Imperial College Bioengineering Department Challenge

Topic – PixyBot Car Race

Watch the race here: https://www.youtube.com/watch?v=mSpLFaTugbU
Final code in my GitHub repository: https://github.com/spaceroborg/PixyBot-Car-Race.git

Keywords – Lane Racing; Visual Heuristics; Bio-inspired Control Strategy

Summary –

For a competitive robotics challenge, I developed and implemented a bio-inspired visual navigation system for a differential-drive robot using a PixyCam. The task was to enable the robot to autonomously follow a lane based on color discrimination, avoid obstacles, and dynamically track visual targets in real time. The system leveraged visual inputs to guide behavior, mimicking perception-driven navigation observed in insects and other animals. The robot was built from scratch using a Raspberry Pi Zero, continuous servos, and PixyCam2.

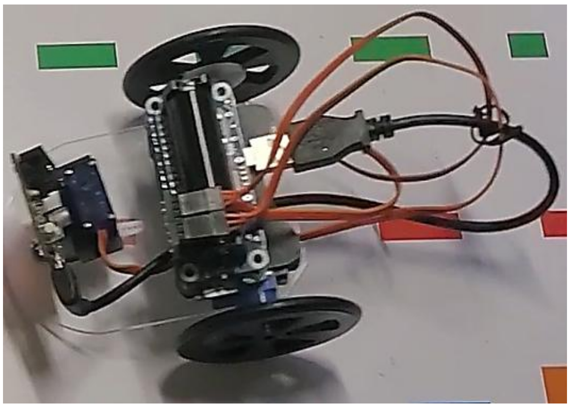

Figure 13: My PixyBot robot car with PixyCam2 and Raspberry Pi Zero.

The Race –

The robot was tasked with completing a course marked by colored lanes while avoiding static and dynamic obstacles. The constraints included limited onboard computation, no access to global positioning or external sensing, and the requirement for closed-loop control driven entirely by real-time camera data.

The challenge was to design a system capable of distinguishing target lane colors from distractors, maintaining lane alignment, and reacting to obstacles—all in a resource-efficient, reactive control framework.

Figure 14: The race track with coloured lanes.

Approach –

I implemented a Python-based architecture that leveraged the PixyCam's color-based block detection. The system used methods such as *getLatestBlocks()*, *isBiggestSig()*, and *isInView()* to extract salient visual features from each frame. Lane following was achieved by identifying the largest colored block of the target signature and calculating its angular offset from the camera's centerline. This angular error was processed through a PID controller to steer the robot via differential drive. The obstacle avoidance module used the size and position of non-lane blocks to adjust motor bias and speed, enabling path correction. Visual tracking of targets was implemented via servo control to maintain object alignment within the PixyCam's field of view.

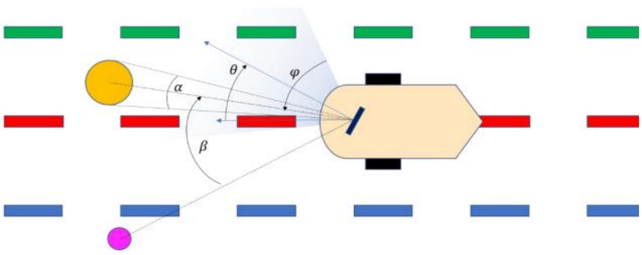

Figure 15: This schematic shows the general conditions of the robot driving environment during the race. The lane markers are labelled by three different colours, the obstacle in yellow, and pylons in magenta by the side of the road.

Bio-Inspired Concepts –

The navigation strategy draws on core principles of bio-inspired robotics, particularly insect-level visual guidance systems. The use of color-based target recognition mimics **selective attention** mechanisms observed in bees and flies, where visual salience informs behavior. The servo-mounted camera tracking reflects **gaze stabilization and target fixation**, akin to vertebrate ocular tracking. The system's reliance on real-time visual input and reactive control embodies the concept of **perception-action coupling**, a central theme in embodied intelligence. Overall, the implementation aligns with biological strategies that emphasize fast, lightweight, and sensor-driven navigation in unstructured environments.

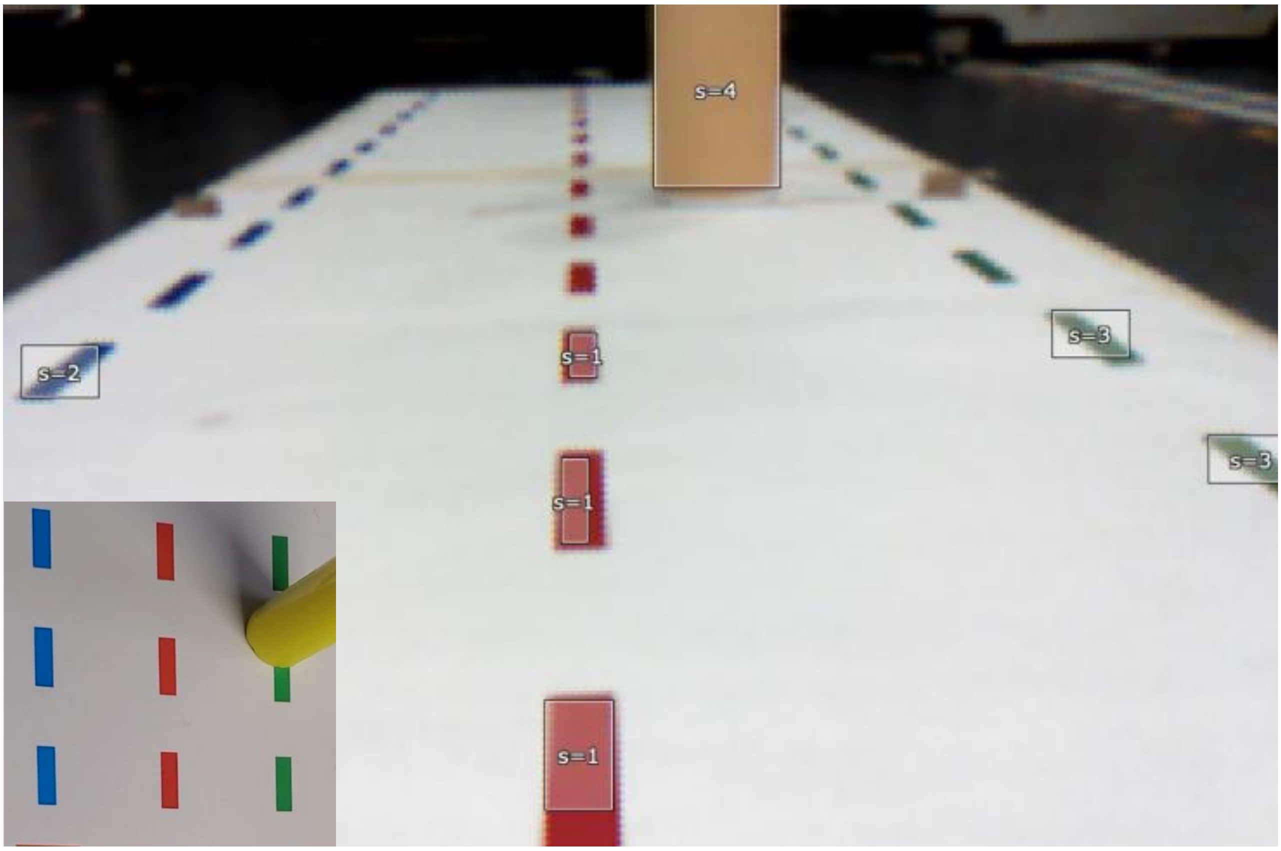

Figure 16: The simplest lane following implementation was to steer the robot toward the largest (nearest) centre land marker. As the nearest marker went under the robot, the next nearest marker became the guide. Visual signals (from target pixels) used directly for motor control were very noisy so a PID controller was used to correct the angular error which led to much more stable and responsive steering.

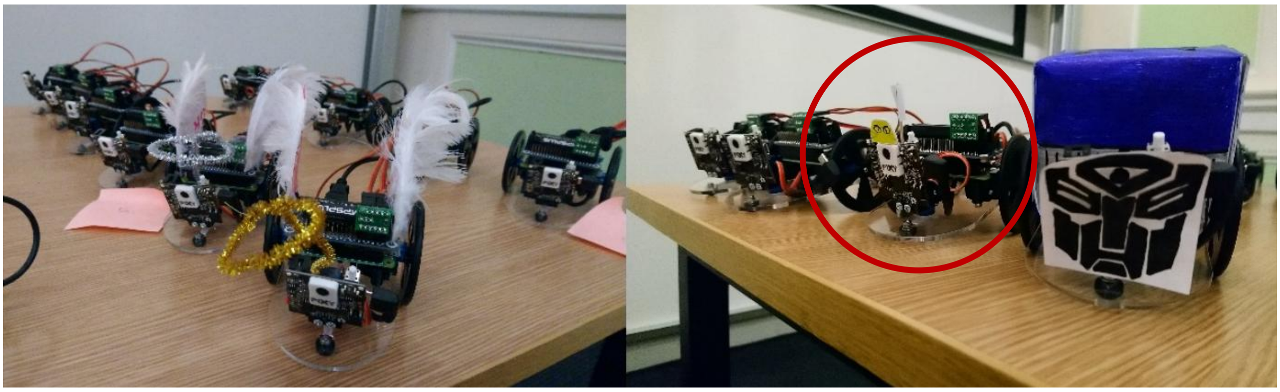

Figure 17: All the robots that participated in the competition. My robot is circled in red.