improved feature representation.

6.1.7 Image Saving

In the last step of pre-processing, the images are saved to the specified output path using the imwrite () function. The final images are saved in the "preprocessed" folder with the same structure as that of input file. The labels are manually uploaded since they do not require any pre-processing.

6.2 YOLO V5 Training and configuration

6.2.1 Environmental Setup

YOLOv5 was implemented in the Google Colab environment. The YOLOv5 repository was cloned from the Ultralytics GitHub, and the project directory was mounted from Google drive.The configuration file for YOLOv5 is "data.yaml".

6.2.2 Model initialization

The architecture selected was YOLOv5s. Pre-trained weights were loaded from the official repository to expedite convergence during training. The **torch.hub.load()** function was used to load the YOLOv5 model from the Github repository.

6.2.3 Training configuration

The YOLOv5 model was trained for 5 epochs with the specific parameters.

- Image size: 640x640 pixels
- Batch size: 32
- Learning rate: 0.01 (SGD optimizer)

6.2.4 Model Architecture

The YOLOv5 model consists of the following layers:

- Input Layer: 3 channels (RGB), followed by a convolutional layer.
- Backbone: Several convolutional layers organized in the form of CSPDarknet53.
- Feature Pyramid Network (FPN): Includes downsample and upsample operations.
- Detection Head: Consists of additional convolutional and detection layers.

6.2.5 Model Summary

- Total Layers: 214
- Total Parameters: 7,025,023
- Total Gradients: 7,025,023
- Total GFLOPs: 16.0

Specifics of Some Layers:

- Convolutional Layers: Utilize different kernel sizes and strides for feature extraction.
- C3 Layers: Components of CSPDarknet53 for improved feature representation.
- SPPF Layer: Spatial Pyramid Pooling with a focus on reducing computational complexity.
- Detect Layer: Final layer responsible for object detection, with anchor boxes defined.

6.3 YOLOv7 Training and configuration

6.3.1 Environmental setup

YOLOv7was implemented in the Google Colab environment. The YOLOv7 repository was cloned from the github.com/WongKinYiu, and the project directory was mounted from Google drive. The configuration file for YOLOv7 is "YOLOv7_data.yaml".

6.3.2 Model initialization

The latest YOLOv7.pt file was downloaded from the GitHub path and the model was trained on a subset of 5050 training images and corresponding label images and the entire validation images and labels.

6.3.3 Training configuration

The YOLOv7 model was trained for 20 epochs with the specific parameters.

- Image size: 640x640 pixels
- Batch size: 16

6.3.4 Model Summary

- Total layers: 407
- Total parameters :37200095
- Total gradients: 37200095

6.4 YOLOv8 Training and configuration

6.4.1 Environmental setup

YOLOv8 was implemented in the Google Colab environment. The YOLOv8 repository was cloned from the Ultralytics GitHub, and the project directory was mounted from Google drive. The configuration file for YOLOv8 is "YOLOv8_data.yaml".

6.4.2 Model initialization

The architecture selected was YOLOv8s.

6.4.3 Training configuration

The YOLOv8 models was trained for 5 epochs with the specific parameters:

- Input Image Size: 640x640 pixels
- Batch Size: 16
- Optimizer: AdamW(lr=0.001667)

Data Augmentation carried out by YOLOv8:

- Horizontal Flip: 50
- Mosaic: 100
- Mixup: 0
- Copy-Paste: 0
- Other augmentations: Blur, MedianBlur, ToGray, CLAHE

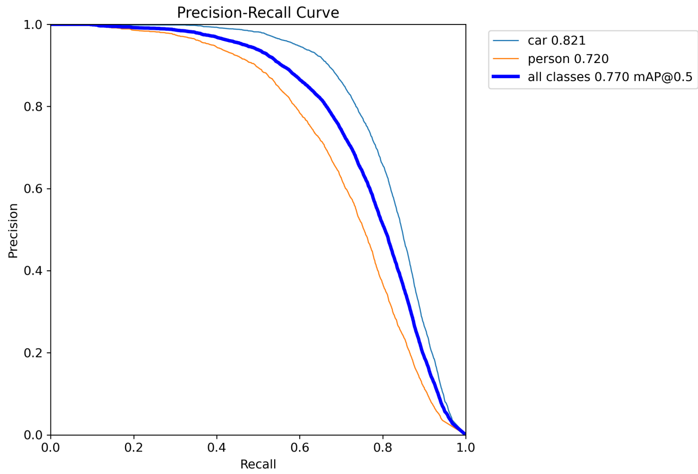

Figure 3: Precision Recall Curve for YOLOv5

7 Evaluation

This section aims to evaluate the performance of YOLOv5, YOLOv7 and YOLOv8 models and analyse various graphs associated with evaluation metrics. Figure 3 shows the Precision-Recall (PR) curve for the YOLOv5 model. The PR curve shows the trade-off between precision and recall for different confidence thresholds. The PR curve for the YOLOv5 model shows that the model achieves a high precision (over 0.7) for a wide range of recall values (from 0.2 to 0.8). This means that the model is very good at predicting objects without predicting too many false positives. The PR curve also shows that the model achieves a high recall (over 0.8) for a wide range of precision values (from 0.5 to 0.7). This means that the model is very good at finding all of the objects in an image without missing too many true positives. Overall, the PR curve shows that the YOLOv5 model is performing very well on the task of car and human detection . Figure 4 shows

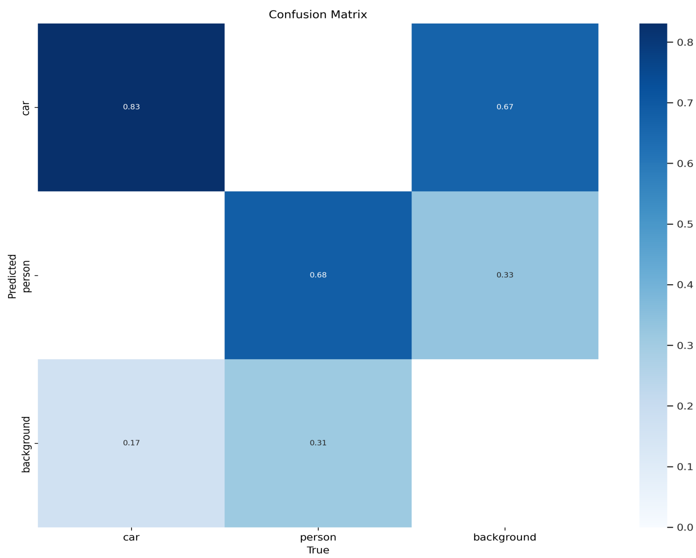

Figure 4: Confusion Matrix for YOLOv5

the confusion matrix for the YOLOv5 model while its attempting to classify cars, people. The rows are the true labels of the objects and the columns represent the predicted labels of the models. YOLOv5 has correctly classified 83 percent of the cars in the dataset and 68 percent of the persons in the dataset.Overall, the confusion matrix shows that the model is performing well on the car and person classes.

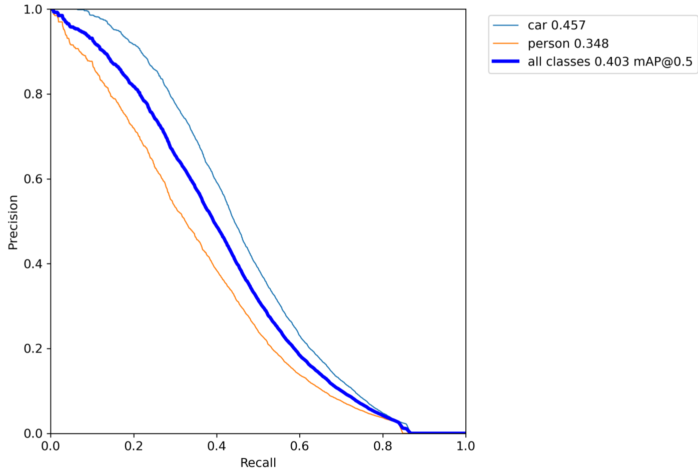

Figure 5: Precision Recall Curve for YOLOv7

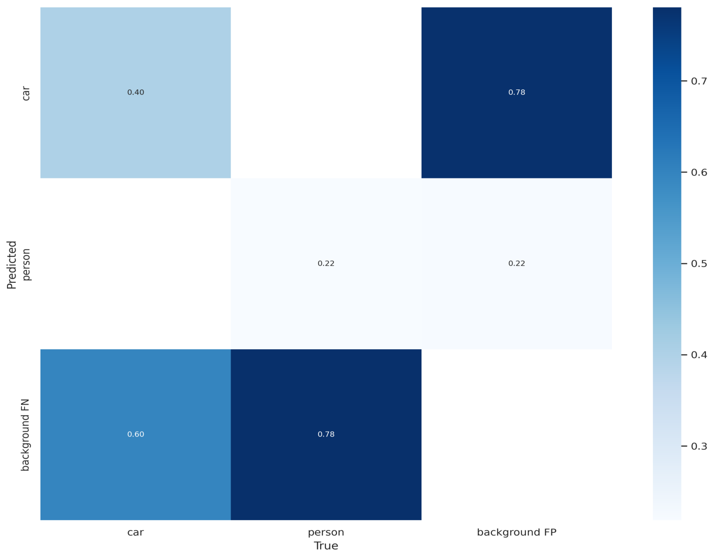

Figure 6: Confusion Matrix for YOLOv7

Figure 5 shows the Precision-Recall curve for YOLOv7. From the image, it is clearly visible that there is less area under the precision recall curve, which indicates lower mean average precision. An overall mAP of 0.4 indicates that the YOLOv7 model is showcasing moderate performance in object detection . YOLOv7 had mAP of 0.457 in detecting car instance and 0.348 in detecting human instances respectively. It indicates that the YOLOv7 is able to identify and localize objects to some extent,but there is room for improvement.Figure 6 reveals the confusion matrix for YOLOv7. The model was able to correctly classify approximately 40 percent of the car instances correctly. But , it was only able to classify 22 percent of human instances correctly.

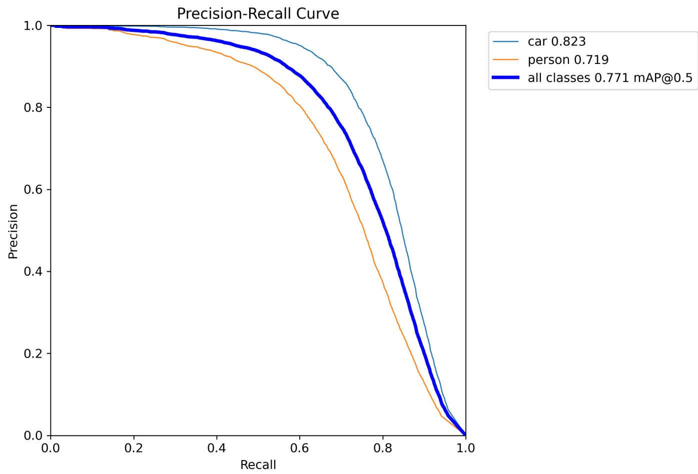

Figure 7: Precision Recall Curve for YOLOv8

Figure 7 depicts the precision recall curve for the YOLOv8. YOLOv8 has a similar performance of that of YOLOv5 with the same mean average precision for human instances . However, it shows a little higher mAP in detecting car instances which lead to an overall higher mAP than YOLOv5. Hence, among all the YOLO models, YOLOv8 models had a higher mAP which means that it would be a promising choice when mAP is the primary evaluation metric. The slightly higher mAP which can be observed in the above comparative analysis suggests a refinement in models capability to detect and classify cars and persons in thermal images. It can be attributed to different optimization strategies and architectural enhancements introduced in the YOLOv8. While the significance in mAP may seem subtle, even slight improvements are noteworthy in the field of object detection. The enhanced mAP in YOLOv8 model signifies an advancement in its precision and recall, paving the way to build a more reliable detection system.

Figure 8 gives us an idea of the correctness of the instance detection by YOLOv8. The YOLOv8 model was able to correctly identify 77 percent of the car instances and 65 percent of the human instances correctly. YOLOv5 model tend to classify instances more correctly then YOLOv7 and YOLOv8.

Table 1 provides a detailed overview of the training and validation losses for YOLOv5 and YOLOv8 models through different epochs.Notably, YOLOv5 shows a consistent decrease in both training and validation box losses, indicating progress and improvement in its ability to predict bounding boxes for objects accurately. This apparent reduction in losses showcases effective learning and improved generalization over the course of training epochs.On the contrary, YOLOv8 follows a similar trend but with slightly higher values for training and validation box losses.

This epoch-wise breakdown allows us to to observe the incremental refinement in the YOLOv5's performance, showcasing lower losses with each epoch. Although YOLOv8 underwent a similar learning trajectory, it had a slight noticeable contrast in localization accuracy. The results underline YOLOv5's superior performance in minimizing errors during both, training and validation, proving its proficiency in object detection tasks. While these models progress through epochs, these metrics serve as critical indicators of

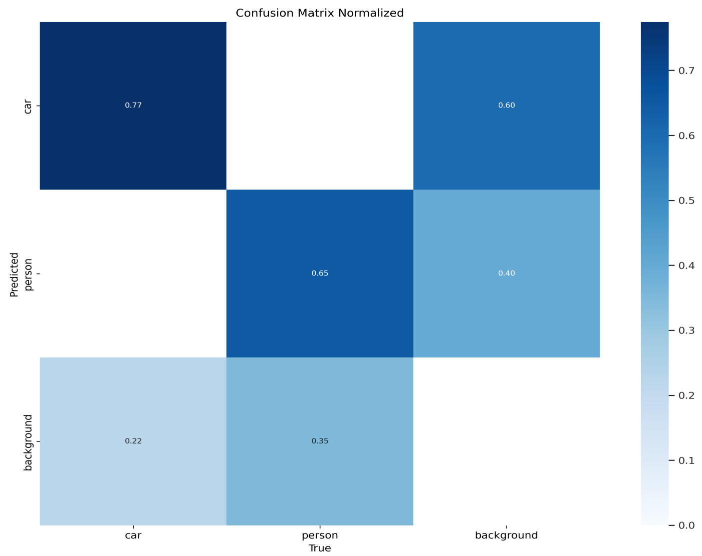

Figure 8: Confusion Matrix for YOLOv8

Model	Epoch	Train/Box Loss	Val/Box Loss
YOLOv5	0	0.079958	0.054795
YOLOv5	1	0.059836	0.048696
YOLOv5	2	0.053889	0.048351
YOLOv5	3	0.048373	0.043722
YOLOv5	4	0.046595	0.042407
YOLOv8	0	1.5517	1.6221
YOLOv8	1	1.5095	1.6067
YOLOv8	2	1.4671	1.6068
YOLOv8	3	1.4109	1.437
YOLOv8	4	1.3824	1.4379

Table 1: Training and Validation Metrics for YOLOv5 and YOLOv8

both model's learning capabilities and generalization to unseen data.

In Summary, the above table offers valuable insights about the different learning trajectories of YOLOv5 and YOLOv8, emphasizing the former's ability to achieve excellent localization accuracy during the training and validation phases.Thus, training and validation losses coupled with metrics like precision, recall, and mAP,would help contribute to holistic evaluation of every model's performance in object detection.

Table 2 displays the precision, recall and mean average precision values for YOLOv5, YOLOv7 and YOLOv8. Here, "All" represents the average of the car and person detection values. Among these, YOLOv7 fail to perform optimally when compared to YOLOv5 and YOLOv8.It is lagging in all the three critical evaluation metrics.Among YOLOv5 and YOLOv8, he YOLOv8 had slightly higher precision and recall values than YOLOv5 .The YOLOv5 model and YOLOv8 model showed similar capabilities while considering mean average precision . The average GPU memory utilized during every epoch by YOLOv5,YOLOv7 and YOLOv8 was 7.59 Gigabytes, 12.5 Gigabytes and 4.06 Gigabytes. It reveals that YOLOv7 was being too computationally expensive followed by YOLOv5 and YOLOv8. Overall, it can be concluded from the results that YOLOv8

Model	Class	Precision	Recall	mAP50
YOLOv5	All	0.806	0.655	0.77
	Car	0.784	0.744	0.821
	Person	0.829	0.567	0.72
YOLOv7	All	0.527	0.367	0.403
	Car	0.479	0.454	0.457
	Person	0.574	0.281	0.348
YOLOv8	All	0.811	0.666	0.771
	Car	0.814	0.736	0.823
	Person	0.808	0.595	0.719

Table 2: Precision, Recall and mAP for YOLOv5, YOLOv7, and YOLOv8

was the best performing model in this study in terms of both efficiency and computational requirements. There may be multiple reasons behind this optimal performance of YOLOv8. YOLOv8 have introduced several architectural improvements and modifications over YOLOv5 and YOLOv7. YOLOv8 may also have incorporated advanced optimization strategies or learning strategies during training. The efficient utilization of GPU Memory by YOLOv8 could have contributed to its better performance, since efficient memory usage allows for larger batch sizes,resulting in improved efficiency. Active contribution and collaborative effort by the community also may have enhanced the YOLOv8's detection capabilities.

8 Conclusion and Future work

This study conducted a comprehensive analysis of the YOLOv5,YOLOv7 and YOLOv8 models to assess their performance in the context of object detection for thermal images. The study revealed that each YOLO variant has its unique strengths and weaknesses in terms of precision, recall and mean Average Precision. YOLOv5 model and YOLOv8 had almost similar competitive performance,with the latter slightly outperforming in most metrics. This comparative analysis underscores the importance of selecting the appropriate YOLO variant based on the specific application requirements or computational constraints. The YOLO models displayed promising results despite the challenges posed by thermal image processing which will benefit applications in the field of Autonomous vehicles. However, there is still room for improvement considering precision and mean Average precision metrics and further research should focus on further enhancing the overall efficiency of YOLO models. Exploring the integration of thermal images with other sensors could lead to capturing more intricate details and features in the images leading to formulation of more robust and accurate detection systems. Additionally, testing these models in a wider and diverse range of real-world scenarios will be crucial to get an overall idea of the robustness and reliability of these models. Lastly, it is also important to investigate the potential of newer and upcoming YOLO variants in Thermal Image processing .

9 Acknowledgment

In this section, I would like to express my sincere gratitude to everyone who have contributed to the completion of my thesis. First and foremost, I extend my deepest appreciation to my supervisor Bharat Agarwal, for his valuable mentorship and unwavering support throught the research process. Their constructive feedback and expertise have been instrumental in shaping this research project. I would also like to thank National College of Ireland for providing the necessary resources and facilities for conducting my research. My sincere thanks also extends to my friends and colleagues who offered unwavering support and understanding throught this academic journey.

References

Balasubramaniam, A. and Pasricha, S. (2022). Object detection in autonomous vehicles: Status and open challenges, *arXiv preprint arXiv:2201.07706* .

Gašparović, B., Mauša, G., Rukavina, J. and Lerga, J. (2023). Evaluating yolov5, yolov6, yolov7, and yolov8 in underwater environment: Is there real improvement?, *2023 8th International Conference on Smart and Sustainable Technologies (SpliTech)*, IEEE, pp. 1–4.

Ignatious, H. A., Khan, M. et al. (2022). An overview of sensors in autonomous vehicles, *Procedia Computer Science* **198**: 736–741.

Krišto, M., Ivasic-Kos, M. and Pobar, M. (2020). Thermal object detection in difficult weather conditions using yolo, *IEEE access* **8**: 125459–125476.

Kurfiss, N. and Jackson, M. (2023). Radar allows surveillance cameras to sense before they can see. Analog Devices EngineerZone Spotlight.
URL: *https://ez.analog.com/ez-blogs/b/engineerzone-spotlight/posts/radar-allows-surveillance-cameras-to-sense-before-they-can-see*

Lianqiao, L., Xiai, C., Huili, Z. and Ling, W. (2019). Recognition and application of infrared thermal image among power facilities based on yolo, *2019 Chinese Control And Decision Conference (CCDC)*, IEEE, pp. 5939–5943.

Miethig, B., Liu, A., Habibi, S. and Mohrenschildt, M. v. (2019). Leveraging thermal imaging for autonomous driving, *2019 IEEE Transportation Electrification Conference and Expo (ITEC)*, IEEE, pp. 1–5.

Olorunshola, O. E., Irhebhude, M. E. and Evwiekpaefe, A. E. (2023). A comparative study of yolov5 and yolov7 object detection algorithms, *Journal of Computing and Social Informatics* **2**(1): 1–12.

Rivera Velázquez, J. M., Khoudour, L., Saint Pierre, G., Duthon, P., Liandrat, S., Bernardin, F., Fiss, S., Ivanov, I. and Peleg, R. (2022). Analysis of thermal imaging performance under extreme foggy conditions: applications to autonomous driving, *Journal of imaging* **8**(11): 306.

Shokri, D., Larouche, C. and Homayouni, S. (2023). A comparative analysis of multi-label deep learning classifiers for real-time vehicle detection to support intelligent transportation systems, *Smart Cities* **6**(5): 2982–3004.

Shorten, C. and Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning, *Journal of big data* **6**(1): 1–48.