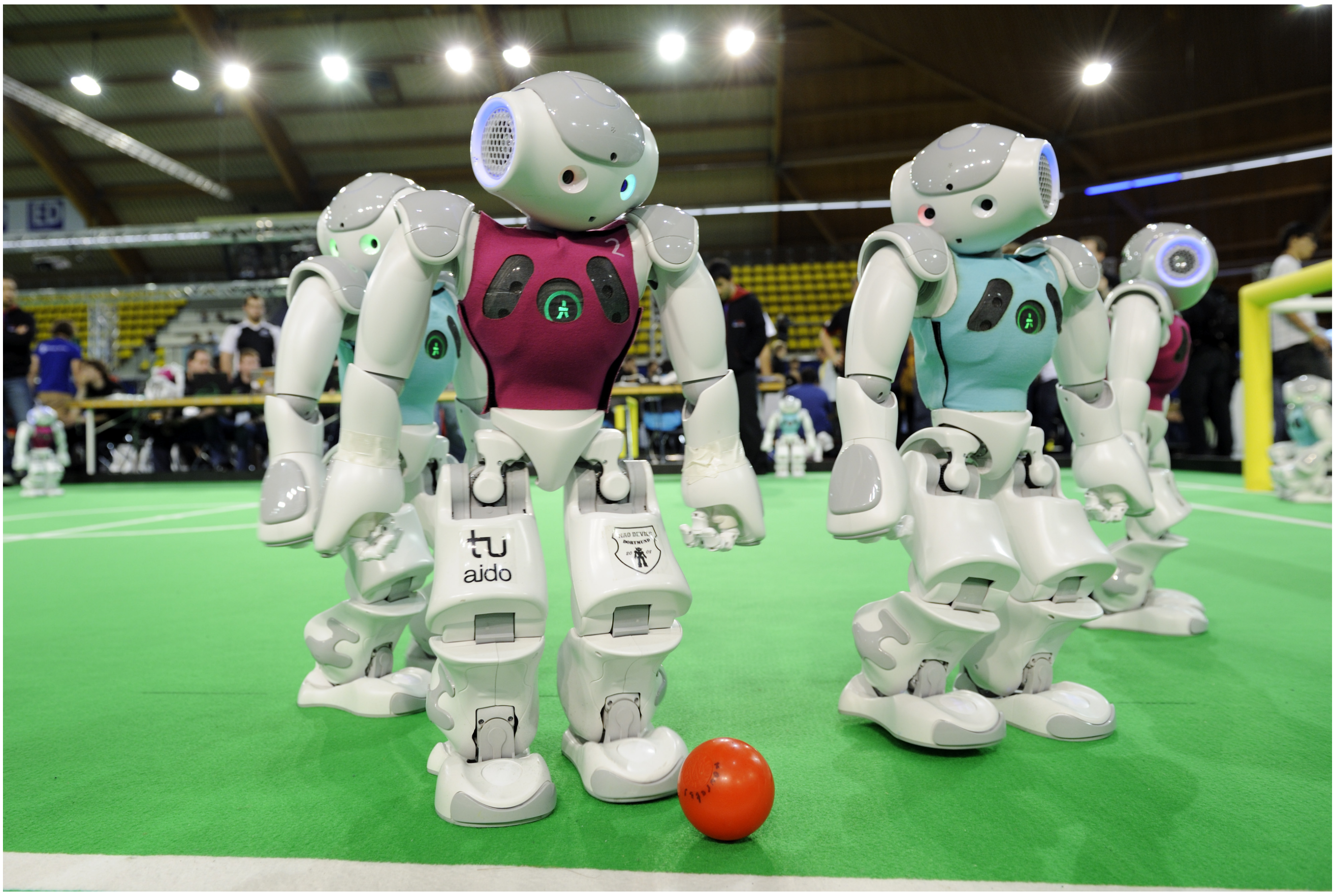
Recognizing All Opponents:

A Real-Time Object Detection Approach for Robotic Soccer Jerseys



Pearl P.D. Owusu

Layout: typeset by the author using LaTeX.
Cover illustration: Bart van Overbeeke

Recognizing All Opponents:

A Real-Time Object Detection Approach for Robotic Soccer Jerseys

Pearl P.D. Owusu
12502340

Bachelor thesis
Credits: 18 EC

Bachelor *Kunstmatige Intelligentie*

University of Amsterdam
Faculty of Science
Science Park 900
1098 XH Amsterdam

Supervisor
Dr. A. Visser

Informatics Institute
Faculty of Science
University of Amsterdam
Science Park 900
1098 XH Amsterdam

February 2, 2024

Abstract

In the realm of robotic soccer, the ability to swiftly and accurately recognize opponents is crucial for effective game play. This thesis delves into the ins and outs of real-time object detection in robotic soccer, particularly focusing on the recognition of opponents using the YOLO algorithm versions 5 and 8. The research is conducted within the framework of the RoboCup Standard Platform League, where NAO robots partake in soccer matches. The primary objective is to enhance the NAO robot's game play by improving the identification of opponents' soccer jerseys. The study explores key performance metrics with a specific focus on evaluating the outcomes of training the YOLO models and offering a comparative analysis between the two versions. This provides insights into the effectiveness of real-time object detection in the context of robotic soccer, contributing to the ongoing discourse in the field.

Contents

1

Introduction

1.1 Object detection

Object detection is a pivotal task in computer vision (1). At its core, computer vision involves identifying multiple objects within an image or a video frame. A generic pipeline includes the acquisition, preprocessing, analyzing and evaluating of the data (2). Object detection adds another layer of complexity by not only identifying objects within an image or video but also localizing them with bounding boxes (3). These bounding boxes serve as rectangular borders that enclose the objects of interest. By performing object detection, intelligent systems can understand their visual surroundings, leading to informed decision-making and appropriate actions, similar to how humans perceive the world (4). This capability opens up a wide range of applications, from autonomous driving and surveillance systems to augmented reality and robotics(5).

Zooming in on robotics, a fascinating application of this technology can be found in the domain of robotic soccer. It explores various fields of artificial intelligence (AI), including behavior representations and visual search algorithms (6). Robotic soccer relies on object detection to enable robots to perceive their environment. Firstly, it allows the robots to locate and track the ball in real-time (7). They can effectively navigate the field, anticipate the ball's trajectory, and plan their actions accordingly. In addition, it also contributes to the overall safety of the game. By accurately detecting the presence of other robots or obstacles, the robots can avoid collisions and minimize the risk of damage to themselves or other objects on the field (8). Furthermore, it enables the robots to differentiate between teammates and opponents. By accurately identifying their teammates, the robots can coordinate their movements and strategize collectively to outplay the opposing team. Also, being able to recognize opponents allows the robots to anticipate their actions, adapt their defense strategies, and effectively counter their attacks. Object detection, therefore, plays a crucial role in enhancing the robots' ability to collaborate and compete in a dynamic environment.

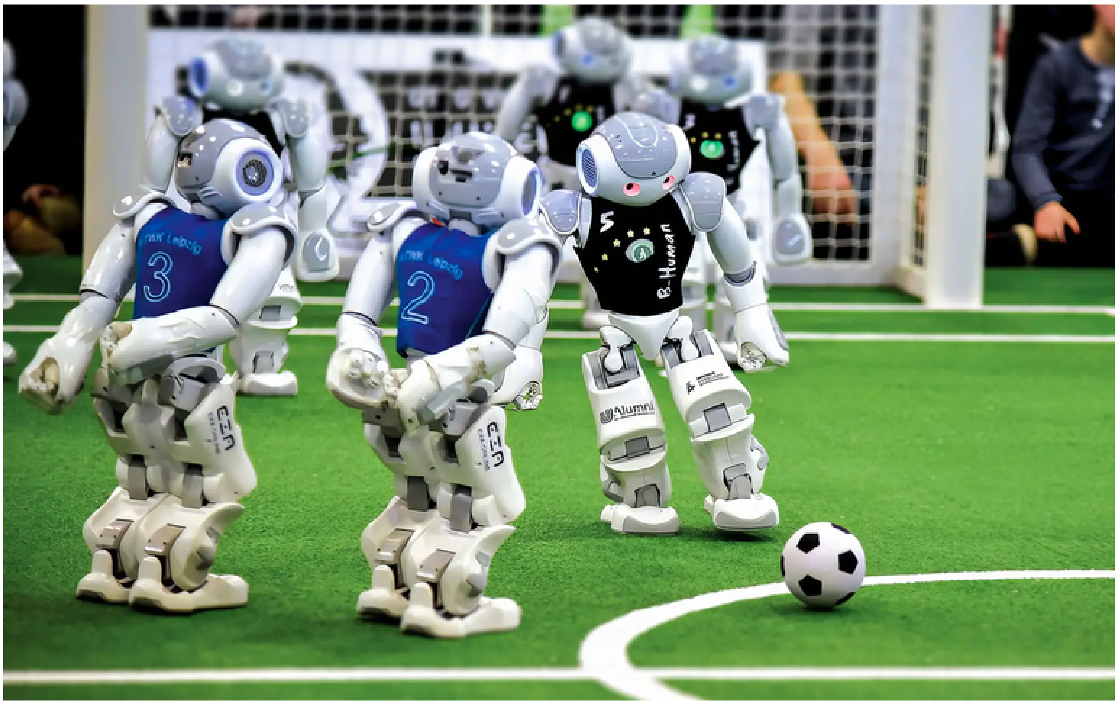


Figure 1.1: A soccer match between teams HTWK Leipzig and B-Human during the RoboCup SPL in 2023[1]

1.2 RoboCup Standard Platform League

For this project, the main object detection tasks are relevant to the RoboCup Standard Platform League(SPL)[2]. In this league, autonomous robots, specifically the NAO robot, engage in soccer matches. The SPL is an international championship that is held annually and serves as a stage where teams from various universities contribute to the research and development of systems and functionalities for NAO robots. The participating teams in the SPL address a diverse array of challenges, ranging from advanced perception and navigation to the complexity of seamless team coordination. The NAO robot, with its humanoid form and limited sensing capabilities, presents unique challenges for object detection. First of all, the robot's vision system relies on a pair of cameras mounted in and on its

[1] `https://lp.unitedrobotics.group/robocup2023-in-bordeaux`

[2] https://spl.robocup.org

head, which provide a limited field of view. Additionally, the robot operates in dynamic and unpredictable environments, where lighting conditions and fast-moving objects can pose significant difficulties for object detection algorithms. To tackle these challenges, participating teams in the SPL apply various computer vision and machine learning techniques to develop robust and accurate object detection algorithms that can handle the complex and dynamic nature of the game.

1.3 Related work

1.3.1 Traditional approach

Over the years, much research has been dedicated to object detection in the context of robotic soccer. Fabisch et al. (9) discuss robot recognition during the RoboCup SPL games and lays the foundation for the problem addressed in this thesis. It highlights the importance of recognizing opponents and provides insights into early methods of a vision based approach. Another early method that was often applied was based on color segmentation, since color is an effective and robust visual cue in a complex environment. (10). Color segmentation is the process of separating different elements in an image based on their color properties. For example, by segmenting the color of the ball, robots can more easily detect its presence and incorporate it into their motion planning and decision-making algorithms. This has previously been applied by Menashe et al. (11). However, color segmentation for object detection in robotic soccer poses several challenges. The lighting conditions in different environments can affect the appearance of the objects, making it difficult to define a fixed color range. To tackle this, robots can perform color calibration before the game, adjusting the color thresholds based on the current lighting conditions. Color segmentation was also involved when identifying opponents based on predefined color ranges corresponding to their jerseys. While this is effective in scenarios with uniform jerseys, these approaches fell short when teams started introducing a variety of designs. This limitation prompted a shift towards deep learning techniques, particularly deep neural networks (DNNs). (12) Consequently, the primary challenge of this project lies in adapting to changing jersey designs, making conventional color segmentation ineffective.

1.3.2 Recent developments

Up to now, challenges still persist in enhancing object detection in robotic soccer. One recently published approach by (13) therefore combines basic robot vision techniques on grayscale images with candidate classification through a convolutional neural network (CNN) to detect all objects on an SPL field. Other innovations include the JET-Net for real-time object detection in mobile robots, a model framework for efficient object detection based on CNNs as well (14). Because there was a need for large annotated datasets for training DNNs, recent efforts have focused on enhancing the available data through Generative Adversarial Networks (GANs)(15). This augmentation strategy helps address the challenge of limited annotated images in the dataset. These contributions lay the groundwork for this project. They underscore the importance of creating and expanding datasets, which highlight advanced techniques for improved object detection.

1.4 Research question

Analyzing the related work, conventional object detection algorithms often face challenges in achieving real-time performance without sacrificing accuracy. In light of this, the research question is:

- *How can we enhance real-time object detection in robotic soccer-playing systems, specifically focusing on recognizing diverse jerseys?*

1.5 Thesis outline

In the subsequent sections of this thesis, we delve into the specifics of our approach, focusing on the NAO robot in the RoboCup SPL. Our goal is to contribute to the evolving field of robotic soccer by addressing the critical need for a enhanced real-time object detection model. This will have a particular emphasis on evaluating the performance of the chosen models.
The main aim is to create an extended training data set that includes jerseys of teams participating in the RoboCup SPL. This extended data set will be used to train object detection algorithms to improve the recognition of opponents in robotic soccer.

2

Background

2.1 You Only Look Once (YOLO)

You Only Look Once (YOLO) is a family of object detection models, with its first version being presented by Redmon et al. (16) in 2015. The name refers to the algorithm's ability to analyze the entire image in a single pass, unlike traditional object detection algorithms that require multiple passes (17). As shown in Figure 2.1, the input image is passed through a CNN to extract features, which is performed using the DarkNet architecture, a high performance open source neural network framework (18). YOLO then divides the output of this layer into a S×S grid. Each grid cell is responsible for predicting bounding boxes and associated class probabilities for any object contained within its boundaries. Afterwards, a non-maximum suppression (NMS) algorithm is applied to remove identical detections and improve the overall accuracy of the model. This efficient approach allows YOLO to achieve real-time object detection. As of now, the YOLO family consists of 8 models, with YOLOv8 being the most recent addition. With each new version, the model evolves to enhance the accuracy and improve the speed of object detection. The latest developments were seen in YOLOv5 and YOLOv8.

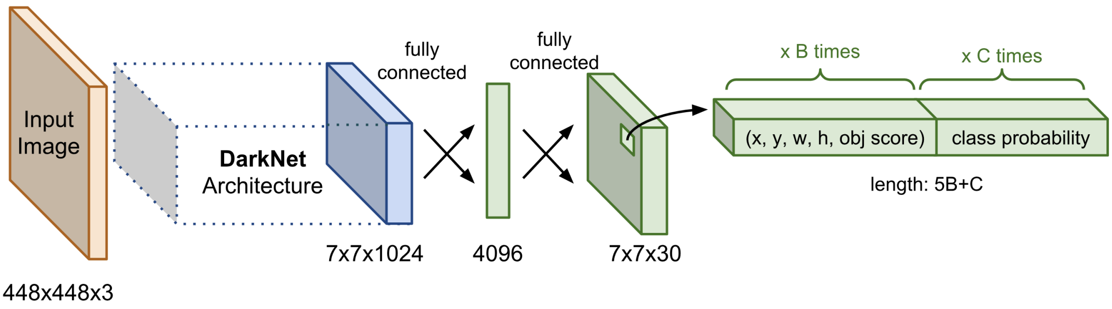


Figure 2.1: The network architecture of the first YOLO model[1]

2.1.1 Comparison YOLOv5 versus YOLOv8

As explained by Selcuk and Serif (17), the YOLOv8 model is essentially the same as YOLOv5. However, there are a few updates made that are worth noting. The first difference is the size of the grids. YOLOv8 uses a 3×3 grid, whereas YOLOv5 uses a 6×6 grid.

Another difference between the YOLOv8 and YOLOv5 models is that the YOLOv8 is an anchor-free model, while the YOLOv5 is an anchor-based model. An anchor-based model uses a predefined set of anchor boxes of different sizes and aspect ratios. The model predicts the location and size of the bounding boxes relative to these anchor boxes. The predicted bounding boxes are then adjusted based on the offset between the anchor boxes and the ground-truth boxes. This approach helps the model accurately detect objects of varying sizes and aspect ratios. On the other hand, an anchor-free model does not use anchor boxes. Instead, it directly predicts the center point and size of the bounding boxes. This method reduces the complexity of the model and eliminates the need for manually defining anchor boxes.

2.2 Performance metrics

Performance metrics play a pivotal role in understanding and evaluating the efficacy of a model, and within the realms of object detection, the following key metrics are provided.

[1] https://learnopencv.com/mastering-all-yolo-models/

2.2.1 Intersection over Union (IoU)

The intersection over union (IoU) metric provides a quantitative measure of the similarity between two bounding boxes, often used to assess the accuracy of object detection algorithms. The IoU is calculated by determining the ratio of the area of overlap (the intersection) between the ground truth and the predicted bounding boxes to the area of union of these boxes.(19)

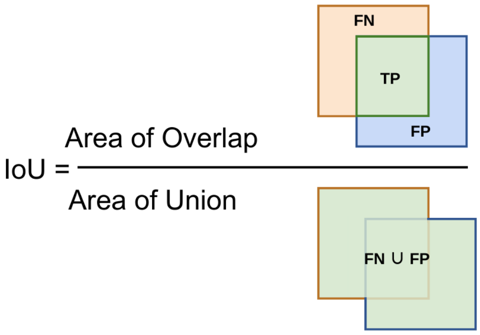


Figure 2.2: IoU is calculated by dividing the area of overlap between the bounding boxes by the area of union, with the ground truth box having an orange border and the predicted box having a blue border.

As shown in figure 2.2, the area of overlap is the blue square in the numerator image. This is the region where an object detection algorithm identifies an object that exactly match the ground truth box. This selection is known as true positives (TP). The area of union in the denominator combines the detection results and then subtracts the true positives. This is to prevent those objects from being double counted. The blue region are objects that were erroneously detected by the model. These are known as false positives (FP). The objects in the orange region, which corresponds with the ground truth box, should have been detected by the algorithm but were missed. These missed pixels are known as false negatives (FN). A perfect prediction is indicated by an IoU value of one, meaning the predicted bounding box precisely aligns with the ground truth bounding box. In that case, FP, TP and FN are all equal to 0. Conversely, an IoU of zero signifies an inaccurate prediction, as there is no overlap between the predicted and ground truth bounding boxes. This can be evaluated as the following equation:

$$\text{IoU} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}}$$

2.2.2 Average Precision (AP)

The average precision (AP) metric describes a model's accuracy at predicting one particular object class. To understand average precision, we will first look at precision. Precision measures the proportion of correctly predicted positive (TP) instances out of all instances (TP + FP) predicted as positive by the model. This can be noted as the following:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

Average precision takes into account precision values at different levels of recall. Recall, on the other hand, measures the proportion of correctly predicted positive instances out of all actual positive instances (true positives) in the dataset. It tells us how well the model can find all positive instances. This is noted as:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

2.2.3 Mean Average Precision (mAP)

The mean average precision (mAP) metric describes a model's accuracy at predicting one particular object class. The mAP calculates the average precision for each class and then averages these values across all classes. It provides a single summary score that reflects the overall performance of the model across multiple classes. The average of the mean average precision calculated at varying IoU thresholds, ranging from 0.50 to 0.95. This is also noted as mAP@[0.5 : 0.95]

3

Method

3.1 Dataset

In this thesis, a curated subset of the extensive dataset from the 2022 RoboCup SPL Video Analysis Challenge, provided by team B-Human, and the RoboCup SPL Jersey dataset, provided by RoboCup are used.

3.1.1 B-Human Dataset

B-Human is a RoboCup SPL team of the University of Bremen and the German Research Center for Artificial Intelligence (DFKI). They have provided a dataset[1], that offers a collection of images captured during RoboCup 2019 soccer games. The dataset was carefully labeled, with each image containing annotations for the ball, all players, their respective jersey colors, and jersey numbers.
The original dataset consisted of 35,000 labeled images, encapsulating the dynamics of the soccer matches. Each image represents a snapshot from the perspective of a camera positioned on soccer fields. The available subset of the dataset showcases 5,000 images. These images show a diverse range of scenarios, including various numbers of NAO robots, footballs, occlusions, and varying lighting conditions.

3.1.2 RoboCup SPL Dataset

The next dataset is provided by the RoboCup SPL organization. This has been retrieved from the open-access git repository[2].
This dataset contains 120 images of 22 teams and will be sufficient to initiate this first phase of the thesis. Per team, the high-resolution images contain the front

[1] https://github.com/bhuman/VideoAnalysis

[2] https://github.com/RoboCup-SPL/Robot-Jerseys

and back view of the jersey, displayed on a flat surface or worn by a NAO robot. The images contain the full front, back or side view of the jerseys. An example of this can be seen in Figure 3.1. This image shows the jersey of team NAO Devils. The magenta jersey on a clear image with low noise background is ideal for object detection due to its distinct color, providing high contrast against the muted background. Also, the absence of distractions reduces interference, allowing the object detector to focus on the jersey. Clear object boundaries in the image contribute to accurate localization, enhancing the model's precision. Overall, these conditions make this a favorable image to use for this thesis.

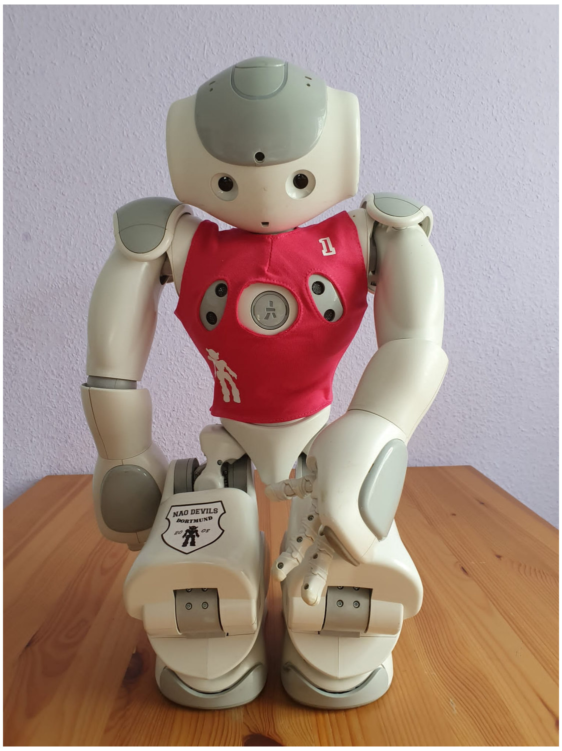


Figure 3.1: Image in the dataset containing the front view of the NAO Devils' magenta jersey.

3.1.3 Preprocessing

To align the dataset with the objectives of our thesis, specific alterations were made. The occurrences of two additional classes present in the SPL dataset, goal post and penalty spot, as were excluded as they are not within the scope of this project.

Another alteration that was made to the dataset was auto-orienting the images. This process involved automatically adjusting the orientation of the images to ensure consistency and ease of analysis. By auto-orienting the images, we aimed to eliminate any potential biases or inconsistencies that could arise from varying image orientations. Furthermore, the images in the RoboCup dataset have been resized to 640×640 format. This step is essential to ensure your dataset is consistent before training the model. As regards to the images in the B-Human dataset, they were consistently obtained from the a single camera. This ensures that the images all had a size of 1920×1080. For YOLO to work optimally, all images were resized to 640×640.

3.1.4 Annotation

For the annotation process, Roboflow [1] was used. This is a computer vision platform that simplifies the process of building models. With its user-friendly interface and robust features, the acquired data can quickly be labeled and exported to any format. This enabled efficient experimentation and iteration. This was experienced during the labeling of the RoboCup dataset. In the case of B-Human's dataset, the original labels were sufficient to train the object detection model to recognize NAO robots and footballs, taking into account their diverse jersey colors and jersey numbers.

3.1.5 Augmentation

Roboflow can also apply augmentation to the data accordingly. The augmentation process ensures that our models are exposed to a diverse yet manageable set of real-world scenarios. An example of this can be seen in Figure 3.2. This balances the need for comprehensive training with computational efficiency. This was applied according to the following settings:

- Outputs per training example: 3
- Flip: Horizontal, Vertical
- 90° Rotate: Clockwise, Counter-Clockwise
- Rotation: Between -15° and +15°
- Hue: Between -15° and +15°
- Saturation: Between -25
- Brightness: Between -20

[1] https://roboflow.com/

- Exposure: Between -15
- Noise: Up to 1.45

This process increased the dataset from 120 images to 288 images. Lastly, a mosaic grid is applied as an extra step of augmentation by YOLO during training. The amount of grids can be adjusted before training. For this thesis, the default setting of randomly sampling 4 images into 1 image was used.

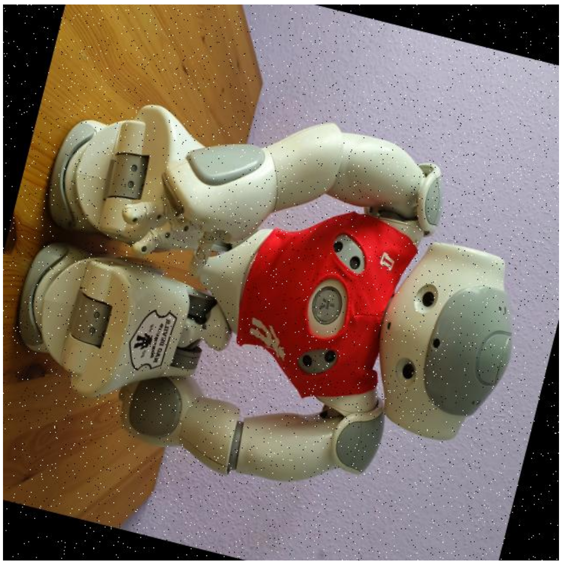


Figure 3.2: Image in the dataset containing the front view of the NAO Devils' magenta jersey. Augmentation has been applied in the form of noise, brightness, saturation and a 105° counter-clockwise rotation.

3.1.6 YOLO formatting

The YOLO format is a specific format for annotating object bounding boxes in images for object detection tasks. In this format, each image in the dataset should have a corresponding text file with the same name as the image, containing the bounding box annotations for that image. In YOLO, a bounding box is represented by five values: the four normalized x and y coordinates of the center of the bounding boxes as well as the object class. To make coordinates normalized, we take pixel values of x and y, which marks the center of the bounding box on the x- and y-axis. Then we divide the value of x by the width of the image and value of y by the height of the image. width and height represent the width and the height of the bounding box. They are normalized as well.

3.2 Training

Both YOLOv5 and YOLOv8 were installed and trained using the Google Colab [1] and Paperspace Gradient[2] platforms, serving as cloud workspaces that run on 8 GB GPUs. Furthermore, any additional dependencies and libraries are also installed to the computer. The RoboCup dataset was randomly split into a training set of 201 images, and test set of 58 images a validation set of 29 images according to a 70/20/10 ratio. Similarly, the B-Human dataset is split into a training set of 3500 images, a test set of 1000 images and a validation set of 500 images. A subset of training set of B-Human is then combined with the training set of RoboCup SPL. This combined dataset in created in such a way that the various teams are equally represented in the data. This brings the total of the training data to 350 images, the test data to 100 images and the validation data to 70 images. This ensures that the training data is large and diverse enough to have normal distribution to avoid biases. Finally, in order to maintain consistency, as in the previous related studies, the researchers opted to utilize the smallest version of YOLOv5 (YOLOv5n). Similarly, in this thesis, YOLOv8's smallest model (YOLOv8n) was used. These models underwent training for 300 epochs on the same computer with identical specifications. Furthermore, a batch size of 16 training images was chosen. After completing the training process, a diverse range of test batch images, matrices, and graphs were acquired.

[1] https://colab.google/

[2] https://docs.paperspace.com/gradient/

4

Experiments

4.1 Results

For the evaluation of our approach, we measure the performance of YOLOv5 versus YOLOv8. The results of the models will be demonstrated through the set of chosen metrics, ensuring a robust evaluation of its performance.

4.1.1 YOLOv5 training

Figure 4.1 shows the result of the YOLOv5 training on the RoboCup SPL dataset after 272 epochs. At this amount of epochs, the model already reached mAP results on all classes of 0.893. To avoid overfitting, the model was stopped early as no improvement was observed in last 100 epochs. The best results were achieved at 172 epochs. The model was then tested on the validation set, which can be seen in Figure 4.1.

4.1.1.1 Accuracy in predicting the jersey classes

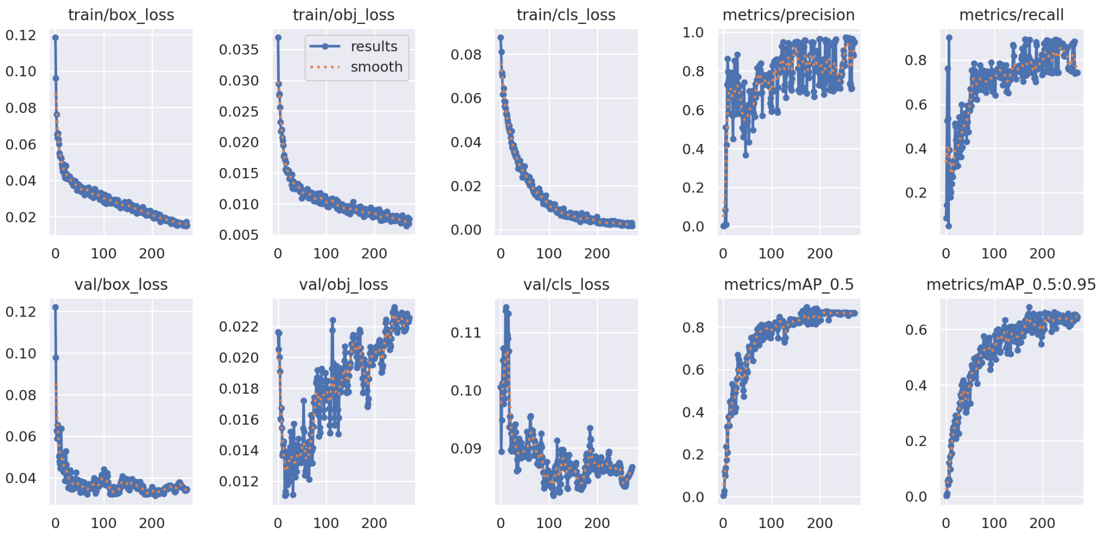


Figure 4.1: Metric results of YOLOv5 training on the combined dataset after 272 epochs with a batch size of 16 training images

YOLOv5 Model Summary						
Class	Images	Instances	Precision	Recall	mAP50	mAP@0.5:0.95
All	24	100	0.819	0.81	0.893	0.681
Austin Villa	24	6	0.755	1	0.995	0.846
B-Human	24	12	1	0	0.0425	0.034
Bembelbots	24	8	0.543	1	0.995	0.645
Berlin United	24	6	0.826	1	0.995	0.549
Camellia Dragons	24	7	0.685	1	0.995	0.895
HTWK	24	9	1	0.894	0.995	0.787
NAO Devils	24	7	0.57	1	0.995	0.895
Naova	24	6	0.876	1	0.995	0.723
Nomadz	24	9	0.788	1	0.995	0.895
R-ZWEI	24	4	1	0	0.995	0.796
Rinobot Jaguar	24	9	1	0.941	0.995	0.66
RoboEireann	24	6	0.855	0.5	0.524	0.312
Tjark	24	6	0.664	1	0.995	0.796
rUNSWift	24	5	0.897	1	0.995	0.697

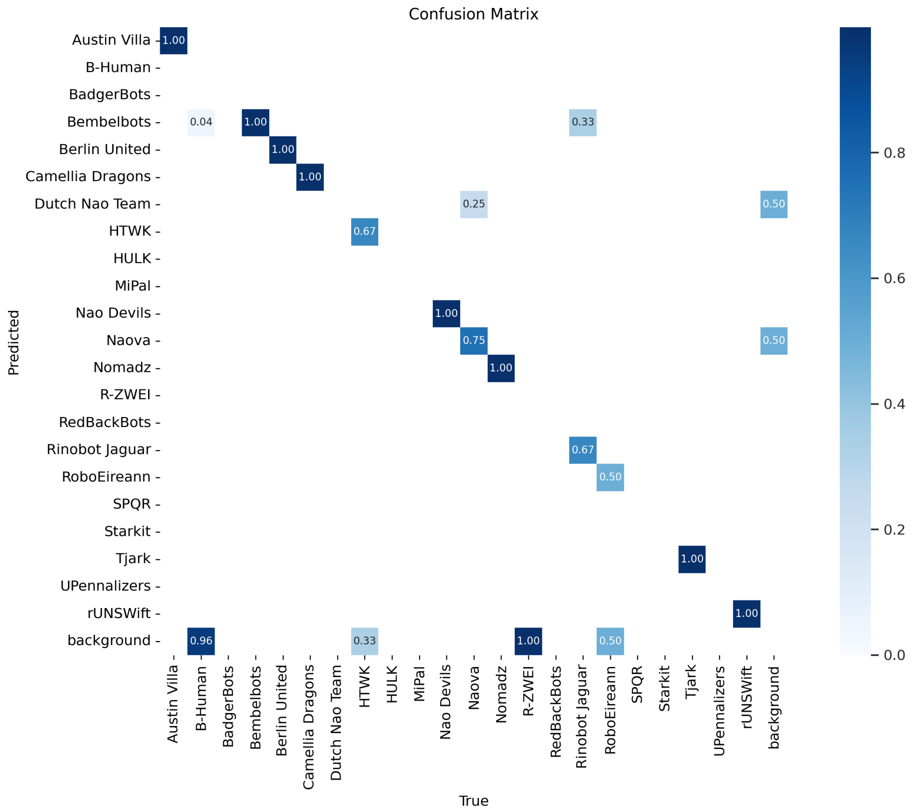


Figure 4.2: Confusion matrix of the performance results of YOLOv5 on the combined dataset, with values from 0.0 until 1.00.

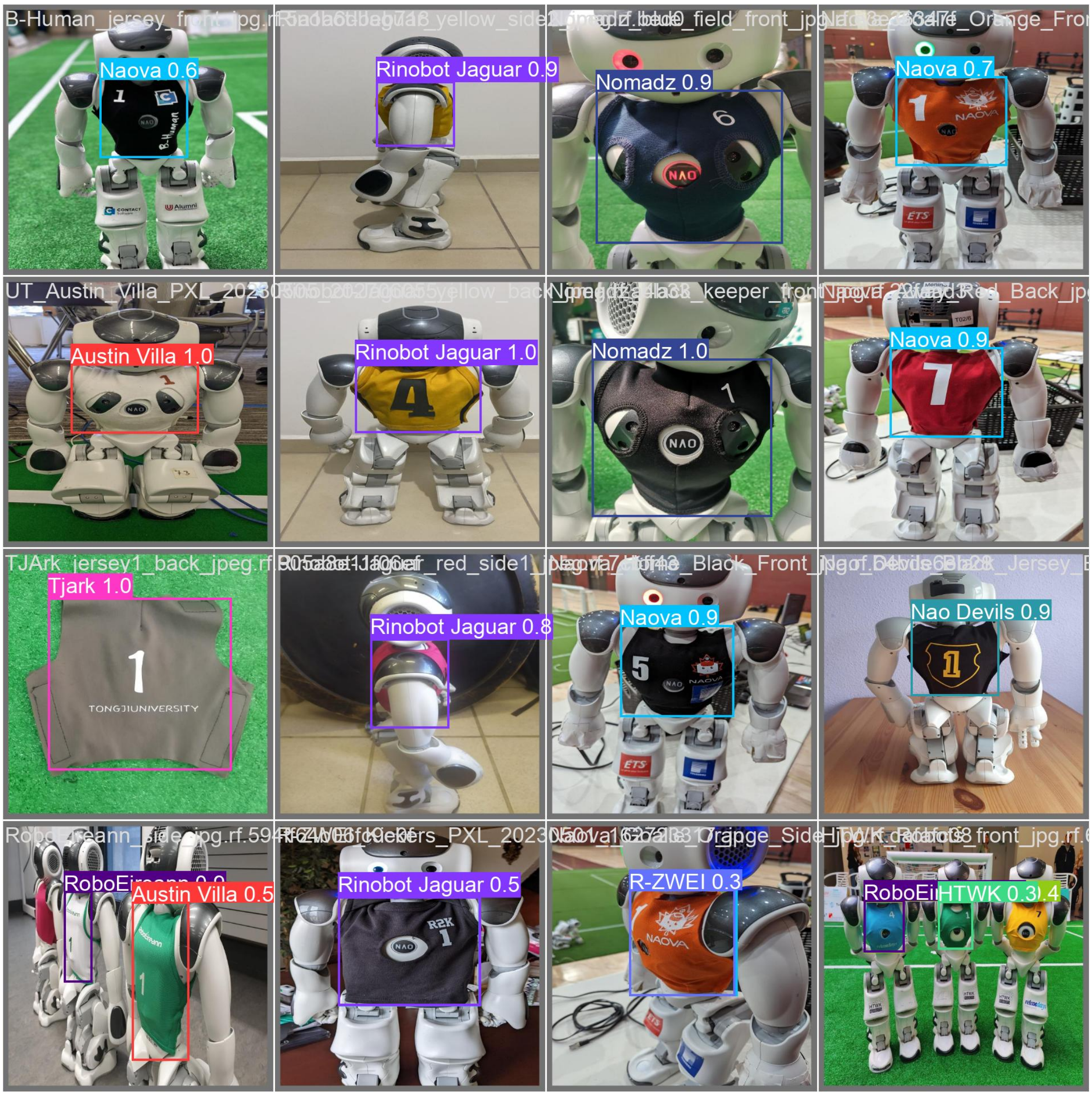


Figure 4.3: Prediction of YOLOv5 on validation set.

The model summary of YOLOv5 indicates that it has very high mAP values for most jersey classes. However, there are a few outliers in terms of precision and recall. Specifically, if we focus on the classes B-Human and R-ZWEI, we can observe that the precision value is 1, indicating that all the detections made for these classes are correct. However, the recall value for these classes is 0, which means that none of the instances of these classes were successfully detected. As a result, the number of TP is also 0, leading to B-Human being incorrectly classified as a FN and R-ZWEI being incorrectly classified as a FP in Figure 4.3. This performance shortfall underscores the need for further investigation into the specific challenges posed by these teams' appearances in the dataset. Furthermore, 9 classes have achieved a recall value of 1. While maximizing recall ensures that all positive instances are captured, it may also result in higher false positive rates, thus lowering precision. In the context of robotic soccer, it is favorable to have high recall values. Especially when dealing with real-time object detection, where decisions have to be made fast and strategically.

4.1.2 YOLOv8 training

Figure 4.5 shows the result of the YOLOv8 training on the dataset after 209 epochs. At this amount of epochs, the model already reached mAP results on all classes of 0.887. Similar to YOLOv5, the model was stopped early as no improvement was observed in last 50 epochs. The best results were achieved at 159 epochs. The model was then tested on the validation set, which can be seen in Figure 4.4.

4.1.2.1 Accuracy in predicting the jersey classes

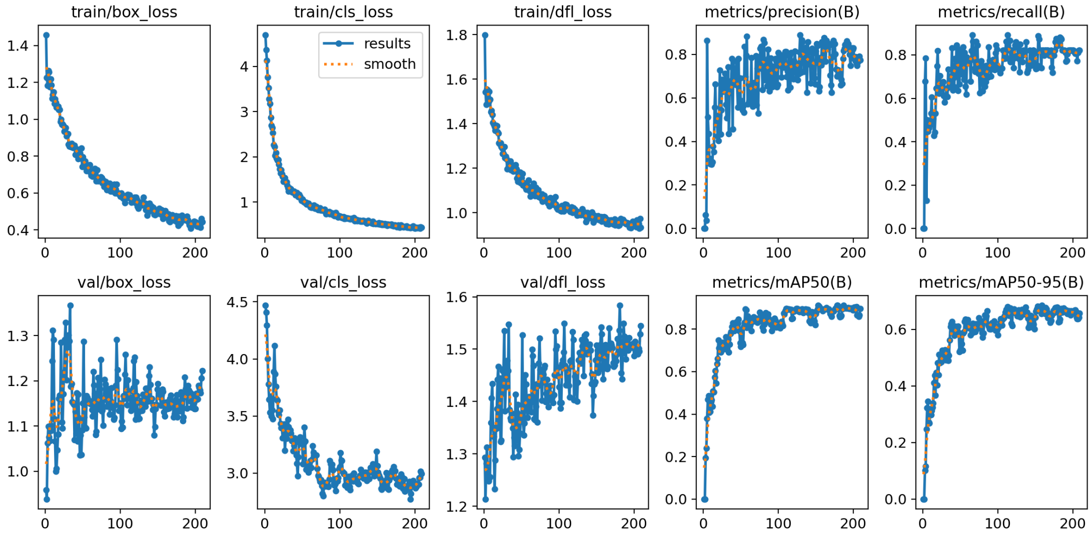


Figure 4.4: Metric results of YOLOv8 training on the combined dataset after 209 epochs with a batch size of 16 training images

YOLOv8 Model Summary						
Class	Images	Instances	Precision	Recall	mAP50	mAP@0.5:0.95
All	24	100	0.773	0.821	0.887	0.685
Austin Villa	24	6	0.698	1	0.995	0.895
B-Human	24	12	0	0	0	0
Bembelbots	24	8	0.953	1	0.995	0.638
Berlin United	24	6	0.868	1	0.995	0.5
Camellia Dragons	24	7	0.78	1	0.995	0.895
HTWK	24	9	1	0	0.83	0.638
NAO Devils	24	7	0.807	1	0.995	0.796
Naova	24	6	0.795	1	0.995	0.854
Nomadz	24	9	0.859	1	0.995	0.921
R-ZWEI	24	4	0.931	1	0.995	0.796
Rinobot Jaguar	24	9	0.807	1	0.995	0.697
RoboEireann	24	6	0.736	1	0.638	0.362
Tjark	24	6	0.788	0.5	0.995	0.697
rUNSWift	24	5	0.803	1	0.995	0.895

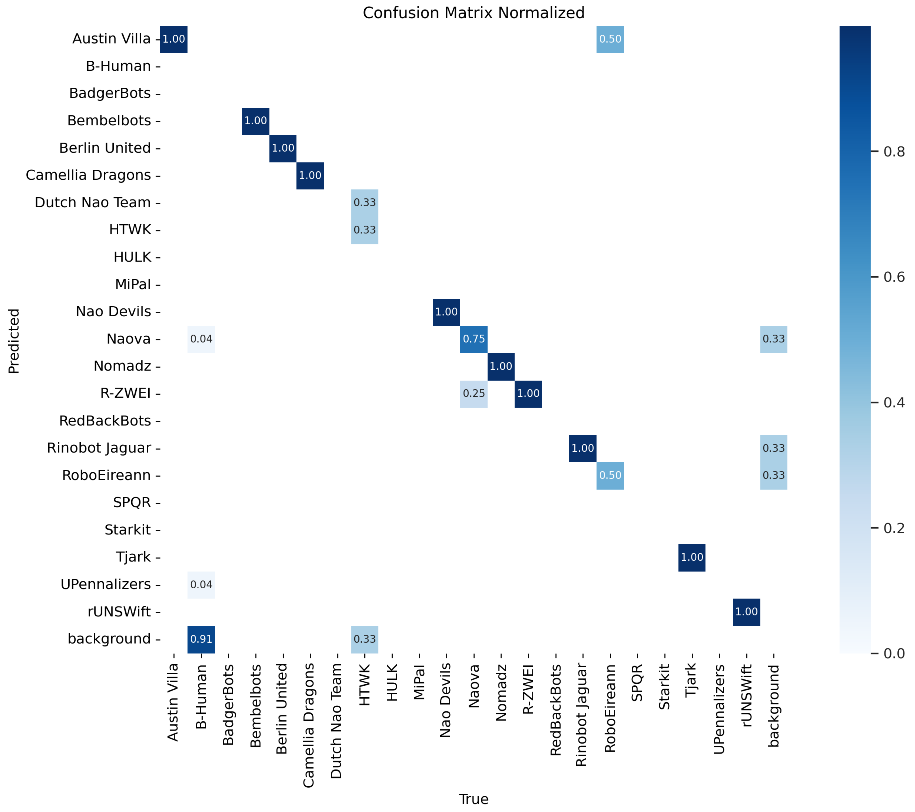


Figure 4.5: Confusion matrix of the performance results on the combined dataset, with values from 0.0 until 1.00.

Now shifting our attention to the model summary of YOLOv8, we find that B-Human scores 0 across all the evaluation metrics. This can also be observed in Figure 4.6, where B-Human is wrongly detected as the Dutch NAO Team, resulting in a TP value of 0. Additionally, HTWK also has a recall value of 0. Looking at the confusion matrix in Figure 4.5, HTWK has a TP of 0.33. This suggests that it has been partially detected but not fully recognized. A visual representation of this can also be seen in Figure 4.6, where the three HTWK robots are pictured in the bottom left corner. Two of them have been detected, but the third one has not (FN). Nevertheless, with YOLOv8, there are 11 classes in total with a recall value of 1. As previously stated, a high recall is favorable in this domain. However, this may cause the model to have a high number of FP predictions, which eventually results in a lower precision. It is therefore necessary to

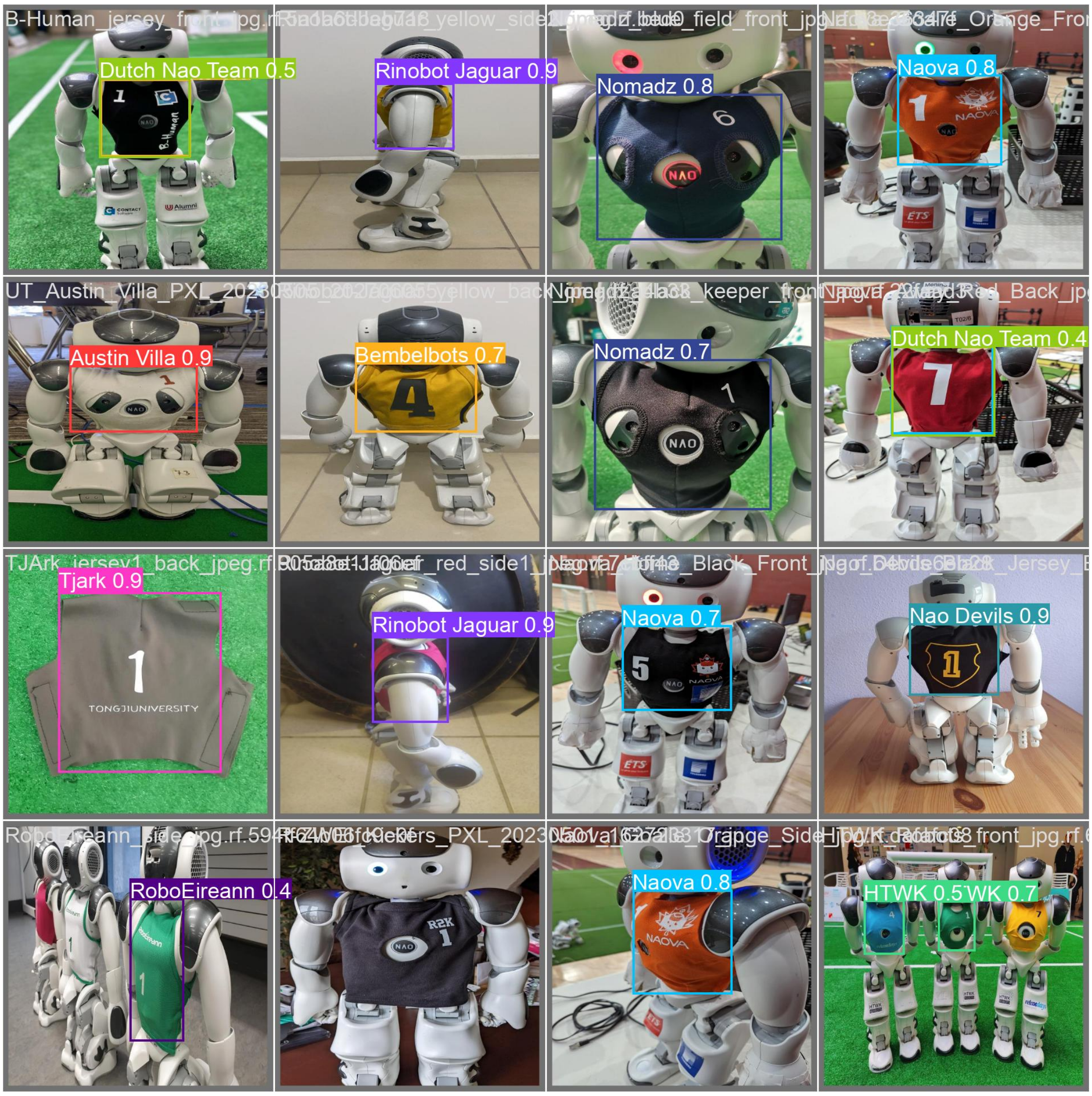


Figure 4.6: Prediction of YOLOv8 on validation set.

5

Discussion

5.1 Conclusion

This thesis aimed to explore the performance of object detection models when trained using real-world datasets. The focus of the thesis was specifically on detecting objects relevant to the RoboCup SPL, with a specific emphasis on recognizing diverse jerseys. The main research question centered around enhancing real-time object detection in robotic soccer-playing systems.

The thesis conducted experiments using YOLOv5 and YOLOv8 models in combination with a curated dataset. The expectation was that the YOLOv8 model would lead to sufficient model performance. However, the results showed that both the YOLOv5 and YOLOv8 models achieved similar results. The overall mAP for object detection at different IoU thresholds, specifically 0.5 to 0.95, was 0.681 for YOLOv5 and 0.685 for YOLOv8. To answer the research question, these experiments show that both the YOLOv5 and YOLOv8 models can achieve comparable results in terms of object detection performance. This can be used to enhance the overall performance of recognizing opponents during a soccer game.

5.2 Comparison with relevant work

As earlier relevant studies have shown, the YOLO framework is a great fit for detecting objects in dynamic environments. While the models show competence in recognizing opponents, the results explain that there are areas for potential improvement. This includes optimizing training strategies, fine-tuning parameters in the models, applying other augmentation techniques and exploring the use of GANs. As previously mentioned, GANs can be used to generate synthetic data that closely resembles real data, which can be used to further improve the models' performance. In our case, the provided datasets were limited so the expectation would be that it can help in reducing bias and improving the generalization ability of the models.

5.3 Aspects

During the evaluation of the results, a few points were noticed. Regardless of the YOLO model, a few teams were incorrectly detected, particularly the B-Human jerseys. The mAP at different IoU thresholds was 0.034 for YOLOv5 and 0 for YOLOv8. These values are significantly low compared to the performance of the other teams and the overall performance of the models. One possible explanation for this low performance is that multiple teams had black jerseys with similar designs, which made it challenging for the model to accurately detect them. However, it is worth noting that the Nomadz jersey was accurately detected with a mAP@[0.5 : 0.95] of 0.895 for YOLOv5 and 0.921 for YOLOv8. Another suggestion that was considered was the number of instances available in the dataset. However, this suggestion was rejected as the B-Human jersey was prominent in the dataset. These findings indicate that further research and analysis are needed to understand and address the issues observed.

5.4 Future Work

For this thesis, a small selection of YOLO models were used. As stated in the method, the purpose of using these models was to maintain consistency. Knowing that the YOLO family consists of eight model types, each with its own unique characteristics and capabilities, it is worth mentioning that a ninth version of YOLO, called YOLOv9, is expected to be released in the near future. It is therefore important to demonstrate their generalizability to other YOLO models in future work. This means that further research and experimentation should be conducted to assess how well the selected models perform when applied to different YOLO variants. By doing so, we can gain a better understanding of the strengths and limitations of these models. Another suggestion for a future thesis would be to combine synthetic data with real-world data. This can provide a more comprehensive and diverse dataset for analysis and modeling, as it allows for a more robust representation of the underlying patterns and trends in the data.

6

Acknowledgements

I would like to express my sincere gratitude to my family, friends and my supervisor, dr. Arnoud Visser, for their invaluable support and encouragement throughout the completion of this thesis. Their guidance, feedback, and encouragement have been instrumental in creating this work.

References

[1] Rodrigo Verschae and Javier Ruiz-del Solar. **Object Detection: Current and Future Directions**. *Frontiers in Robotics and AI*, **2**, 2015. 4

[2] *Real-Time Motion Detection and Surveillance using Approximation of Image Pre-processing Algorithms.* IEEE, 2019. 4

[3] **Computer Vision Application Analysis based on Object Detection**, 2023. 4

[4] Qiang Bai, Shaobo Li, Jing Yang, Qisong Song, Zhiang Li, and Xingxing Zhang. **Object Detection Recognition and Robot Grasping Based on Machine Learning: A Survey**. *IEEE Access*, **8**:181855–181879, 2020. 4

[5] *Computer Vision.* Apress eBooks, 2023. 4

[6] **Autonomous robot soccer matches**. AmsterdamVrije Universiteit, Department of Computer Sciences, 2016. 4

[7] *An Embedded Monocular Vision Approach for Ground-Aware Objects Detection and Position Estimation.* Springer eBooks, 2023. 4

[8] **Deep-Learning-Based Context-Aware Multi-Level Information Fusion Systems for Indoor Mobile Robots Safe Navigation**, 2023. 4

[9] Alexander Fabisch and Tim Laue. **Robot Recognition and Modeling in the RoboCup Standard Platform League**. 2010. Not an article, but @inproceedings. How published? 6

[10] Xu Zhang, Hanbin Wang, and Qijun Chen. **Evaluation of color space for segmentation in robot soccer**. In *2014 IEEE International Conference on System Science and Engineering (ICSSE)*, pages 185–189, 2014. 6

[11] Jacob Menashe, Josh Kelle, Katie Genter, Josiah Hanna, Elad Liebman, Sanmit Narvekar, Ruohan Zhang, and Peter Stone. **Fast and Precise Black and White Ball Detection for RoboCup Soccer**. In Hidehisa Akiyama, Oliver Obst, Claude Sammut, and Flavio Tonidandel, editors, *RoboCup 2017: Robot World Cup XXI*, pages 45–58, Cham, 2018. Springer International Publishing. 6

[12] SR Sun. **Faster R-CNN: towards real-time object detection**. *Advances in Neural Information Processing Systems (NIPS)*, 2015. 6

[13] Francisco Leiva, Nicolás Cruz, Ignacio Bugueño, and Javier Ruiz del Solar. **Playing Soccer without Colors in the SPL: A Convolutional Neural Network Approach**, 2018. 6

[14] Bernd Poppinga and Tim Laue. **JET-Net: Real-Time Object Detection for Mobile Robots**. In *Robot Soccer World Cup*, 2019. Not an article, but @inproceedings. How published? 7

[15] Hidde G.J. Lekanne gezegd Deprez. **Enhancing simulation images with GANs**, 2020. Not an article, but thesis (@misc). How published? 7

[16] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. **You Only Look Once: Unified, Real-Time Object Detection**, 2016. 8

[17] Burcu Selcuk and Tacha Serif. **A Comparison of YOLOv5 and YOLOv8 in th Context of Mobile UI Detection**. In Muhammad Younas, Irfan Awan, and Tor-Morten Grønli, editors, *Mobile Web and Intelligent Information Systems*, pages 161–174, Cham, 2023. Springer Nature Switzerland. 8, 9

[18] Joseph Redmon and Ali Farhadi. **Yolov3: An incremental improvement**. *arXiv preprint arXiv:1804.02767*, 2018. 8

[19] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. **Generalized Intersection over Union: A Metric and A Loss for Bounding Box Regression**, 2019. 10

7

Appendix

List of Figures