Computer Vision

Final project report

Detect person in thermal image using YOLOv5

Group 9

22BI13181 Do Cong Tuan Hung
22BI13148 To Thanh Hai
22BI13115 Pham Hoang Duong
22BI13043 Trinh Duc Anh
22BI13146 Nguyen Dinh Hai

May 2025

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Thermal human detection finds people using heat signatures instead of visible light. It works in darkness, smoke, or harsh weather—useful for security, search & rescue, and driver assistance. In this project, we train a YOLOv5n model to spot humans in thermal frames.

The pipeline is as follow:

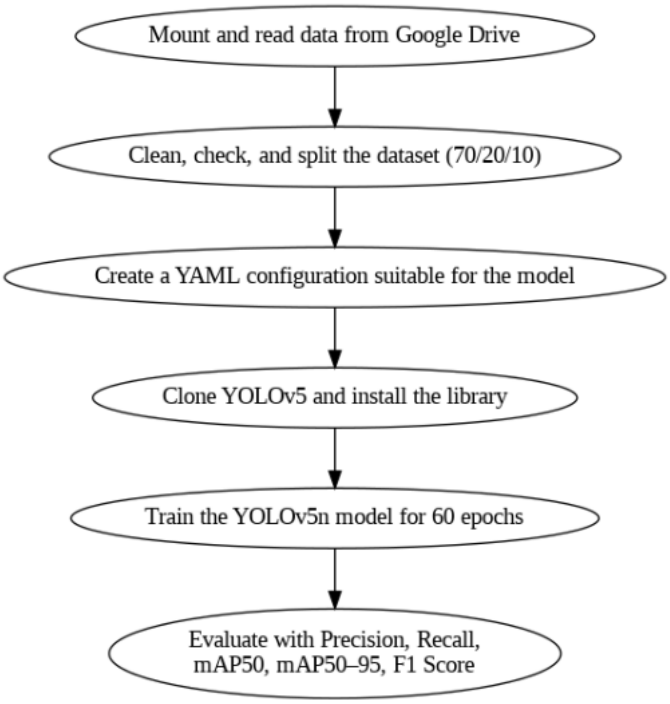
Mount and read data from Google Drive
Clean, check, and split the dataset (70/20/10)
Create a YAML configuration suitable for the model
Clone YOLOv5 and install the library
Train the YOLOv5n model for 60 epochs
Evaluate with Precision, Recall, mAP50, mAP50–95, F1 Score

Figure 1.1: Training Pipeline

1. **Data access**: mount and read images and labels from Google Drive.

2. **Dataset preparation**: clean errors, verify labels, then split into 70% training, 20% validation, 10% testing.
3. **Configuration**: build a YAML file so YOLOv5 reads our data correctly.
4. **Model Setup**: clone the YOLOv5 repo, install dependencies, and load the lightweight YOLOv5n pre-trained weights into Google Colab.
5. **Training**: train for 60 epochs.
6. **Evaluation**: Evaluate with Precision, Recall, mAP50, mAP50–95,F1 Score.

1.1 Related Work

One of the well-known studies that inspired us to undertake this project is ”Human Detection in Thermal Images Using YOLOv8 for Search and Rescue Missions.” In their 2023 study, Rizk and Bayad proposed a human detection system for search and rescue missions using YOLOv8 applied to thermal imagery. They introduced a curated dataset of 17,148 grayscale thermal images containing over 90,000 human instances, collected from multiple sources (e.g., HIT-UAV) and annotated specifically for human detection. The dataset includes diverse conditions such as weather, time of day, and camera types (e.g., UAVs, handheld, vehicle-mounted). Models were trained and evaluated across five YOLOv8 variants, demonstrating strong potential for real-world deployment in low-visibility SAR scenarios.

The model created by Rizk and Bayad will also serve as a point of reference and comparison with our YOLOv5n model.

Chapter 2

Dataset and Preprocessing

We used a thermal image dataset inspired by and partially based on the UNIRI-TID dataset. The dataset consists of grayscale thermal images containing human subjects captured under various conditions. Images were collected in different environments and lighting situations to simulate real-world scenarios where RGB-based detection may fail.

What other studies have used this dataset for? The UNIRI-TID thermal image dataset has been widely used in various projects aimed at detecting humans in thermal imagery under challenging environmental conditions. One such project was conducted during the SSIP 2020 (Summer School on Image Processing), where students implemented the YOLOv4 model using the Darknet framework to detect persons in thermal images. The project provided participants with hands-on experience in deep learning techniques and demonstrated promising detection accuracy despite the variability and noise present in thermal data.

Another significant application of the dataset was presented in the research paper titled “Thermal Object Detection in Difficult Weather Conditions Using YOLO” by Marina Ivašić Kos and colleagues (the original creator of the UNIRI-TID dataset). This study used the UNIRI-TID dataset to train a YOLO model for detecting people in thermal images, specifically under adverse weather conditions such as fog and light rain. The research validated the effectiveness of YOLO-based models in low-visibility scenarios and highlighted the practical utility of thermal data for safety-critical applications.

The preprocessing pipeline of our project included the following steps:

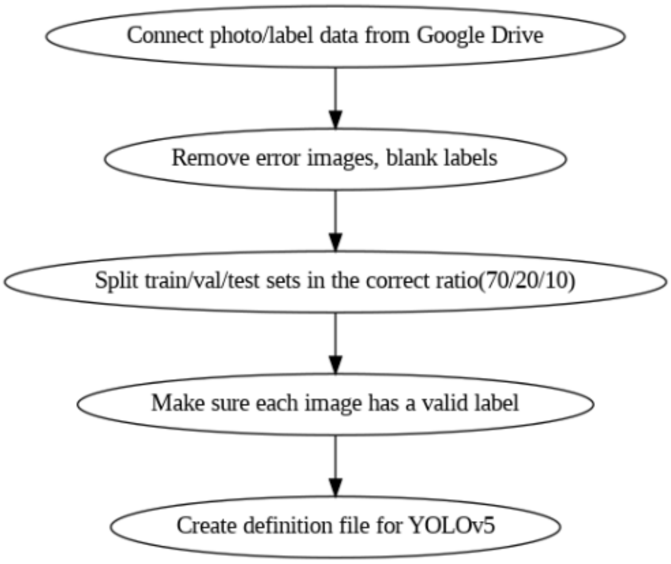
Connect photo/label data from Google Drive
Remove error images, blank labels
Split train/val/test sets in the correct ratio(70/20/10)
Make sure each image has a valid label
Create definition file for YOLOv5

Figure 2.1: Preprocessing Pipeline

1. **Mounting Data**: Image and label files were mounted from Google Drive for fast access during training.

2. **Cleaning**: Corrupted images and empty label files were removed to prevent training errors.

3. **Integrity Check**: Each image was verified to have a corresponding valid label.

4. **Splitting**: The dataset was divided into training (70%), validation (20%), and testing (10%) sets to ensure fair model evaluation and avoid overfitting.

5. **YAML Configuration**: A data.yaml file was created to define paths and class labels, allowing YOLOv5 to correctly read the dataset.

This preprocessing ensured that the dataset is correctly mounted, formatted, and ready for training with the model YOLOv5n.**The input data** will be **thermal images**, and **the output** will be **thermal images with bounding boxes** identifying the detected individuals.

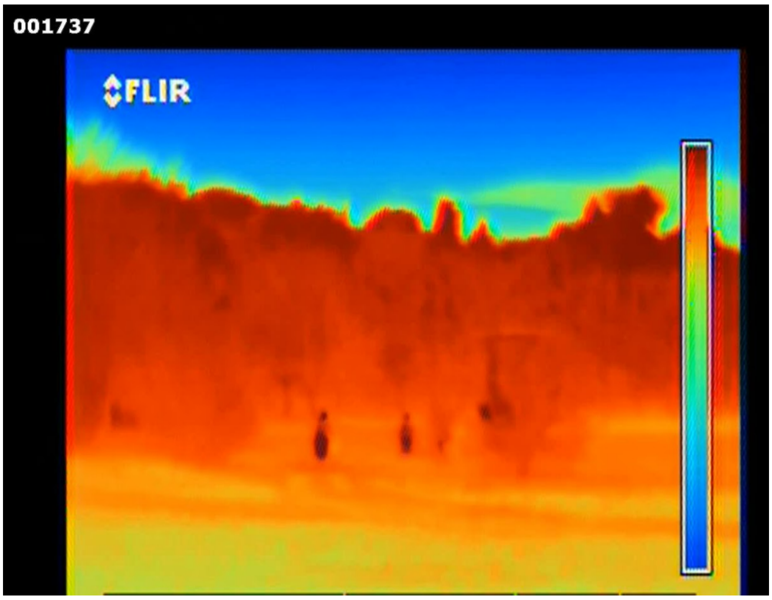

Figure 2.2: Example of an Input Thermal Image

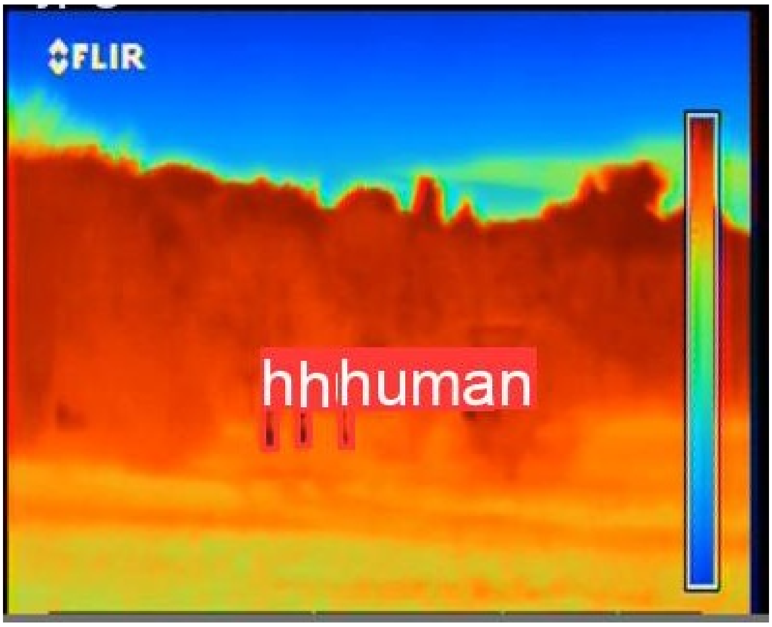

Figure 2.3: Example of Output Thermal Image

Chapter 3

Model Training

For this project, we used the YOLOv5n model, the smallest and fastest variant in the YOLOv5 family. It is well-suited for environments with limited computing power, such as Google Colab, and performs well on binary classification tasks like human vs. background detection.

The training process involved cloning the YOLOv5 repository, setting up dependencies, and configuring training parameters through opt.yaml. The model was trained for 60 epochs on 640×640 pixel input images, using a batch size of 16.

The training ran for approximately 2.2 hours. In total, the model had around 1.76 million parameters and consumed 4.1 GFLOPs per inference. By the end of training, the model showed strong convergence and reliable detection performance, especially at the IoU 0.5 threshold.

3.1 Hyper-parameters Adjustment

In this section, we will present both the adjustments to our hyper-parameters to our model, as well as those of Human Detection in Thermal Images Using YOLOv8 for Search and Rescue Missions that we mentioned in the Related Work section, in order to compare and contrast, provide reasons for our and their approaches, and explain why these adjustments are suitable for both models.

Note: Rizk and Bayad use multiple YOLOv8 models at the same time, and from direct communication with them, we know that they use Hyper-parameters and Data Augmentation consistently for all models.

Hyper-parameters used:

Parameter	**YOLOv5n (Ours)**	**YOLOv8 (Rizk & Bayad)**	**Remarks**
Learning rate (lr_0)	0.01	0.01	Identical initial learning rate; appropriate for lightweight or mid-scale training.
Final LR fraction (lrf)	0.1	∼0.01 (default)	Gradual decay in v5n stabilizes learning; YOLOv8 likely uses a smaller decay.
Momentum	0.937	0.925	Consistent—helps smooth weight updates and convergence.
Weight Decay	0.0005	0.0005	Prevents overfitting in both cases.
Warmup Epochs	3.0	Not specified	Used in YOLOv5n to avoid unstable early gradients; not emphasized in YOLOv8.
Warmup Momentum	0.8	Not specified	Smoother transition during initial epochs for v5n.
Box Loss Gain	0.05	∼0.05	Standard value; stable for thermal localization.
Class Loss Gain	0.5	∼0.5	Single-class (human), so no need for higher class weight.
Object Loss Gain	1.0	1.0	Same objectness focus; vital for detection-centric tasks.
IoU Threshold	0.2	0.25	Lower IoU helps detect small/distant persons, suitable for SAR.
Anchor Threshold	4.0	Not mentioned	Loose anchor assignment helps cover diverse object sizes.
Focal Loss Gamma	0.0	0.0	Focal loss disabled—class imbalance not critical here.

Table 3.1: Comparison of YOLOv5n and YOLOv8 Configurations for Thermal Human Detection

Suitability Analysis

Our Configuration (`YOLOv5n`)

- A lightweight model (`YOLOv5n`) suitable for resource-constrained devices, such as those mounted on actual UAVs.
- Thorough use of *warmup* helps prevent gradient instability when training small models on challenging data like thermal images.
- `IoU threshold = 0.2` and `anchor_t = 4.0` ensure detection of very

small or hidden human figures — ideal for real-world search & rescue missions.

- *Focal loss* is not used, which is reasonable since there is only one class (human), so no severe class imbalance exists.

Rizk and Bayad's Configuration (`YOLOv8`)

- A more powerful model, using various `YOLOv8` variants from small to large, aimed at comprehensive evaluation on a large dataset (17k images).
- Warmup is not explicitly mentioned, but given the large `batch size` and deeper architecture, skipping an extensive warmup phase is justifiable.
- Focuses more on input diversification via *augmentation* rather than extensive hyperparameter tuning.

3.2 Data Augmentation

To improve generalization, several data augmentation techniques were applied during training. These augmentations helped the model handle variations in scale, orientation, and image quality. Similarly to the Hyper Parameter section, we will also compare them with Human Detection in Thermal Images Using YOLOv8 for Search and Rescue Missions.

The augmentations used include:

Technique	YOLOv8	YOLOv5n	Remarks

Flip	HorizontalFlip(p=0.5)	fliplr=0.5,
		flipud=0.0
			Both perform horizontal flipping; YOLOv5n disables vertical.

Crop & Resize	RandomResizedCrop(
	640×640,
	scale=0.75–1.0,
	p=0.5)
		scale=0.2 (resize only)	YOLOv8 crops and zooms in for partial object views; YOLOv5n only rescales.

Rotation	Rotate(limit=6°, p=0.5)	degrees=5.0	Both simulate small camera angle shifts.
Color Adjustment	HueSaturationValue(±20), ToGray(p=0.25), RandomBrightness-Contrast	hsv(h=0.015, s=0.4, v=0.6)	Both simulate lighting/camera color variation. YOLOv8 includes grayscale.
Blur	GaussianBlur(0.75px)	Not used	YOLOv8 mimics soft focus or motion blur.
Occlusion	CoarseDropout (1–5 patches)	copy paste(p=0.1)	YOLOv8 simulates occlusion by hiding regions; YOLOv5n adds occluded objects.
Perspective	Not used	perspective=0.0005	YOLOv5n includes slight perspective distortion.

Image Mixing	Not used	mosaic=1.0,
		mixup=0.1
			YOLOv5n boosts variation by combining images.

Label Smoothing	Not used	label smoothing=0.1	YOLOv5n softens one-hot labels to prevent overconfidence.

Table 3.2: Comparison of YOLOv5n and YOLOv8 Data Augmentation Pipelines

It can be seen that the v8 models only focuses on gentle and intuitive transformation techniques such as cropping, rotating slightly, grayscale conversion, blurring, etc., which help the model learn geometric and lighting features well. Meanwhile, our model uses many powerful data augmentation techniques such as mosaic, mixup, copy-paste to increase the diversity of layouts and training situations, affirming the higher quality of the dataset and the ability to train, predicting a better result.

3.3 Model Evaluation

Evaluating model performance is a crucial step in ensuring its effectiveness. Various evaluation metrics are employed to quantify how well the model detect object within unseen thermal images. This section introduces the key evaluation metrics used in this study, including **Precision**, **Recall**, **mAP50**, **mAP50-95**, and **F1-score**.

- **Precision**: The ratio of correctly predicted positives to all predicted positives.
  Formula:
  $$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{3.1}$$
- **Recall**: The ratio of correctly predicted positives to all actual positives.
  Formula:
  $$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3.2}$$
- **mAP50**: Mean Average Precision at an Intersection-over-Union (IoU) threshold of 0.5.
  Formula:
  $$\text{mAP}_{0.5} = \frac{1}{|C|} \sum_{c \in C} \text{AP}_c(\text{IoU} = 0.5) \tag{3.3}$$
- **mAP50-95**: Mean Average Precision averaged over multiple IoU thresholds from 0.5 to 0.95.
  Formula:
  $$\text{mAP}_{0.5:0.95} = \frac{1}{10} \sum_{t \in \{0.50, 0.55, \ldots, 0.95\}} \left(\frac{1}{|C|} \sum_{c \in C} \text{AP}_c(\text{IoU} = t) \right) \tag{3.4}$$
- **F1-score**: The combination of precision and recall.
  Formula:
  $$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.5}$$

3.3.1 Why the comparison with Rizk and Bayad's v8 models?

Throughout this report, multiple variations of similar topic were presented but we decided to use the specific models made by Rizk and Bayad for the following reasons:

1. The models trained on the same dataset (UNIRI-TID) use an approach that differs significantly from ours. We use a different framework, hyperparameters, and data augmentation methods. So, despite using the same dataset, comparing the two would be misleading. Their work deserves to be appreciated on its own terms, and our model follows a completely different direction. By doing so, we contribute a new perspective using a lightweight model with specific configurations rather than slightly tweaking someone else's setup, which could weaken the results.

2. We were, however, inspired by the work of Rizk and Bayad. Although the datasets differ, our training approach closely follows theirs, as detailed in the "Model Training" section. This similarity allows for a fair comparison without undermining the validity of either model.

3.3.2 Results

We trained the model for 60 epochs (2.19 hours total) on a single class ("human") with 1,760,518 parameters. The computational cost was 4.1 GFLOPs. The results are documented across multiple aspect which we will go over one by one.

3.3.2.1 Validation Results

On the held-out validation set, the model achieved:

Metric	Value
Precision	0.991
Recall	0.991
mAP50	0.994
mAP50-95	0.662

Table 3.3: Validation results of our Model

These results show extremely high precision and recall at IoU = 0.5, higher than the related work YOLOv8 model in most metric, with the same IoU.

Model	**Precision**	**Recall**	**mAP50**	**mAP50-95**
YOLOv8x	0.927	0.879	0.946	0.651
YOLOv8m	0.932	0.863	0.939	0.641
YOLOv8n	0.910	0.828	0.913	0.589

Table 3.4: Performance Comparison of YOLOv8 Models

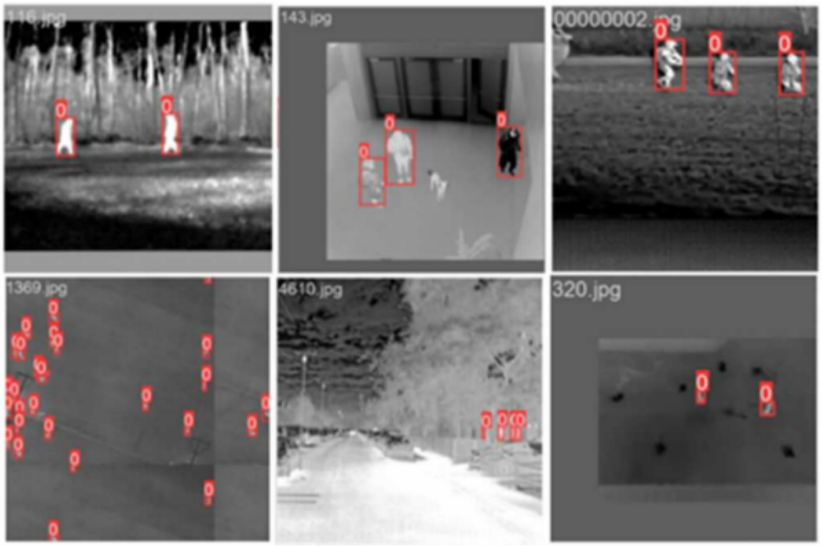

Figure 3.1: Training and Validation performance of YOLOv8 model.

3.3.2.2 Test Performance

For further information, the table below show a separate test split of 1,201 images, the class-level breakdown was:

Class	Precision	Recall	F1-score	Support
No Human	1.00	0.64	0.78	11
Human	1.00	1.00	1.00	1190
Accuracy			0.998	1201
Macro avg	1.00	0.82	0.89	1201
Weighted avg	1.00	1.00	1.00	1201

Table 3.5: Test results of our Model

- **Mean Average Precision**(mAP): 0.9966
- **F1-score**: 0.9983

Inference time on GPU (per image):

- **Mean**: 0.123s
- **Standard Deviation**: 0.040s
- **Min**: 0.096s
- **Max**: 0.751s

3.3.2.3 Error Case

As seen, the values of the metrics are not 100%, meaning there exist cases of errors where humans cannot be detected in thermal images.

Below are some example of error case:

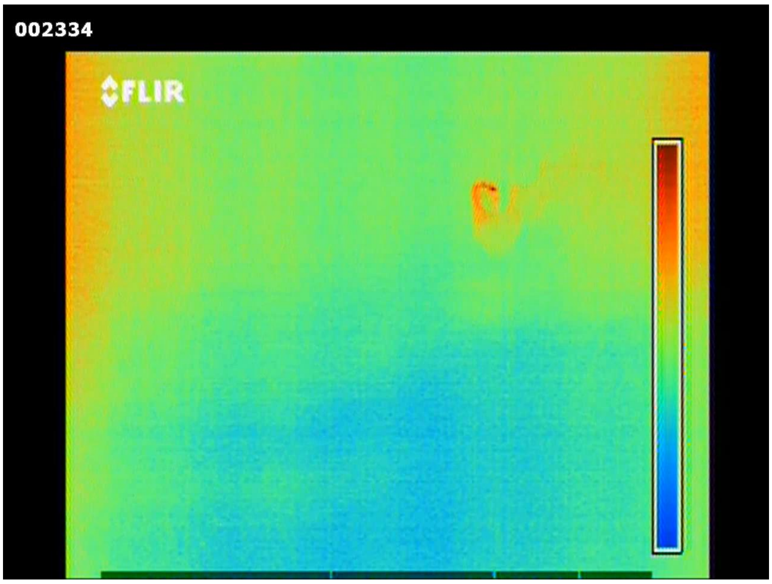

Figure 3.2: Fail Image 1

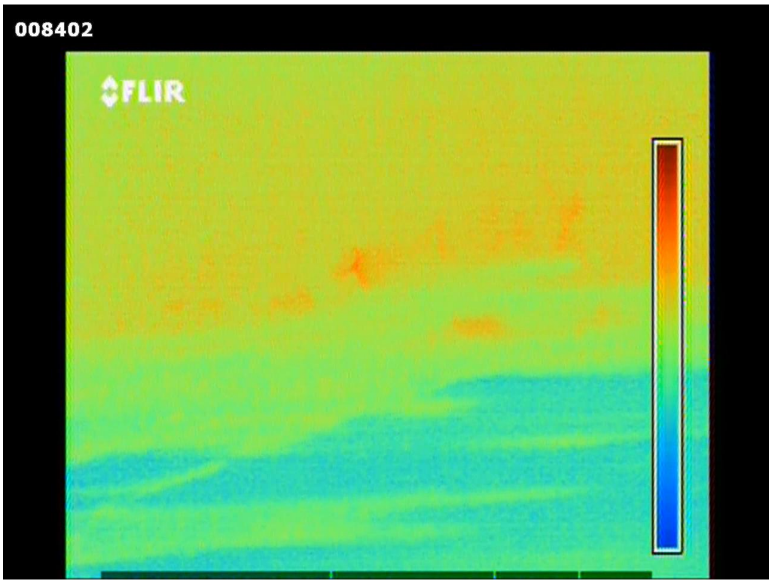

Figure 3.3: Faile Image 2

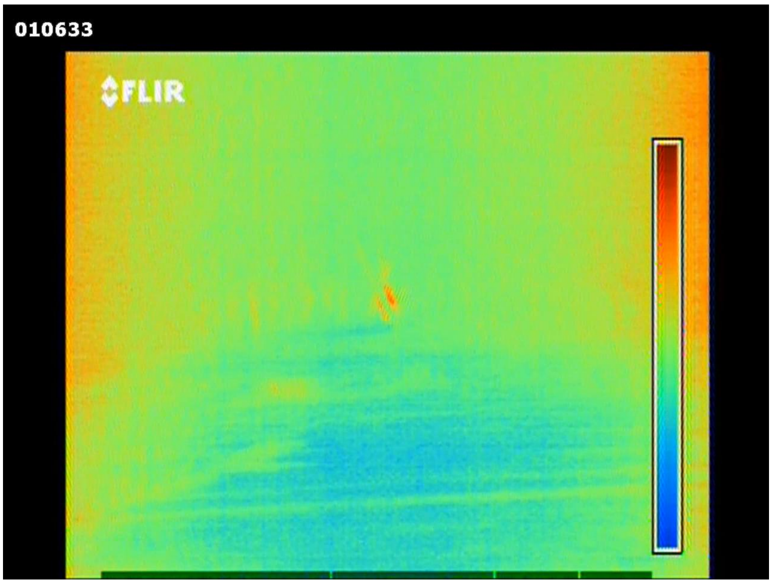

Figure 3.4: Fail Image 3

Our model fail to detect human in these images possibly due to:

- Low contrast: Low visual contrast which affected the model's ability to distinguish human with the environment.
- Blurry edges: There is no clear edges of the human for the model to pick up.

3.3.2.4 Confusion Matrix

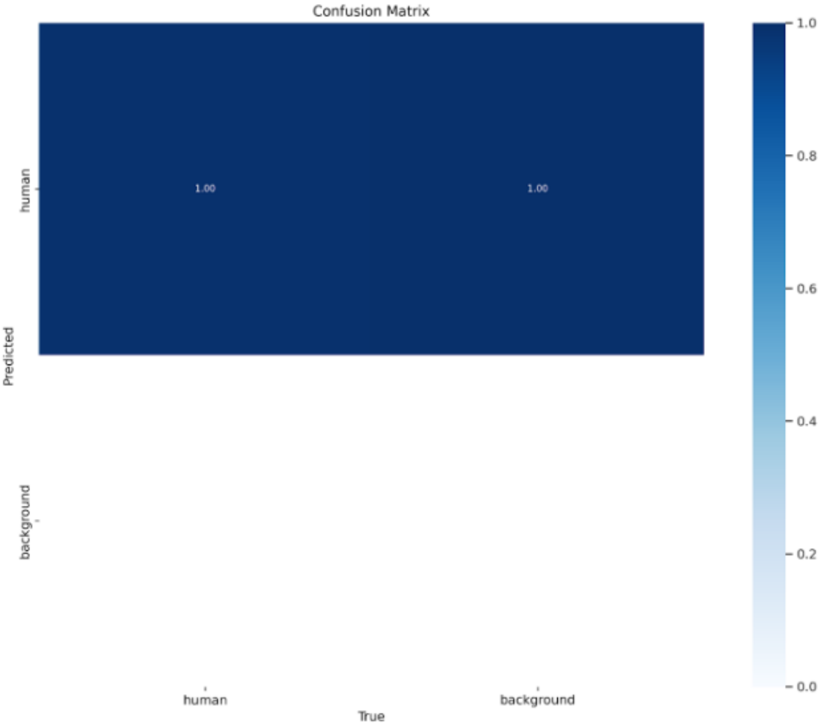

Figure 3.5: Confusion Matrix

- All true ”human” cases were correctly predicted (100% true positive).
- No false positives or false negatives.
- Perfect separation between ”human” and background.

3.3.2.5 Confidence Threshold Analysis

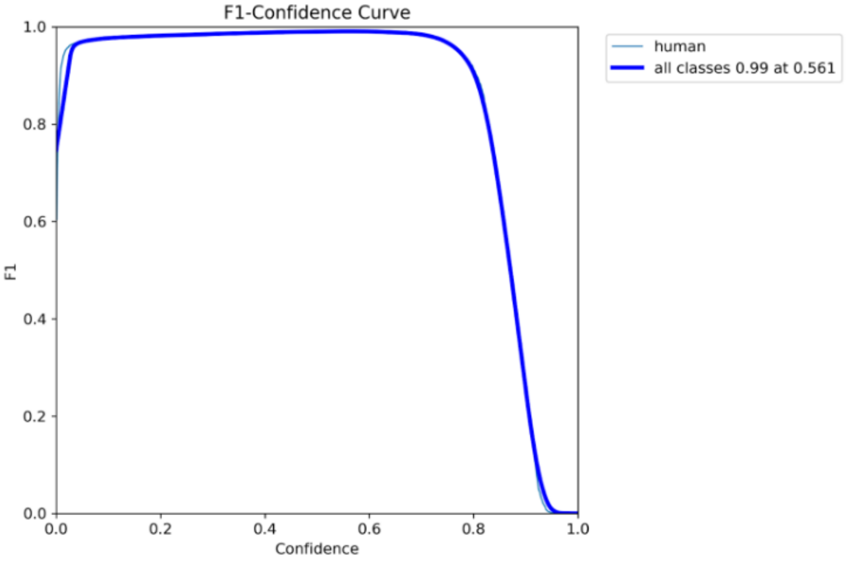

Figure 3.6: F1-Confidence Curve

When comparing with the F1-Confidence images of 3 YOLOv8 models provided from Related Work, we realize that our model really gives superior results in the task.

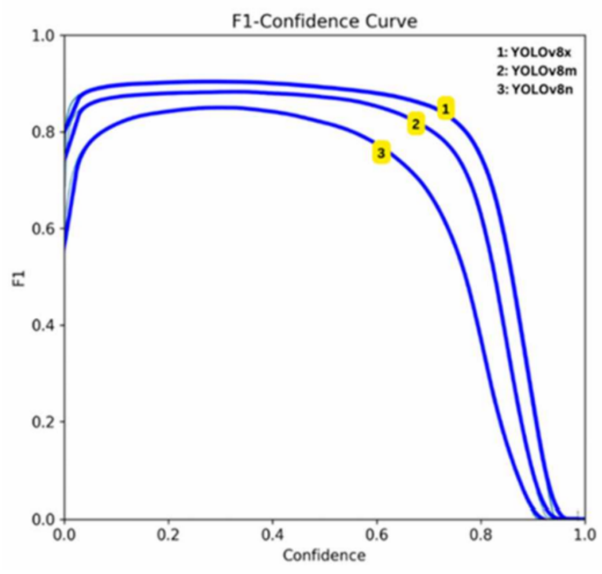

Figure 3.7: Comparative Analysis of 1-Scores for 3 models of YOLOv8

F1-Confidence Curve:

- Peak F1 = 0.99 at confidence of around 0.56.

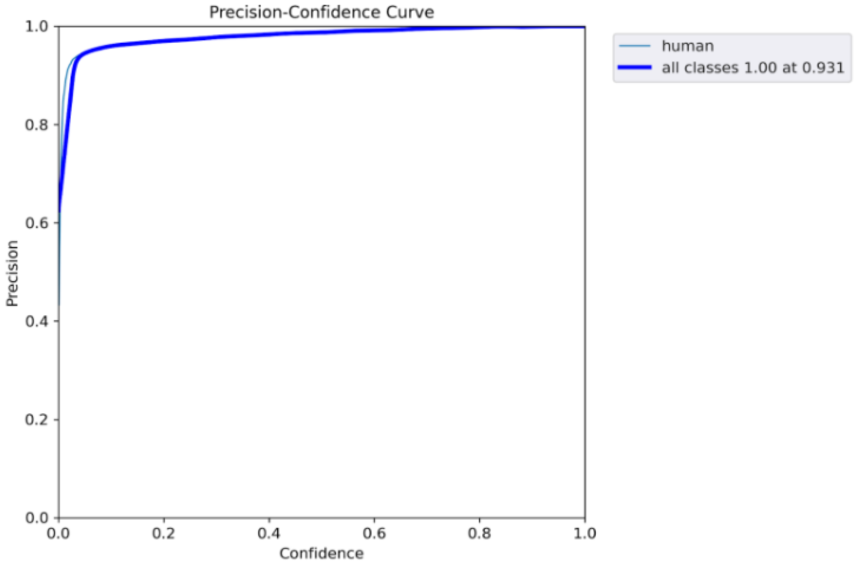

Figure 3.8: Precision-Confidence Curve

Precision-Confidence Curve:

- Precision reaches around 1.0 at confidence of 0.93.
- Near zero false positives.

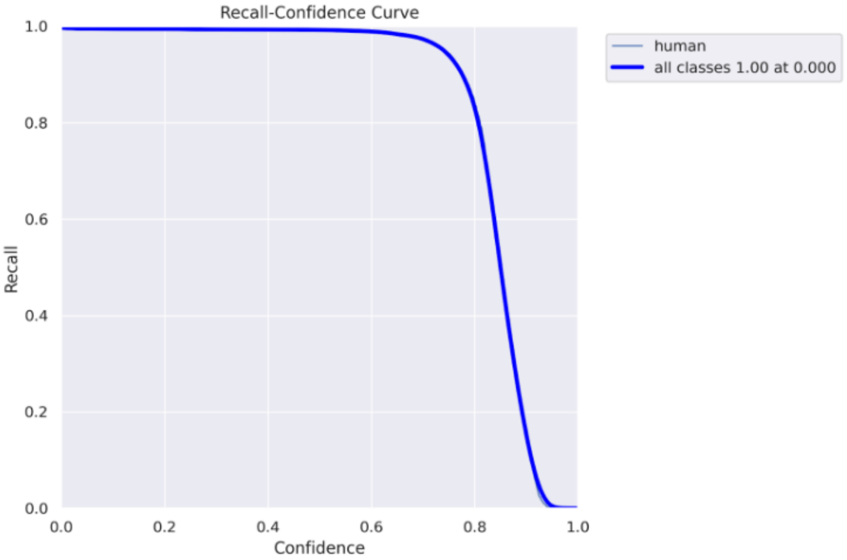

Figure 3.9: Recall-Confidence Curve

Recall-Confidence Curve:

- Recall stays around 1.0 at confidence of 0,85.
- Drops quickly beyond as low-confidence detections are discarded.

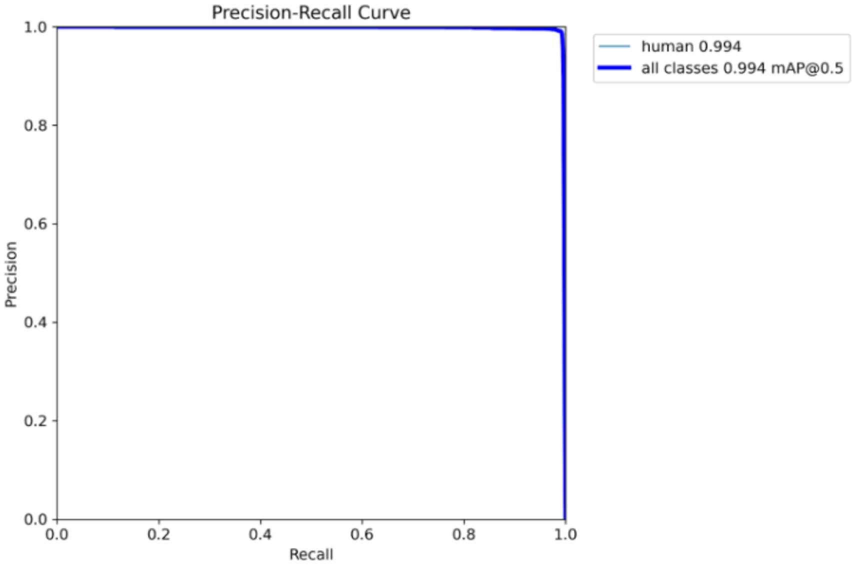

Figure 3.10: Precision-Recall Curve

Precision-Recall Curve:

- Curve hugs the top-right corner.
- Confirms mAP50 = 0.994

3.3.2.6 Bounding Box Distribution

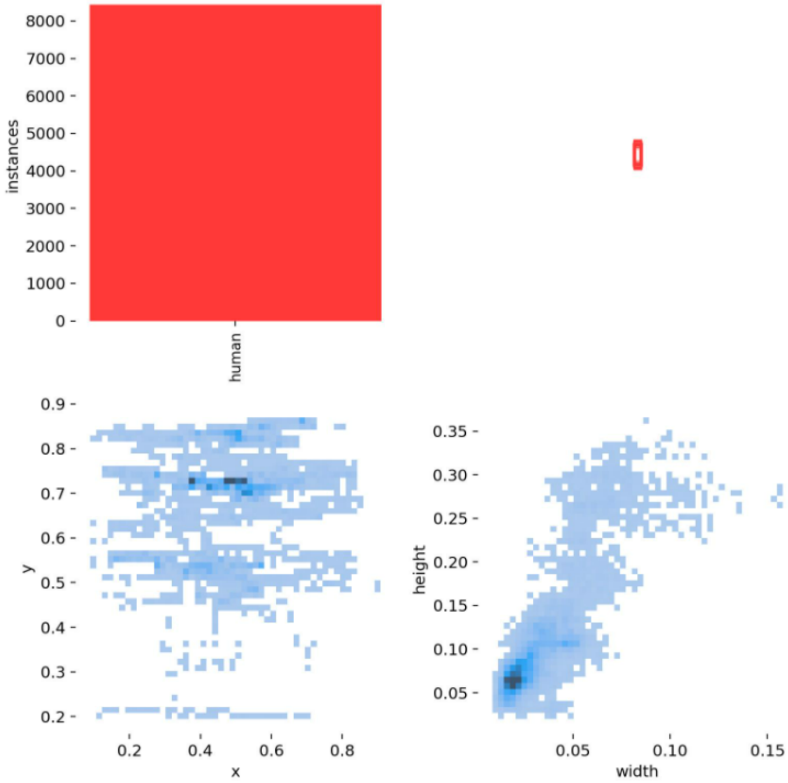

Figure 3.11: Bounding Box Distribution

Class Frequency & Spatial Heatmaps:

- **Frequency**: around 8000 ”human” instances.
- **Center(x, y)**: Detections cluster around y = 0.6 - 0.8 (mid-lower image).
- **Size (w, h)**: Width = 0.03 - 0,07, Height = 0.1-0.25.

3.3.2.7 Correlation Matrix

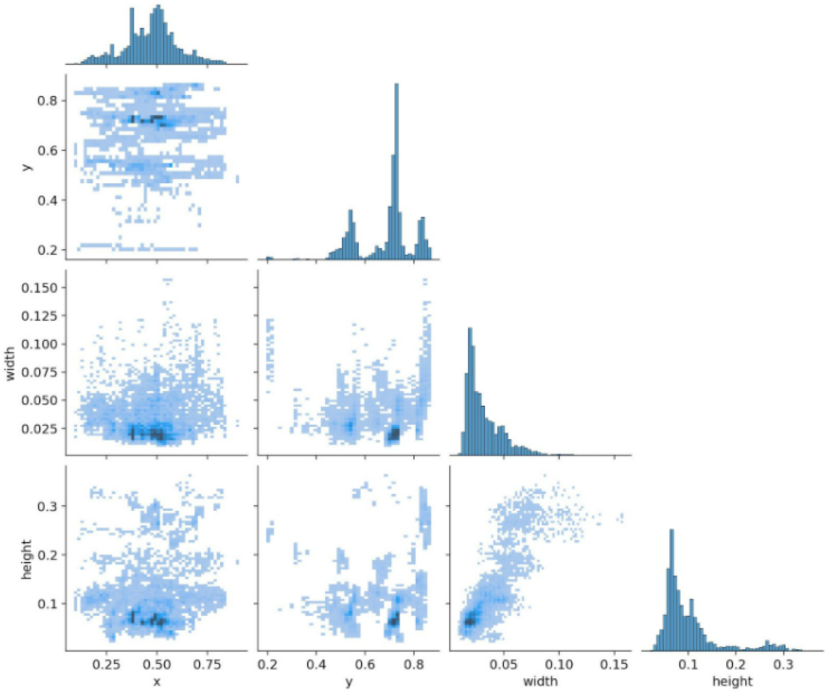

Figure 3.12: Correlation Matrix

- **Diagonals**: Histograms show most boxes centered (x=0.5) and tall.
- **Off-diagonals**:
  - width vs height: positive correlation.
  - Other pairs: no strong correlation.

Chapter 4

Conclusion

This work developed a fast, lightweight detector specialized for thermal images. Training used minimal compute yet yielded near-perfect precision, recall, and F1 on both validation and test sets. Confidence analysis identified 0.56 as the optimal threshold, while higher thresholds eliminated false positives. Bounding-box and correlation studies confirmed that detected human patterns in thermal data align with real-world distributions. With average inference under 0.13 s per frame (8 FPS), the model is accurate and swift. Furthermore, when compared with other models, for example the YOLOv8 models in the related work section, it shows high efficiency, outperforming a lighter and less complex model, confirming the applicability of the study.

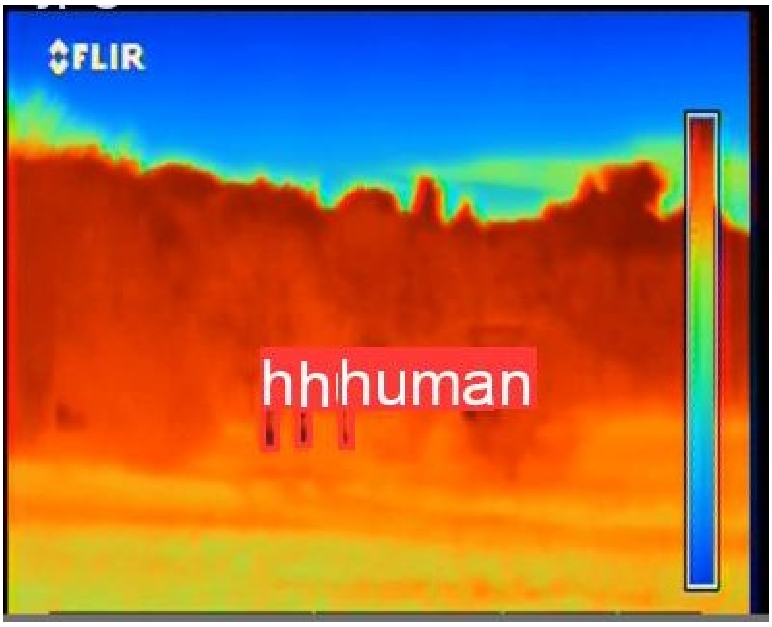

Figure 4.1: Our Sample Results

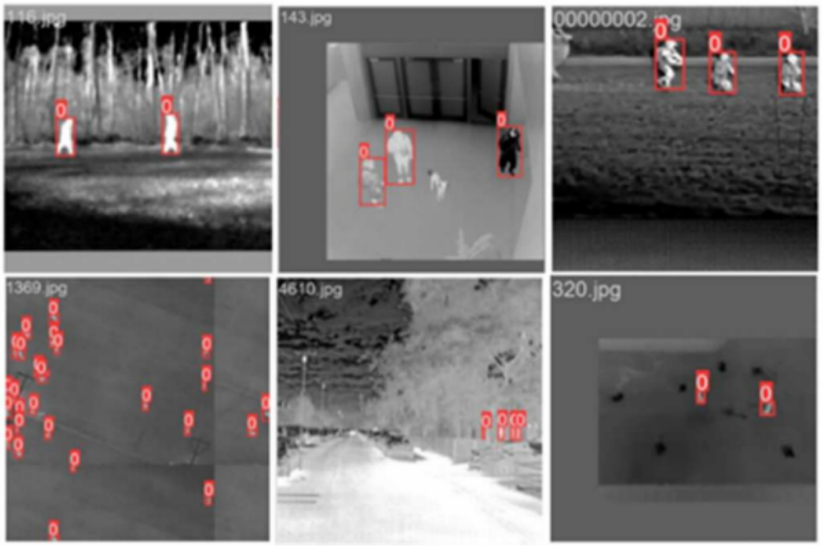

Figure 4.2: Related Work YOLOv8 Models Sample Results

However, there may exist incorrect labels that may affect the heatmap. Moreover, some images in the dataset are of the same scene, which may overlap in some aspects. All of which are room for improvement.

Chapter 5

Reference

Dataset: Thermal image dataset

Inspiration:

- Human Detection in Thermal Images Using YOLOv8 for Search and Rescue Missions
- Person Detection in Thermal Images