RedbackBots Team Report RoboCup Soccer SPL 2025

Timothy Wiley, John Thangarajah, Sera Babayigit, Michael Chatham, Kairus Delgado, Tom Ellis, Mark Field, Benjamin George, Samuel Griffiths, Shannon Hanidar, Peter Hong, Yashal Hookmani, Brayden Killeen, Thamadi Kulathunga, Karen Laurentia, Prawal Lohani, Luke Mullan, Murray Owens, Minh Hoang Pham, Clayton Phillips, Arturo Sandoval, Jonathan Theofilas, James Thomson, Rishi Verma, Kurniawan Zhong Zhen

Artificial Intelligence Innovation Lab,
School of Computing Technologies, STEM College,
RMIT University, 124 La Trobe St, Melbourne, Victoria, Australia.
w: http://www.rmit.edu.au/ailab e: stem.redbackbots@rmit.edu.au

1. Introduction

RedbackBots competed in our third on-site competition for the RoboCup Soccer Standard Platform League (SPL) in the 2025 international competition held at Salvador, Brazil. We placed 11th in the combined Champions Cup/Challenge Shield and 3rd among the Challenge Shield teams. Our team is part of the AI Innovation Lab at RMIT University which aims to develop and extend solutions to further human capabilities, improving our quality of life using artificial intelligence. We see RoboCup Soccer SPL as an excellent platform for research and innovation, and as an education platform for students in RMIT University to learn and apply skills in the field of Artificial Intelligence.

In 2025, we improved our codebase by overhauling our vision system, whistle detector and team communication protocols. Additionally, we continued development of our Augmented Reality (AR) tool GameSight to increase its usability.

2. Team Organisation

RedbackBots was established in 2019. Our team is formed of undergraduate and postgraduate students from Computing and Engineering programs at RMIT University. The team is supervised by Dr. Timothy Wiley and Prof. John Thangarajah who have been involved in RoboCup for 15 years. Our team is joined by final year ‘Capstone’ students who lead the development of our GameSight visualization tool as part of their final semester Work Integrated Learning course. Our team holds regular meetings led by the student lead, thus, our students retain a strong ownership over the direction and running of the RedbackBots team.

Our 2025 SPL travelling team consisted of 6 students (Karen Laurentia, Kurniawan Zhong Zhen, Mark Field, Samuel Griffiths, Tom Ellis, and Thamadi Kulathunga), and 1 staff member (Timothy Wiley). The team was supported remotely in Melbourne by 3 students (Murray Owens, James Thomson, Rishi Verma).

Figure 1: 2025 RedbackBots Travelling Team (Salvador, Brazil)

RedbackBot's participation in RoboCup Soccer SPL is summarised below.

Year	Participation	Results
2025	RoboCup 2025 Combined Challenge/Champions	11th Place Overall 3rd Place among Challenge Shield teams
	Technical Challenges	**Winner**, Open Research Challenge **Winner,** Research Innovation Award
	Leaderboard Fastest Walk	3rd Place
	Leaderboard Best Kick	3rd Place
2024	RoboCup 2024 Challenge Shield	3rd Place
2023	RoboCup 2023 Challenge Shield	5th Place
2022	Technical Challenges	8th Place Overall 3rd Place Visual Referee Challenge
2021	Qualified	Withdrew due to COVID-19 travel restrictions
2020	Qualified	Competition cancelled due to COVID-19 pandemic

The RedbackBots team convenes weekly where members collectively engage in codebase development, organisation, and hold regular practice matches. Our team leverages Microsoft Teams and Discord as communication platforms, fostering an environment where members can both seek assistance and collaborate effectively. We maintain our codebase using Bitbucket. We have made our 2025 code release available through the RMIT GitHub Organisation.

3. Awards and Recognition

Highlights of RedbackBots awards and recognition include:

- *Winner*, 2025 Soccer SPL Open Research Challenge
- *Winner*, 2025 Soccer SPL Research Innovation Award
- *Best Paper*, alt.HRI 2025. 20th Annual IEEE/ACM International Conference on Human-Robot Interaction (HRI 2025), CORE A*, for Sanfilippo, F., Wiley, T., & Rousi, R. RoboCup Soccer Autonomy Uprising: How Crowds, Referees, and Humanoid Robots are Redefining the Future of Human-Robot Interaction.

4. Codebase

The codebase for our 2025 code release consists of our own contributions build on top of the previous work of RoboCup SPL teams. We acknowledge the following code-use as part of the development of our RedbackBots software:

- Primarily the rUNSWift 2019/2020 code release is the original basis of our codebase for the software architecture, locomotion, vision, and communication modules (UNSW Computing 2019).
- We have incorporated into our codebase updated Machine Learning models from the rUNSWift 2022 code release (UNSW Computing 2022).
- The B-Human (and Nao Devils) 2022 code release, for the structure and creation of a bootable Nao V6 Ubuntu 20.04 image, along with general build configuration scripts, and general robot configuration management scripts (BHuman 2022).
- The B-Human 2022 code release for the Nao V6 camera driver compatible with the Nao V6 Ubuntu 20.04 image (BHuman 2022).
- The B-Human 2023 and 2024 code releases (BHuman 2023, 2024) for their vision system and neural network weights. The 2023 release is used for the penalty mark and ball detector weights.
- The Nao Devils 2023 code release (NaoDevils 2023) for their whistle detector implementation and the Nao Devils 2024 code release (NaoDevils 2024) for their whistle neural network weights

Our codebase is divided into the following modules:

- Motion, implemented in C++, that includes an integrated walk engine and kick engine.
- Perception, that includes Vision for all object recognition tasks.
- Whistle, that includes a whistle detector.
- Localisation, that provides the state estimation of the robot including its on-field position and a shared-ball estimation.
- Behaviour, that provides the decision-making and action selection of the soccer playing software.

- Infrastructure, that provides data structures and software tools for supporting the other modules, including a C++/Python interface in Boost, and libraries for loading Machine Learnt models, and build-chain tools.

In 2025, we have made contributions to:

- Perception:
  - Vision system overhaul
  - Nao devils whistle detector port
- Team communications:
  - Event-based optimization
- Game Sight

Additionally, in 2024 we have developed a novel Augmented Reality (AR) visualisation system, GameSight, for the MetaQuest platform. GameSight enables visualising GameController packets and the information available to the Team Communication Monitor in real-time projected in AR onto the field with full colour feed-through rendering.

We describe our contributions in this team report, and refer to the published work of rUNSWift, B-Human and Nao Devils for the work on which we depend.

4.1. RedbackBots 2025 Code Release

Our code release for 2025 is available on the RMIT University GitHub Organisation at:

https://github.com/rmit-computing-technologies/redbackbots-coderelease

The 2025 code release is located under the tag "coderelease2025".

4.2. GameSight 2025 Code Release

We have released our GameSight AR visualisation tool for 2025 on the RMIT University GitHub Organisation at:

https://github.com/rmit-computing-technologies/redbackbots-gamesight-coderelease

The 2025 version of the code release is located under the tag "coderelease2025".

5. Perception

Our contributions within Perception have been focused on porting B-Humans vision system and Nao-Devils Whistle Detection.

5.1. B-Human Vision Port

For Robocup 2025, we have ported B-Humans vision systems into our C++ codebase, adapting their streamable architecture to our blackboard architecture. We used the 2024 release for all detectors excluding the penalty mark and ball detector, for which we used the individual detectors from the 2023 code release. We also ported and used B-Humans referee detection.

5.2. Whistle Detection

At RoboCup 2024, the constantly changing background noise caused significant issues with our former whistle detector. Calibration took too long to complete before each game, forcing us to rely on multiple calibration configs that were often in limbo between either being too sensitive or not sensitive enough.

For RoboCup 2025, we ported the Nao-Devils whistle detector (presented in the 2024 symposium) into our C++ codebase and scrapped our previous python-based whistle detector. During the competition, we used only one calibration config, and changes were made only to accommodate different whistle types (hand and mouth whistles)

This also resulted in a significant CPU usage reduction as our previous detector was a standalone python script that polls with IO-heavy operations to communicate with our C++ codebase. The port was minimally modified to use BHuman's CompiledNN as the neural network inference backend.

6. Behaviours

We have separated our field positions from our roles. Any robot can go into any role (e.g. striker, defence) from any assigned position.

6.1. Position Selector

We dynamically assign each robot's position depending on the active status of other robots. The field is divided into four quadrants: two attacking and two defending. If all robots are active, they are each assigned one quadrant. If a robot becomes inactive, the neighbouring robot will expand its zone to cover the inactive robot's quadrant. When only one robot is active, it is assigned the special position of 'Superstar', whose quadrant covers the entire field.

6.2. Role Selector

We also dynamically assign roles to each robot. Robots will enter the striker role if the ball is inside its quadrant, or it is a half centre circle distance away from the ball. We use a scoring system (described in 5.3) to ensure that only one robot goes into striker at a given time. Non-striker robots are either assigned to defence or standby, depending on whether the ball is in the attacking or defending half. Defending robots will move between the ball and the goal to intercept it. When the team has lost sight of the ball for a certain amount of time, each robot will return to its default position and turn to search for the ball.

7. Camera Calibration

Camera calibration proved essential for our robots' vision systems in the inconsistent lighting conditions at RoboCup 2025. Before every game and on each field, we performed calibrations to ensure the robots could localise on the field while identifying the ball. We sometimes had to compromise line perception to prioritise ball detection when shadows caused blending issues.

8. Team Communication

8.1. "Mine" Calls

We implemented a coordinated strategy where each robot calculates its relative "ability to kick the ball," known as the "ball score." This score reflects how well-aligned, close, and visible the ball is to the robot.

The robot with the lowest ball score sends a packet to confirm to the team that it is the best candidate to kick the ball. This approach minimises packet use, as robots do not need to send "heartbeat" packets at regular intervals. Instead, packets are sent only when a robot believes it is the new best kicker or when other robots overestimate its ability (e.g., after the ball is kicked away).

When comparing ball scores, robots' factor in the time since the last packet was sent, adding a set amount for each second elapsed. For example, if a packet was sent four seconds ago, the score is adjusted accordingly. If a robot decides it is superior and sends a packet, but another robot remains the best by a margin, it quickly sends another packet to reaffirm its status. This method reduces overall packet usage, especially when a single robot is alone with the ball, mimicking the "mine" call in a real soccer game with real-time play negotiation.

8.2. Event-Based Optimisation

We have completely rewritten our team communication protocols to operate more efficiently on dynamic events. Previously, our communication system relied on sending fixed-size (and fixed content allocations) packets at regular intervals, regardless of whether meaningful information needed to be transmitted. This approach limited the extensibility of the information we could send to team-mates, as if we wanted to send an additional piece of information, it would need to replace information already allocated in the packet.

The new communication system contains only a small static packet header, containing only the Player Number and an Event Hash, followed by a dynamically allocated set of data transmitted by events. With a finite set of events that can be sent, the Event Hash describes which of the available event types have been triggered and transmitted within the current packet. Each triggered event with associated data is then serialised in its most compact form (e.g., boolean values are compressed to a single bit), with no memory alignment, to best utilise the available packet space for data only when it is needed.

Our event-driven architecture centres around two blackboards: the transmitter and receiver. The transmitter manages event scheduling, criticality levels, and determines when packets should be sent based on time-to-send thresholds and total payload size. Rather than sending at fixed intervals, packets are dispatched only when events expire or when accumulated event data would fill a packet. Higher-priority events are packed first when overflow occurs, ensuring critical information always gets through.

9. Software Infrastructure

Our infrastructure is originally derived from rUNSWift and we have incorporated B-Human's build system. The root level structure of our infrastructure is described below.

Folder	**Purpose**
Build	The location where C++ build files, and the bootable Image is placed.
Config	Software and robot configuration files for the RedbackBots team. In our code release, configuration files have been replaced with default values.
Docs	Contains some pages from our internal team wiki. It is stripped down to include only steps on getting our code running on a robot.
Install	Software and scripts for installing the RedbackBots software on a Nao robot. Static files that are deployed to the Nao. These include operating system configurations, static robot configuration files for use in the RedbackBots software, audio files, and machine learnt models.
Make	Software and scripts for building the RedbackBots software, primarily consisting of CMake configuration files. Also includes scripts for generating the bootable USB image.
Src	C++ and Python source files of the RedbackBots codebase.
Util	Utility files, at present only containing pre-compiled libraries for various operating systems architectures, including x86_64 and arm64.

Contained within the SRC folder, noted above, is the bulk implementation of our soccer capable code, divided into subdirectories, summarises in the table.

SRC Folder	Purpose
behaviour	All python files/modules that implement our robot behaviours implemented in a python sub-system called from within the C++ code. The entry point into the Python sub-system is the behaviour.py file.
boost_test	Standalone C++ program for testing Boost functionality on the Nao robot and linked against the provided Boost Buildchain. By default, this program is not compiled, but can be enabled within the CMake configuration files.
lola_test	Standalone C++ program for testing LOLA connectivity on the Nao robot. By default, this program is not compiled, but can be enabled within the CMake configuration files.
offnao	Standalone local program, written in C++, for connecting to the RedbackBots software running on a Nao robot and observing the robot state. Currently offnao supports viewing of processed camera images, vision features, localisation, shared ball information, and the Blackboard state.
robot	C++ source files for the robot soccer code, that is cross-compiled and deployed onto the Nao robot.

Further information about the software infrastructure is documented within our code release.

10. League Contributions

10.1. League Tools (GameSight)

RedbackBots developed an Augmented Reality (AR) visualiser, GameSight, compatible with TCM and GC. This enables visualisation of localised game information through AR platforms including the Meta Quest. This was demonstrated at the 2024 & 2025 competition, is available for teams to download and use themself and will form part of our code release. We plan to continue developing this platform to enable a more diverse range of toolsets that can be used league wide and to explore unique human robot interaction research.

GameSight is an augmented reality (AR) visualisation system developed by RedBackBots that integrates directly with the RoboCup Game Controller and NAO robot networks to provide real-time match awareness through Meta Quest devices. Built in Unity using a component-based architecture, GameSight fuses multiple data streams—including robot telemetry, camera feeds, and control messages—into a single immersive AR interface. Its five-layer system design manages AR tracking, input detection, field calibration, network communication, and data visualisation, enabling users to view robot states, localisation data, and game metrics directly over the live field. A dual-interaction system supports both hand-tracking and controller inputs, while the offline-ready design allows full functionality without internet access, essential for restricted competition environments. GameSight also supports live mirroring to external displays for collaborative analysis and demonstrations. A user study is underway to evaluate how AR overlays improve human understanding of robot behaviour, localisation, and match dynamics within RoboCup's Standard Platform League.

10.2. Committee Roles

The following RedbackBots team member served as a committee role in the below seasons:

RoboCup Season	Executive Committee	Technical Committee	Organizing Committee
2025	Timothy Wiley	Tom Ellis	Samuel Griffiths
2024	*N/A*	*N/A*	Tom Ellis
2019–2022	*N/A*	Timothy Wiley	*N/A*

11. Outreach and Community Events

A core mission of RedbackBots is outreach in STEM (Science, Technology, Engineering and Mathematics), promotion of the benefits of Autonomous Robotics within the public view, and encouraging primary and secondary school students to pursue studies in STEM fields. RedbackBots engages in various private and public outreach events, with highlights including:

- Formula 1 Grand Prix, RMIT STEM Technology Hub, 2025.
- Melbourne Royal Children's Hospital Visits, 2023, 2024.
- Australian National Science Quiz, 2023.

12. Research Publications and Articles

RedbackBots supports the research initiatives of the SPL and views our league as an important avenue of research in autonomous robotics. We also support the active public promotion of RoboCup and the SPL. Our team publications and articles are below.

12.1. Awards

Best Paper, alt.HRI 2025. 20th Annual IEEE/ACM International Conference on Human-Robot Interaction (HRI 2025) for Sanfilippo, F., Wiley, T., & Rousi, R. *RoboCup Soccer Autonomy Uprising: How Crowds, Referees, and Humanoid Robots are Redefining the Future of Human-Robot Interaction.*

12.2. Thesis Publications

Owens, M. (2023). *Visual Referee Signals for RoboCup Standard Platform League*. Master's Minor Thesis.

Lohani, P. (2022). *Visual Referee Challenge for RoboCup Soccer*. Master's Minor Thesis

12.3. Research Papers

Rodriguez, A. I .S, Ellis, T., Killeen, B., Owens, M., Griffiths, S., & Wiley, T. (2025) *GameSight: Augmented Reality Visualisation for the RoboCup Soccer SPL*. Proceedings of the RoboCup Symposium (to appear)

Abstract: We introduce GameSight, an open-source Augmented Reality (AR) system for the Meta Quest platform providing real-time visualisation of internal robot states and behaviours for the RoboCup Soccer Standard Platform League (SPL). Current methods for monitoring SPL games depend on tools displayed on computer screens, or extensive post-game log analysis. This can limit interpreting robot dynamics as they occur, impeding gameplay improvements, debugging, and adjustments between competition games. Pass-through AR systems that project 3D renderings onto a live view of a scene can provide spatially aligned information that is easier for humans to interpret compared to alternatives. AR systems mirror visual and spatial forms of human-to human communication, and can create a "more natural" feeling in robotto-human communication, that is, communication within the context of Human-Robot

Interaction (HRI). GameSight uses pass-through AR technology to overlay 3D spatially aligned renderings of robot localisation, behavioural decision-making, and sensor information onto a live view of the soccer field. GameSight was trialled during the 2024 RoboCup World Cup, showing a benefit for technical experts in testing gameplay, and for non-technical audiences to better understand Soccer SPL games. This highlights GameSight's potential, not just for teams, but a wider use with HRI for improving robot-to-human communication.

Sanfilippo, F., Wiley, T., & Rousi, R. (2025) RoboCup Soccer Autonomy Uprising: How Crowds, Referees, and Humanoid Robots are Redefining the Future of Human-Robot Interaction. In: 2025 20th ACM/IEEE International Conference on Human-Robot Interaction (HRI). pp. 1131-1139. DOI:10.1109/hri61500.2025.10974154

Abstract: This paper explores the dynamics of Human-Robot Interaction (HRI) in public spaces, focusing on how humanoid robots engage with human crowds in the competitive RoboCup Soccer environment. We examine the role of spectatorship, where emotional engagement arises through indirect observation of engineering-driven competition, drawing parallels between human soccer and robot sports. The potential for autonomous systems to elicit collective emotions and systematically study such experiences is investigated. Using the Autonomy Levels for Unmanned Systems (ALFUS) framework, we assess RoboCup soccer robots' autonomy in terms of mission complexity (MC), environmental complexity (EC), and external system independence (ESI). Additionally, the Autonomy and Technology Readiness Assessment (ATRA) method supports gradual capability enhancement, providing a roadmap to higher autonomy. Based on this established methodology, we introduce the Robot-Crowd Interaction Framework (R-CIF), a novel conceptual framework defining the roles of actors involved, to connect theoretical insights with real-world applications. This work highlights the significance of crowd affectivity in robotic sports to boost public engagement and proposes directions for future research on collective emotional dynamics in HRI.

Owens, M. & Wiley, T. (2024). *Lightweight Real-Time Gesture Recognition for Dynamic Soccer Referee Signals*. Proceedings of the RoboCup Symposium

Abstract: We present a lightweight keypoint-based Recurrent Neural Network (RNN) approach to the 2023 RoboCup Standard Platform League Visual Referee Challenge. The goal of the challenge is to classify 13 different static and dynamic referee gestures during a robotic soccer match. The developed solution must be lightweight and perform well from anywhere on the field. Multiple referees may be present on the field, with the distinguishing trait of the visual challenge referee being their red gloves. We use the lightweight Convolutional Neural Network pose detector BlazePose to extract the pose key-points. To isolate the referee in the frame, HSV colour segmentation is used to find red glove regions and mask unneeded sections of the image. A lightweight RNN is used to classify the sequence of keypoints. We compare the results of our work to previous solutions developed for the 2022 Visual Referee Challenge. Analysis on real-time testing shows our method achieves strong performance for all gestures on many different locations on the soccer field.

Lohani, P. & Wiley, T (2023). *Hybrid Methods for Real-time Video Sequence Identification of Human Soccer Referee Signals*. Proceedings of the RoboCup Symposium

Abstract: The work presents a faster hybrid approach in order to identify different signals from human soccer referee in real-time for the “Visual Referee Challenge” which is introduced in 2022 RoboCup Standard Platform League (SPL). The main objective of the challenge is to develop a solution that could be applied to the resource constrained humanoid Nao V6 robots for better in-game human-robot interaction. Our approach consists of a lightweight machine learning model combined with hand-crafted heuristic method. Further, the performance of the devised approach is contrasted against the 2 sophisticated Convolutional Neural Networks (CNNs) models that are specifically designed for identifying different key-points in human body. The result in single image shows that there is significant trade-off between the high accuracy of the CNN models and faster processing time of our approach. However, the analysis on real-time video sequence demonstrates that the proposed hybrid model produce strong performance in much shorter time and is preferred over more powerful and time-consuming CNNs, such that, the devised algorithm achieved 3rd position in the official competition.

12.4. Media Articles

AIHub. *Feature in: RoboCup2024 – daily digest: 21 July*. (2024) RedbackBots, were featured on the AIHub's final day digest for RoboCup 2024 for our innovative AR live visualisation system of RoboCup games - GameSight.

Acknowledgements

RedbackBots acknowledges the support of RMIT University, along with the School of Computing Technologies and the RMIT Artificial Intelligence Advanced Innovation Experience Hub (RAIsE Hub) both within the STEM College for their significant investment in Artificial Intelligence and Robotics. The Artificial Innovation Intelligence Lab, under which our team resides, was launched in late 2019. Our RedbackBots RoboCup team is a key part of the lab's strategic direction. RMIT University the RAIsE Hub in late 2024 with a focus on AI applications for research, education and industry which has provided the support for our RedbackBots, including our 2025 participation.

We also acknowledge support from the former Centre for Industrial Artificial Intelligence and Research Innovation (CIAIRI) across 2021 – 2024. CIAIRI funded our initial launch and participation in RoboCup from 2019 to 2024.

RedbackBots also acknowledges the large number of teams who have contributed to the various codebases of which our team has developed our code, and the resources provided by the RoboCup Soccer SPL.

References

Allen, J. (1977), *Short term spectral analysis, synthesis, and modification by discrete Fourier transform*, IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 25, no. 3, pp. 235-238.

Bazarevsky, V., Grishchenko, I., Raveendran, K., Zhu, T., Zhang F. and Grundmann, M. (2020). *BlazePose: On-device Real-time Body Pose tracking*. https://arxiv.org/abs/2006.10204.

BHuman (2022). *BHumanCodeRelease, Tag: coderelease2022*. GitHub repository. https://github.com/bhuman/BHumanCodeRelease/releases/tag/coderelease2022. Last accessed October 2023.

BHuman (2023). *BHumanCodeRelease, Tag: coderelease2023*. GitHub repository. https://github.com/bhuman/BHumanCodeRelease/releases/tag/coderelease2023. Last accessed July 2025.

BHuman (2024). *BHumanCodeRelease, Tag: coderelease2024*. GitHub repository. https://github.com/bhuman/BHumanCodeRelease/releases/tag/coderelease2024. Last accessed July 2025.

Cao, Z., Hidalgo Martinez, G., Simon, T., Wei, S., Sheikh, Y.A. (2019). *Openpose: Real-time multi-person 2d pose estimation using part affinity fields*. IEEE Transactions on Pattern Analysis and Machine Intelligence. pp. 72-91.

Hochreiter, S. & Schmidhuber, J. (1997). Long Short-Term Memory. Neural Computation. 9(8) pp. 1735-1780

NaoDevils (2023). *CodeRelease, Tag: CodeRelease2023*. GitHub repository. https://github.com/NaoDevils/CodeRelease/tree/CodeRelease2023. Last accessed October 2025.

NaoDevils (2024). *CodeRelease, Tag: CodeRelease2024*. GitHub repository. https://github.com/NaoDevils/CodeRelease/tree/CodeRelease2024. Last accessed October 2025.

Newell, A., Yang, K., Deng, J. (2016). *Stacked hourglass networks for human pose estimation*. European conference on computer vision. pp. 483-499.

UNSW Computing (2019). *rUNSWift-2019-release*. GitHub repository. https://github.com/UNSWComputing/rUNSWift-2019-release. Last accessed October 2023.

UNSW Computing (2022). *rUNSWift-2022-release*. GitHub repository. https://github.com/UNSWComputing/rUNSWift-2022-release. Last accessed October 2023.