Tech United Eindhoven Team Description 2025

A.S. Deogan, D.M.J. Hameeteman, S.T. Kempers, R.M. Beumer, J.J. Olthuis, W.H.T.M. Aangenent, A. Anil, W. Beks, W. Bocian, L. Blommers, R. van den Bogaert, P.E.J. van Brakel, M. Briegel, D.J.H. Bruijnen, H.L. Chen, A. Chiscuzzu, S. Doodeman, Y.G.M. Douven, A. Guillot, G.P.M. van Hattum, D. Joshi, K.A. Knapik, I. Kolodko, A. Krishnaswamy, G.M. Langhout, H.C.T. van de Loo, T. Nisslbeck, F.B.F. Schoenmakers, S. Subramanian, P. Teurlings, and M.J.G. van de Molengraft

Tech United Eindhoven
De Rondom 70, P.O. Box 513 5600 MB Eindhoven, The Netherlands
techunited@tue.nl
https://www.techunited.nl/

Abstract. The Tech United Eindhoven Middle-Size League (MSL) has become world-champion eight times, including at RoboCup 2024. Moreover, the team achieved first place in the Technical Challenge and second place in the Scientific Challenge at RoboCup 2024. In the past year, the team has made considerable developments in various areas. In this work, we highlight the automatic dataset generation for training vision networks, incorporating algorithms that calibrate the robots automatically, and working towards a quadrupedal goalkeeper. This paper describes how these developments will contribute to Tech United's goal of again becoming world-champion at RoboCup 2025.

Keywords: RoboCup Soccer · Middle Size League · AI Vision · Automatic Calibration · Quadruped

1 Introduction

Tech United Eindhoven represents Eindhoven University of Technology (TU/e) in the RoboCup competition. The team joined the Middle Size League (MSL) in 2006 and played in 14 finals of the world championship, winning it eight times. The MSL team consists of 6 PhD, 2 MSc, 2 BSc and 5 former TU/e students. Furthermore, we have 8 TU/e staff members and 2 members not related to TU/e. This paper describes the major scientific improvements of the Tech United soccer robots over the past year and elaborates on some of the main developments for RoboCup 2025.

This paper starts with a description of the fifth generation soccer robots to be used during the RoboCup 2025 competition in Section 2. Then, we present two developments to increase the level of autonomy of our robots. Section 3 explores the methodology for generating synthetic datasets to train convolutional neural network-based object detection algorithms, removing the human labelling steps. Section 4 discusses strides made towards enabling robots to perform self-calibration, reducing the need for human involvement in match preparation.

Advancements made towards deploying the MIT Mini Cheetah quadruped robot as a goalkeeper are highlighted in Sect. 5. Finally, Sect. 6 presents the conclusion of this paper.

2 Robot Platform

The Tech United soccer robots are called TURTLEs, which is an acronym for Tech United Robocup Team: Limited Edition. Their development started in 2005, and through years of experience and numerous improvements they have evolved into the fifth generation TURTLE, shown in Fig. 1.

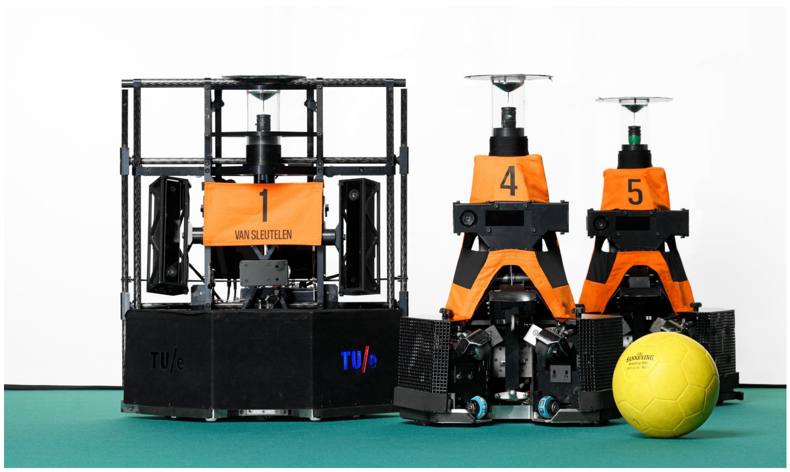

Fig. 1: Fifth generation TURTLE robots, with the goalkeeper on the left-hand side. (Photo by Bart van Overbeeke)

The software controlling the robots consists of four modules: Vision, World-model, Strategy, and Motion. These modules communicate with each other through a real-time database (RtDB) designed by the CAMBADA team [2]. Further details on the software architecture can be found in [6].

3 Synthetic Data Generation

Due to the increasing progression of AI vision capabilities, more MSL teams are adopting this method for object detection. Convolutional Neural-Networks (CNNs) commonly outperform traditional methods whilst providing robust detection for a wide range of objects [5]. These networks are tuned by feeding them with many annotated images (on the order of thousands for small datasets). For

MSL, the model uses annotations of the robots and balls to learn and recognize patterns and features in the labelled images. Currently, teams manually capture and annotate images, which is a long and gruelling process prone to error.

Existing labelled datasets can be used for training, drastically speeding up the annotation process. Methods that employ existing CNN models to label new datasets cannot be used in MSL due to the lack of publically available, robust models capable of detecting MSL robots. Therefore, we investigate synthetic dataset generation to rapidly and automatically generate vast annotated datasets.

3.1 Simulation Environment

Figure 2 shows the environment for the generation of synthetic data. Unreal Engine is used as the simulation environment, LUMA AI is used for the generation of photorealistic 3D Gaussian splat (3DGS) models [4]. 3DGS's provide photorealistic models capable of real-time rendering, enabling rapid generation of large-scale, diverse synthetic datasets.

Fig. 2: Example of the virtual environment, including the field and goals, the lights, the background augmentation, and 3D Gaussian splats of balls and robots.

For the generation of representative synthetic datasets, it is critical to identify the factors that need to closely mimic the real-world. To ensure that the network trains on the actual features of the objects, the most important factor is photorealism of the 3D models of the objects. These photorealistic models are created by capturing images of the object, where only a normal smartphone camera is required, and feeding them into LUMA AI, which generates the 3D models within an hour [4].

The main advantage of this method is the ease of introducing domain randomization in the dataset. Changing the lighting conditions, object orientations, lens parameters, and camera orientations are standard features in many simulators. Applying a uniform distribution, with predefined bounds, to these variations provides an automated method of gathering synthetic images. Furthermore, the calculations of the 2D bounding boxes are automated by projecting the 3D coordinates of the objects onto the 2D image plane and capturing the extents in the screen space [1].

Generating one labelled synthetic image using this method takes approximately 1 second, drastically speeding up the process of gathering and annotating diverse datasets for training a robust object detection algorithm.

3.2 Results

Trainings were conducted based on YOLOv8 with 150 epochs, a batch size of 16, and approximately 2500 images without Automatic Mixed Precision (AMP) training. Additional parameters used are the default training settings provided by Ultralytics [3]: an initial and final learning rate of 1×10^{-2}, a momentum of 0.937, and a weight decay of 5×10^{-4}.

Table 1 quantifies the method in match gameplay using conventional metrics, namely, precision, recall, F1-score, and mean average precision (mAP). mAP50 provides a single number metric to summarise the detection performance of the model in multiple classes of objects. Similarly, mAP50-95 is determined across ten intersection-over-union thresholds ranging from 0.5 to 0.95, with increments of 0.05. In robot soccer, precision is more important than recall, as false positives lead to incorrect decisions, such as detecting background objects as the ball.

The mAP50 being around 87% shows that our method provides a balanced and solid approach, providing a good starting point for generating synthetic data for the MSL. The lower recall is due to the limited size of the dataset used, however, in the future, the number of images will be increased significantly which will increase the recall accordingly. Furthermore, using the detection algorithm alongside an estimation and tracking algorithm reduces the system's dependency on a high recall. Especially since this high precision low recall ratio means that we limit the number of undesirable false positives (ghost obstacles), only strengthening the estimation and tracking algorithms.

Figure 3 shows an example of the synthetic data trained YOLO performance on the validation dataset. This checks how well it performs in a real-world match scenario with motion blur. Since the datasets do not inherently model motion

Table 1: Validating our method the object detection for generated datasets on real life match gameplay.

mAP50↑	mAP50-95↑	Precision↑	Recall↑	F1-score↑
0.868	0.7037	0.953	0.759	0.845

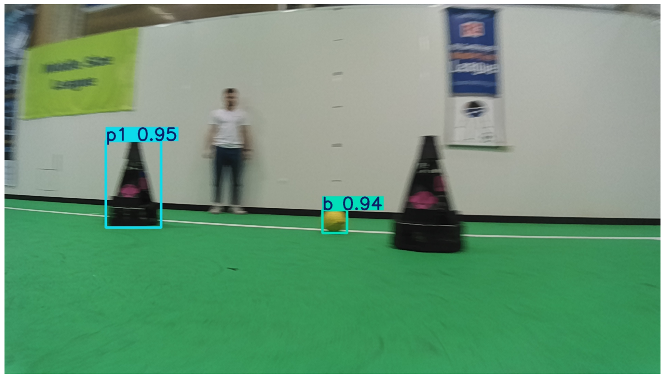

Fig. 3: Example of the validation dataset from real life match gameplay, where 'p1' represent the robot and 'b' represents the ball.

blur, it is possible that this can affect the performance of the network. However, tools such as Roboflow could be used, which use AI techniques to introduce motion blur in static images.

Given these results, our aim is to use this method to detect robots and balls at RoboCup 2025.

4 Automatic Calibration

In RoboCup, the goal is to develop fully autonomous robots capable of playing soccer without human intervention. While robots are able to operate autonomously during matches, a lot of human involvement is required for match preparation. Various aspects of our robots, including ball detection, localization on the field, ball handling, and shooting mechanisms, require periodic calibration. One key issue is the need for constant recalibration, particularly for ball detection, which currently depends on lighting conditions. By implementing AI vision, as discussed in Sec. 3, we can create a more robust ball detection system that adapts to changing lighting without needing recalibration. The other aspects of the robots that require frequent recalibration are more mechanical in nature. For example, to shoot the ball, the robots need to know the physical alignment of the shooting mechanism with respect to the vision system. Collisions with obstacles and other robots, which still frequently occur in a match, lead to slight changes in this alignment. It would therefore be very beneficial if the robots could correct for these changes in the calibration values in real-time during a match. As a first step, we want the robots to perform this real-time calibration outside the match, such that we still have control over the calibration. Then, when this proves robust and reliable, we can incorporate the real-time calibration into our match play.

We propose the robotic equivalent of a human warming-up session. During this warm-up, the robots will perform real-time calibration and test the overall system functionality to ensure everything is working properly. In a simple setup, the robots move in a predefined pattern while passing the ball to each other, allowing them to calibrate both their shooting and receiving mechanisms. More advanced warming-up sessions can include shooting at the goal to fine-tune the height and speed of the robots' shots.

Currently, the real-time calibration of the angle between the vision system and shooting mechanism is implemented. Up until now, calibration has been done by manually performing a shot, taking an image of the ball leaving the shooting mechanism, and adjusting the value of the shooting angle in the software. This procedure is now replaced by an automatic calibration procedure. Whenever the robot passes, it will track the direction in which the ball is moving by analysing the camera data. This is then compared to the location of the intended target. The angle between the two is then used to adjust the calibrated shooting angle. Figure 4 shows a situation in which the intended ball trajectory (red) does not align with the actual shooting angle (yellow).

In a parallel project, we use real-time data analysis to determine whether a recalibration of the robot is necessary. When the robot determines one of its calibration values is no longer correct, for example, after a collision, it will request a substitution to allow it to be fine-tuned in the team technical area. In the future, the combination of real-time calibration possibilities and this validation of the calibration values during play allow for the robot to fine-tune its values when it deems necessary to do so.

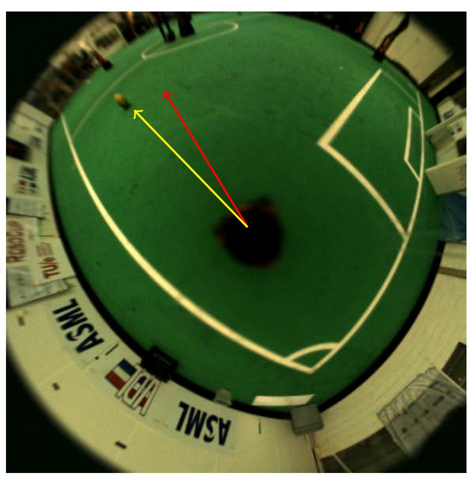

Fig. 4: An example of an image taken by the robot after kicking the ball. The red arrow shows the expected direction of the shot, while the yellow arrow indicates the actual direction of the shot.

5 Mini Cheetah Goalkeeper

As the RoboCup MSL needs to keep evolving to stay relevant within the robotics industry, Tech United is looking into new ways of increasing the dexterity of the robot platform, while maintaining the capacity for fast gameplay, team strategy and high entertainment value. One consideration is to make use of quadruped robots, being able to use its legs for both manoeuvring and handling the ball (so called: loco-manipulation), without the need to move its entire body. Besides, these systems offer increased adaptability to more uneven terrains, such as grass, easing the transition towards playing on an actual outdoor soccer field. Current research focusses on creating quadrupedal goalkeeper that could provide a crucial advantage by enabling jumps and enhanced movement.

The concept of quadrupedal goalkeeping is a complex task that requires the combination of legged locomotion and object manipulation, making it challenging due to the sudden interaction with the environment, particularly a soccer ball. Model-free approaches have proven to be more resilient to modelling errors and better at handling dynamic and unpredictable environments. Therefore, the current method focuses on using hierarchical reinforcement learning (RL), which decomposes the complex locomotion task into smaller low-level skills, enabling efficient learning and reusability. The high-level planning policy coordinates these skill policies by issuing inputs such as the ball's position, and the goalkeeper's position and orientation relative to the goal, for strategic positioning and ball-saving. The architecture shown in Figure 5 provides the overall framework from sensor input to control output for the individual joints.

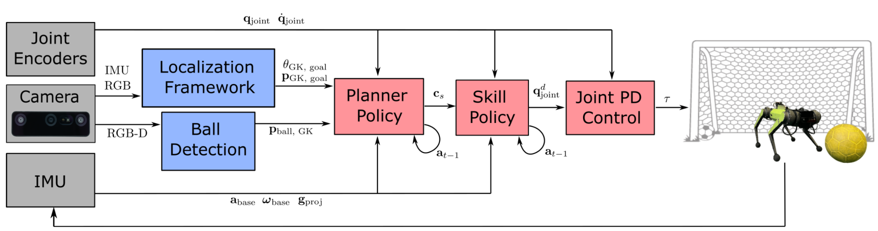

Fig. 5: Data flow schematic of the combined localization and locomotion frameworks. The sensors on the quadruped (gray) provide input to the localization framework (blue) and locomotion framework (red). The localization framework processes raw RGB-D camera data, producing a compact state representation for the high-level planner policy in the locomotion framework. The planner uses this information to command a skill policy, which computes motor torques for the quadruped's movements.

Domain randomization is used to minimize the sim-to-real gap and allow for zero-shot transfer of the trained policy, by randomizing parameters such as initial joint positions, joint friction, body mass, and base-applied force, which helps the policy remain stable under a range of conditions.

Although the trained policies were successfully deployed on the robot, an additional delay in retrieving camera images and the sim-to-real gap still present, resulted in the current method covering 23% of the goal area using a side-step skill. Further optimizations include retraining on the actual hardware and incorporating more skills such as diving and jumping.

6 Conclusion

In summary, this paper outlines the major scientific and technological advancements of the Tech United soccer robots over the past year. The introduction of synthetic datasets for training convolutional neural network-based object detection algorithms has been highlighted, demonstrating their effectiveness in improving performance for robot soccer while significantly accelerating the annotation process. This innovation simplifies the inclusion of additional robots and objects into the system. Furthermore, progress towards more autonomous calibration was presented, including the implementation of automatically calibrating the robot's heading direction for kicking. This approach reduces the reliance on human intervention and takes important steps toward achieving the vision for 2050, where robots will independently warm up by calibrating their sensors and actuators. Lastly, the integration of a quadrupedal goalkeeper marks a significant milestone toward more human-like robots. This development aligns closely with the long-term objective for 2050: the evolution and deployment of legged robots. These advancements contribute to an even higher level of dynamic soccer matches for future RoboCup competitions.

References

1. 2D Bounding Box for Deep Learning Image Detection (2024), https://www.fab.com/listings/46a82ff2-8e9b-4a04-9434-7c7c8bebac63, accessed: 2025-01-22
2. Almeida, L., Santos, F., Facchinetti, T., Pedreiras, P., Silva, V., Lopes, L.S.: Coordinating distributed autonomous agents with a real-time database: The CAMBADA project. In: Computer and Information Sciences - ISCIS 2004. pp. 876–886. Springer Berlin Heidelberg, Berlin, Heidelberg (2004)
3. Jocher, G., Qiu, J., Chaurasia, A.: Ultralytics YOLO (Jan 2023), https://github.com/ultralytics/ultralytics
4. LumaAI: Gaussian Splats for 3D Visualization (2025), https://lumalabs.ai/, accessed: 2025-01-22
5. O'Mahony, N., Campbell, S., Carvalho, A., Harapanahalli, S., Hernandez, G.V., Krpalkova, L., Riordan, D., Walsh, J.: Deep learning vs. traditional computer vision. In: Advances in computer vision: proceedings of the 2019 computer vision conference (CVC), volume 1 1. pp. 128–144. Springer (2020)
6. Ruben M. Beumer et al.: Tech United Eindhoven Middle Size League Winner 2023. In: RoboCup 2023: Robot World Cup XXVI. pp. 428–439. Springer Nature Switzerland (2024)