

Comparison of Detection Models for Unconstrained Objects for Humanoid Robots

Ruth Moreno, Marco Negrete*, Luis Contreras

Universidad Nacional Autónoma de México (UNAM),
Instituto de Ingeniería, Mexico City,
Mexico

marco.negrete@ingenieria.unam.edu

Abstract. Localization for soccer-playing humanoid robots is essential for planning game strategies. Visual localization is commonly achieved by detecting landmarks of the soccer-field, nevertheless, this is difficult due to disturbances caused by motion blur during gait. Moreover, landmarks can have multiple valid bounding boxes that makes difficult the labeling for training. In this work, we compare three state-of-the-art methods for detecting landmarks: YOLO, RT-DETR and Deformable-DETR. We used the TORSO-21 dataset, augmented with data taken during Robocup Humanoid League 2025. We compare the performance using precision, recall and the angular localization error. Our results showed that Deformable-DETR overcome RT-DETR and YOLO in precision but not in recall for all classes. Nevertheless, Deformable-DETR has the lowest angular localization error for all classes but also the highest computing time consumption.

Keywords. Detection models, unconstrained objects, humanoid robots.

1 Introduction

Autonomous humanoid robots require robust localization to navigate in dynamic environments [7]. The oscillating nature of a humanoid robot, due to gait patterns, makes perception a difficult task. Lidar-based localization can be challenging due to the need of correcting measurements according to robot's body motion and camera-based algorithms face the problem of motion blur [1].

In soccer-playing robots, detecting field landmarks such as T, L and X shapes, combined with Bayesian filters, has demonstrated to be a

successful localization method [10], nevertheless, detecting such shapes poses a new problem: these type of landmarks are *unconstrained*, i.e., they cannot be accurately contained in a bounding box. In a soccer field, it is difficult to determine where a L-shape ends.

Many object detection models require a labeled dataset with bounding boxes, nevertheless, with unconstrained objects, such bounding boxes are not well-defined. In recent years, Vision Transformers have become an alternative for detecting objects. These models are referred to as DETR (DEtection TRansformers) [4].

In this work, we compare three state-of-the-art object detection models: YOLOv8 [6], Deformable DETR [16] and RT-DETR [9], focusing on their ability to provide spatially accurate landmark observations rather than precise object extents. We argue that center-based localization error is a more appropriate metric for downstream localization tasks than Intersection over Union (IoU-based) measures. In section 2, we review the related work. Section 3 details the object detection approaches for unconstrained landmarks. Section 4 describes the model training processes. Section 5 presents the experimental results and computational benchmarks, and Section 6 concludes the paper.

2 Related Work

Robust self-localization is a fundamental capability for humanoid robots navigating in dynamic

environments, such as the RoboCup Humanoid Soccer League [12]. A robust localization enables accurate navigation, decision making, and interaction with the environment [13].

In humanoid soccer, localization is particularly challenging due to perceptual ambiguity, frequent occlusions, and the absence of distinctive geometric landmarks. Vision-based localization has been extensively studied in the context of humanoid and mobile robots [13].

Classical approaches rely on geometric feature extraction from field lines and goals, followed by probabilistic state estimation using Kalman filters or particle filters [14].

These methods often assume clean segmentation and well-defined landmark geometry, assumptions that rarely hold in real-world soccer environments [2].

Unlike rigid objects, soccer field landmarks are *unconstrained*; i.e., they do not possess well-defined physical boundaries and their appearance varies significantly with viewpoint and illumination. As a result, standard object detection evaluation metrics based on bounding box overlap may not adequately reflect their usefulness for localization.

Using an appropriate performance metric allows to choose an appropriate model for landmark detection and thus, achieving robust localization.

Recent advances in deep learning have enabled robust object detection under challenging conditions. Single-stage detectors such as YOLO have been widely adopted in robotic perception due to their real-time performance. However, their reliance on anchor-based bounding boxes makes them sensitive to scale variations and annotation inconsistencies, particularly for elongated or non-rigid landmarks.

Transformer-based detectors, including DETR and its variants, address some of these limitations by formulating detection as a set prediction problem using attention mechanisms [15, 5].

Deformable DETR [16] further improves convergence and localization accuracy by attending to a sparse set of relevant image locations, making it well suited for detecting structured but unconstrained landmarks such as field markings.

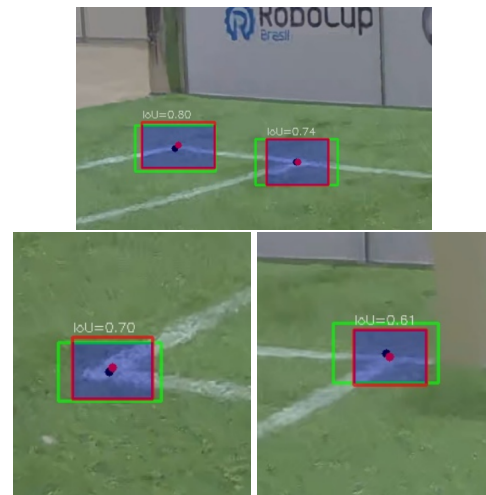


Fig. 1. Example of *unconstrained* landmarks: there are multiple valid bounding boxes

3 Object Detection for Unconstrained Landmarks

Field landmarks such as line intersections and field markings do not correspond to objects with fixed geometry. Their visual appearance is defined by contrast and topology rather than physical extent.

Consequently, bounding box annotations are inherently ambiguous, and multiple valid annotations may exist for the same landmark. Figure 1 shows examples of *L* and *T* shapes in a soccer field. Those landmarks are unconstrained since there can be multiple valid bounding boxes for the same pattern.

To address this challenge, we evaluate object detection models based on their ability to produce consistent and spatially accurate landmark centers. This evaluation criterion aligns with the requirements of the downstream localization task.

To assess the suitability of modern object detectors for humanoid robot localization, we evaluate three representative architectures: YOLOv8, Deformable-DETR, and RT-DETR. These models were selected to cover both anchor-based and transformer-based detection paradigms, as well as different trade-offs between accuracy and computational efficiency.

YOLOv8 represents a widely adopted single-stage detector optimized for real-time performance, while RT-DETR introduces a lightweight transformer-based alternative designed for efficiency. Deformable DETR serves as a high-accuracy transformer-based detector with deformable attention mechanisms that improve spatial localization.

3.0.1 YOLOv8

YOLOv8, developed by Ultralytics, represents one of the most popular models of the "You Only Look Once" family of single-stage object detectors. In contrast to the set-prediction approach of Deformable DETR, YOLOv8 utilizes a deep convolutional neural network (CNN) architecture based on a modified CSPDarknet backbone and a decoupled detection head [8].

This architecture is specifically optimized for high-throughput inference, making it a relevant choice for real-time perception in the RoboCup Humanoid League [11].

A key distinction of YOLOv8 is its anchor-free design. Unlike previous iterations that relied on predefined anchor boxes, YOLOv8 directly predicts the center of an object and the distances to the bounding box edges. This approach simplifies the loss function and improves the model's ability to generalize across different aspect ratios.

For humanoid localization, the efficiency of YOLOv8 allows the vision pipeline to operate at high frame rates, which is crucial for maintaining an up-to-date belief state in the Monte Carlo Localization (MCL) filter.

However, as a CNN-based model, YOLOv8 relies on local receptive fields, which may limit its ability to capture long-range spatial dependencies compared to the attention mechanisms found in Transformers.

In the context of unconstrained landmarks, this can result in higher variance in center-point estimation when landmarks are partially occluded or subject to significant perspective distortion.

3.0.2 RT-DETR

The Real-Time DETection TRansformer (RT-DETR) [9], was the first object detector based on transformers that achieved real-time performance, overcoming the YOLO family both in speed and precision. The RT-DETR keeps the idea of an end-to-end detector but it introduces a hybrid and efficient coder, together with a convolutional backbone, to reduce computational cost. This model also allows to change the layers in the coder, without the need of retraining, to improve inference speed, what makes it a relevant alternative for vision pipelines, although it still uses processing and memory resources typically greater than CNN-based models, such as YOLO family.

3.0.3 Deformable DETR

The original DETR framework, introduced by Carion et. al. in 2020 [4], reformulated the object detection as a problem of set prediction using a coder-decoder transformer trained end-to-end. At the beginning, it was poorly adopted due to the slow convergence (it required 500 epochs approximately) and the difficulty to detect small objects, both problems derived from the use of dense attention in the feature maps.

Deformable DETR extends the original DETR framework by introducing deformable attention modules that attend to a sparse set of sampling points around reference locations [16]. This design significantly improves convergence speed and localization accuracy while maintaining the advantages of transformer-based set prediction. Unlike anchor-based detectors, Deformable DETR does not rely on predefined box priors, making it less sensitive to scale variations in the training data. This property is particularly beneficial for detecting unconstrained landmarks whose bounding box sizes vary widely across annotations.

4 Model Training

The three tested models (YOLOv8, RT-DETR and Deformable-DETR) were trained with a dataset composed of images from the TORO-21 repository and images taken during Robocup

Table 1. Training hyperparameters for YOLOv8

Parameter	Value
Base model	YOLOv8s
Epochs	150
Early stopping patience	100
Batch size	16
Input resolution	640×640
Optimizer	Auto (SGD)
Initial learning rate	0.01
Final learning rate	0.01
Layer freezing	None

Table 2. Training hyperparameters for RT-DETR

Parameter	Value
model	rtdetr-l.pt
epochs	150
patience	100
batch	8
resolution	640
workers	8
pretrained	True

Table 3. Training hyperparameters for the Deformable-DETR model

Parameter	Value
backbone	ResNet-50
encoder	6 layers
decoder	6 layers
object queries	300
pretrained	MS-COCO 2017
fine-tuning	full
epochs	200
lr	10^{-4} a 10^{-8}

2025. Although these three models share the dataset and the detection goal, they differ in the architecture and training tools. In the following sections we describe the dataset and the steps to train each model.

4.1 Dataset Construction

The dataset used for training the three models was constructed using the TORISO-21 repository and images taken by the authors during Robocup

2025. The TORISO-21 repository (Typical Objects in Robocup Soccer 2021) is a dataset specifically designed as a benchmark for vision systems in the context of the Robocup Soccer Humanoid League [3]. It is composed of images of the real field, taken from participating robots, as well as generated images using the Webots simulator. For this proposal, we kept only the images that fulfilled two criteria: (i) they were images of the real world and (ii) they contained only landmarks among five classes of interest: *ball*, *goal*, *L-intersection*, *T-intersection*, and *X-intersection*.

To enrich the dataset, we incorporated 300 images taken by the Pumas team during Robocup 2025. The final dataset consisted of 8900 images distributed in two sets for training and testing with 80% and 20% respectively.

4.2 Training Parameters

To train the YOLOv8 model, we used the Ultralytics YOLO framework. The base model was the *small YoloV8*, trained over the COCO dataset. A full fine-tuning was performed using the dataset described in section 4.1 with the hyperparameters listed in table 1.

Similar to YOLOv8, RT-DETR has a training tool provided by Ultralytics. The data augmentation included random horizontal flip ($p = 0.5$), scaling ($s = 0.5$), translation ($t = 0.1$), changes in the HSV color space ($\Delta_{hsv} = [0.015, 0.7, 0.4]$), and random erasing ($e = 0.4$). Table 2 contains the training hyperparameters for the RT-DETR model.

The Deformable-DETR model was trained using the repository published by Zhu et al.[16]. This model combines a convolutional backbone with a encoder-decoder transformer based on multi-scale deformable attention.

Its architecture has four main modules: (i) ResNet-50 as a backbone to extract hierarchic visual features; (ii) deformable 6-layer encoder to apply multiscale attention to the backbone features; (iii) deformable 6-layer decoder to iteratively refine the box and class predictions through self and cross-attention to the coded features and (iv) a detection head to produce the final predictions for the five classes in the dataset (ball, goal, T-intersection,

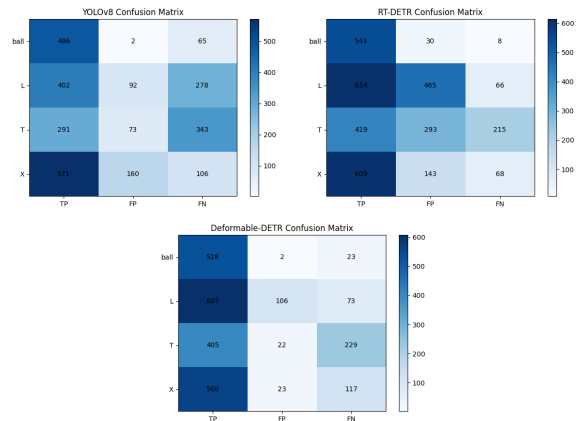


Fig. 2. Confusion matrices for the three trained models



Fig. 3. Example of detected landmarks with the Deformable-DETR model

L-intersection and X-intersection). Table 3 shows the hyperparameters used for training the Deformable-DETR model.

5 Results

We tested the trained models using the 20% of the dataset (see section 4.1). To consider a prediction as a correct detection, we used the well-known IoU metric with a threshold of 0.5. The precision and recall metrics were calculated using this threshold. Figure 2 shows the confusion matrices for the three trained models.

Since we are detecting unconstrained landmarks, IoU is not enough for assessing the performance of a model. Since these landmarks are intended to be used together with a particle

filter for robot localization, an important variable is the correct prediction of the center of the landmark, rather than the correct shape of its bounding box. Considering this, we used the *angular localization error*. This error corresponds to the difference, in pixels, between the predicted horizontal center and the corresponding horizontal center of reference. This error is called angular because of the pinhole camera model. After distortion is corrected in an image, there is a linear relation between horizontal pixels and the horizontal field of view, which can be determined using the intrinsic camera calibration parameters. Thus, the pixel error directly reflects the angular deviation. In this work, we report the mean μ and standard deviation σ of such error in pixels. Figure 3 shows an example of the detected landmarks using the Deformable-DETR model and Table 4 summarizes all the metrics for the three tested models.

5.1 Computational Performance and Real-Time Feasibility

To evaluate the real-time feasibility of the proposed models, inference times were measured under identical hardware conditions. The benchmark was executed on a system equipped with an NVIDIA GeForce RTX 3050 GPU (6GB VRAM) running CUDA 12.8, processing a continuous video stream at a 640×640 resolution.

As summarized in Table 5, YOLOv8 achieved the fastest inference speed at 83.82 FPS (11.93 ms/frame), making it highly suitable for the high-frequency control loops required in the RoboCup Humanoid League. RT-DETR offered an intermediate computational performance of 19.81 FPS (50.48 ms/frame). Conversely, Deformable-DETR recorded the highest computational cost, operating at 10.78 FPS (92.76 ms/frame).

While Deformable-DETR yields the highest spatial accuracy and the lowest angular error for intersections, its heavy processing requirements present a significant challenge. Since deployment on a humanoid robot relies on resource-constrained embedded systems, real-world inference times during a match are expected to be higher.

Table 4. Detection metrics by class and model. μ_{err} and σ in pixels

Class	YOLOv8			
	Precision	Recall	σ	μ
Ball	0.996	0.882	2.48	3.47
Goal	0.897	0.625	3.76	5.41
T	0.799	0.459	3.73	4.77
L	0.814	0.591	3.74	4.72
X	0.781	0.843	4.21	5.81
Class	Deformable DETR			
	Precision	Recall	σ	μ
Ball	0.996	0.958	2.60	2.44
Goal	0.960	0.833	4.49	3.11
T	0.948	0.639	1.74	0.47
L	0.851	0.893	2.29	0.58
X	0.961	0.827	2.07	0.46
Class	RT-DETR			
	Precision	Recall	σ	μ
Ball	0.948	0.985	2.56	3.56
Goal	0.926	0.825	4.15	6.00
T	0.588	0.661	4.03	5.02
L	0.569	0.903	3.71	5.22
X	0.810	0.900	4.09	5.38

Table 5. Real-time inference performance by model (Evaluated at 640×640 resolution)

Model	Average Inference Time (ms)	FPS
YOLOv8	11.93	83.82
RT-DETR	50.48	19.81
Deformable-DETR	92.76	10.78

This emphasizes a critical trade-off: while Deformable-DETR provides highly precise observations, its lower frame rate reduces the update frequency of the perception pipeline, which could impact the robot's overall state estimation and tracking during rapid gait motions.

6 Conclusions and Future Work

Results show that Deformable-DETR has the smallest angular localization errors for T, L and

X-intersections. YOLOv8 presents the highest precision for the ball class, but the slowest recall for the T-intersections, which means a high rate of omissions. RT-DETR has the highest recall for the ball and L-intersection classes, although with smaller precisions in all three intersection classes than the other two models, which indicates a higher rate of false detections. Considering that this detections are intended to be used together with a localization system, recall is more important than precision, since a non detected mark does not give information for localization, while a false detection can be handled by the weights of the observation model.

In general, the Deformable-DETR showed the best performance although it was also the most processing time consuming model. With respect to time of development, YOLOv8 and RT-DETR were easier to train due to the availability of out-of-the-box tools.

As future work, we propose to test more models and to integrate the detected landmarks with a particle filter. Among the suggested models to test is the Segment Anything Model 3 (SAM-3), a prompt based detector recently released by Meta. Also, an important issue is the possibility to deploy each model in an embedded system such as Nvidia Jetson or similar, since this is important to test the usability in a real humanoid robot. Finally, we propose to measure the power consumption of each model once implemented in an embedded system, since visual localization can impact significantly in the battery duration.

Acknowledgments

This work was supported by UNAM-DGAPA under grant PAPIIT-IN118226.

References

1. Ahn, S., Yoon, S., Hyung, S., Kwak, N., Roh, K. S. (2012). On-board odometry estimation for 3D vision-based SLAM of humanoid robot. 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, pp. 4006–4012.

2. **Almeida, A. C., Costa, A. H., Bianchi, R. A. (2017).** Vision-based monte-carlo localization for humanoid soccer robots. 2017 Latin American robotics symposium (LARS) and 2017 Brazilian symposium on robotics (SBR), IEEE, pp. 1–6.
3. **Bestmann, M., Engelke, T., Fiedler, N., Gldenstein, J., Gutsche, J., Hage, J., Vahl, F. (2021).** Torso-21 dataset: Typical objects in robocup soccer 2021. In Robot World Cup. Springer, pp. 65–77.
4. **Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S. (2020).** End-to-end object detection with transformers. European conference on computer vision, Springer, pp. 213–229.
5. **Dosovitskiy, A. (2020).** An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929.
6. **Hidayatullah, P., Syakrani, N., Sholahuddin, M. R., Gelar, T., Tubagus, R. (2025).** YoloV8 to yolo11: A comprehensive architecture in-depth comparative review. arXiv preprint arXiv:2501.13400.
7. **Hong, W., Zhou, C., Tian, Y. (2009).** Robust monte carlo localization for humanoid soccer robot. 2009 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, IEEE, pp. 934–939.
8. **Jocher, G., Chaurasia, A., Qiu, J. (2023).** Ultralytics YOLO.
9. **Lv, W., Zhao, S., Fang, S., Cheng, B., Qiu, J., Singer, J., Cui, C. (2024).** DETRs beat YOLOs on real-time object detection. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).
10. **Nagi, I., Adiprawita, W., Mutijarsa, K. (2014).** Vision-based monte carlo localization for robocup humanoid kid-size league. 2014 13th International Conference on Control Automation Robotics & Vision (ICARCV), IEEE, pp. 1433–1438.
11. **Reis, D., J. Silva, V., Lau, N., Reis, L. P. (2019).** Deep learning for soccer robotics. 2019 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC), IEEE.
12. **RoboCup Federation (2026).** RoboCup Humanoid League official website. <https://humanoid.robocup.org/>.
13. **Siegwart, R., Nourbakhsh, I. R., Scaramuzza, D. (2011).** Introduction to autonomous mobile robots. MIT press.
14. **Thrun, S., Fox, D., Burgard, W., Dellaert, F. (2001).** Robust Monte Carlo localization for mobile robots. Artificial intelligence, Vol. 128, No. 1-2, pp. 99–141.
15. **Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., Polosukhin, I. (2017).** Attention is all you need. Advances in neural information processing systems, Vol. 30.
16. **Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J. (2020).** Deformable detr: Deformable transformers for end-to-end object detection. arXiv preprint arXiv:2010.04159.

Article received on 11/08/2026; accepted on 11/09/2026.

**Corresponding author is Marco Negrete.*