ECE 445

SENIOR DESIGN LABORATORY

FINAL REPORT

Automatic Sorting Robotic Arm for Table Tennis Balls

Team #52

SIQI PAN (3220111242)
ZHONGHAO WANG (3220111411)
XUCHENG WU (3220111011)

TA: Jianzu Zhang

May 15, 2026

Abstract

This report presents the final prototype of an automatic robotic sorter for table tennis balls. The prototype focuses on color-based sorting of white and yellow/orange table tennis balls using a TCS34725 color sensor, an Arduino Uno R3 controller, and an MG90S-servo robotic arm.

The final system reads red, green, blue, and clear-channel sensor values, classifies the ball as white, yellow/orange, or unknown, and then places the ball into the left bin, right bin, or reject path. Testing with five white and five yellow/orange balls over three rounds produced 15/15 correct yellow/orange classifications and 12/15 correct white classifications, for an overall color-classification accuracy of 90%. A standard sorting cycle after initialization took approximately 4.3 s. The servo supply rail was measured under idle, holding, and moving conditions; during active motion it ranged from 3.812 V to 4.125 V with an average of 3.989 V, and no servo stall, abnormal jitter, or reset was observed. The prototype demonstrates a functional low-cost color-sorting mechanism, while also documenting the limitations caused by uncontrolled ball orientation and the unintegrated 5 V regulator PCB.

Contents

1 Introduction

1.1 Background and Motivation

Table tennis is a visually demanding sport because the ball is small, light, and fast. Prior work on elite youth table tennis players notes that players must perceive the ball and its trajectory within milliseconds to initiate a targeted motor response, and that the short distance between players leaves less than 500 ms for movement in some rallies [1]. This means that ball visibility is not a cosmetic issue: inconsistent contrast can affect how early a player detects the ball and prepares a response.

Ball color is one of the controllable factors that affects this visibility. The International Table Tennis Federation (ITTF) specifies that official balls are white or orange and matt [2]. ITTF competition regulations also require the main color of playing clothing to be clearly different from the ball in use, and floor markings must use colors different from the ball unless otherwise approved [2]. These rules show that the sport treats color contrast as part of the playing environment rather than as a decorative preference.

The table-tennis-specific literature supports the same point. Rogers, McMorris, and Morris investigated the effects of yellow and white ball colors on elite players' speed of response under different background colors [3]. Their study was motivated by the idea that the earlier the ball can be perceived, the more time a player has to move into position. They also emphasized that the contrast between ball and background is an important factor in visual perception. Therefore, a training environment that accidentally mixes white and yellow/orange balls can create inconsistent visibility from one drill to the next, especially when wall, floor, table, and lighting conditions favor one ball color over another.

In training, storage, and classroom laboratory settings, white and yellow/orange balls may be mixed together after repeated use. This creates two practical problems. First, players and coaches may want a consistent ball color for a particular drill so the visual background remains controlled. Second, manually separating balls is repetitive and does not scale well when many balls are used. A compact color-sorting device can reduce this manual work while also providing a practical demonstration of sensing, embedded control, and robotic actuation. The final prototype is therefore framed as a two-color sorting system for common white and yellow/orange table tennis balls, rather than as an earlier broader sorting concept.

1.2 Final Implementation Overview

The final implementation uses a TCS34725 color sensor and classifies balls by color. Earlier project planning used a broader sensing objective, but the verified prototype was intentionally narrowed to the more reliable and demonstrable color-sorting function. The TCS34725 was selected because the device provides digital red, green, blue, and clear light-sensing values through an I2C host interface, and the clear channel provides a useful brightness reference for color-ratio classification [4]. This is the only major scope change from the design-document stage and is reflected in the final requirements and verification results below.

The power subsystem was planned around a custom 5 V regulator PCB whose purpose was to step the battery voltage down to a stable logic-and-servo supply and reduce wiring complexity. In the verified build, however, the servo subsystem was operated from the measured battery rail. Therefore, the power-related verification in this report focuses on the actual servo rail voltage during normal operation and on whether the servos completed their commanded motions without stall, abnormal jitter, or reset.

1.3 Final High-Level Requirements

Table 1 lists the final high-level requirements used for verification.

Table 1: Final high-level requirements used for verification.

Requirement	Success criterion
Color classification	The system shall classify table tennis balls as white, yellow/orange, or unknown. Overall white/yellow classification accuracy shall be at least 90% over the tested mixed-ball set.
Sorting action	The arm shall place white balls in the left output region and yellow/orange balls in the right output region; unknown readings shall be rejected by dropping the ball after the same standard action period.
Timing	The system shall complete one standard color-detection and sorting cycle in approximately 4–5 s after initialization.
Power and motion reliability	During normal operation from the tested battery rail, the servo subsystem shall complete commanded motions without visible stall, abnormal jitter, or controller reset.

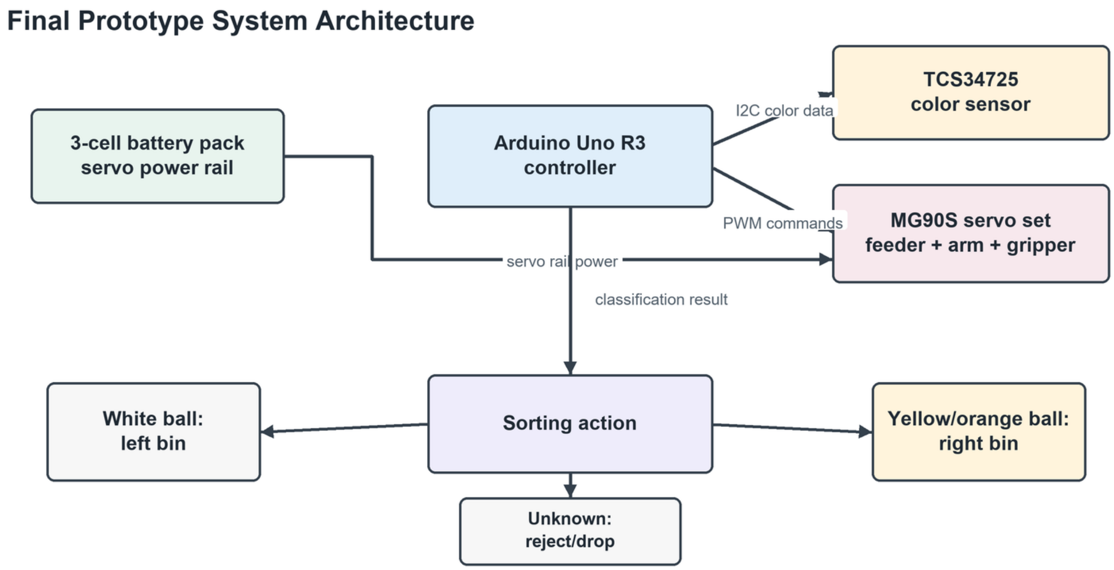


Figure 1: Updated final system architecture for the TCS34725-based color sorting prototype.

2 Design

2.1 System Architecture

Figure 1 shows the final system architecture. The Arduino Uno R3 serves as the main controller. It communicates with the TCS34725 color sensor over I2C and outputs pulse-width modulation (PWM) signals to the MG90S servos. The mechanical system consists of a feeder and a four-servo robotic arm. The arm performs the same general motion sequence for each ball, but the base-servo direction changes according to the detected color.

The final architecture uses one embedded controller and one color sensor, with two color bins plus a reject/drop behavior for uncertain readings. This structure allowed the prototype to be verified reliably with the available hardware while preserving the main embedded-sensing and robotic-sorting objective. Figure 2 shows the Arduino Uno R3 controller used for the final build.

Figure 2: Arduino Uno R3 controller board used as the central embedded controller in the final prototype [5].

2.2 Hardware Specification Basis

The final implementation choices were checked against the relevant manufacturer specifications summarized in Table 2. The Arduino Uno R3 was suitable for the prototype because it provides 14 digital I/O pins, including 6 PWM-capable outputs, which is sufficient for the servo outputs used in the final software [5]. The TCS34725 was suitable for color sensing because it reports separate red, green, blue, and clear-channel values and communicates over I2C [4]. The MG90S servos were selected for compact actuation; TowerPro specifies stall torque of 1.8 kg-cm at 4.8 V and 2.2 kg-cm at 6.6 V, with operating speed of 0.10 s/60 deg at 4.8 V and 0.08 s/60 deg at 6.0 V [6].

Table 2: Manufacturer specifications used to justify final hardware selection.

Part	**Role**	**Relevant specification**
Arduino Uno R3	Controller	ATmega328P board; 5 V operating voltage; 14 digital I/O pins, 6 PWM-capable outputs; 6 analog inputs; 16 MHz clock [5].
TCS34725	Color sensor	Digital red, green, blue, and clear light-sensing values; I2C host interface; IR-blocking filter and clear-channel reference [4].
MG90S	Servo actuator	TowerPro specifies operating voltage around 4.8 V; stall torque 1.8 kg-cm at 4.8 V and 2.2 kg-cm at 6.6 V; speed 0.10 s/60 deg at 4.8 V [6].

2.3 Color Sensing Subsystem

The TCS34725 color sensor is mounted at the ball inspection location. For each cycle, the Arduino reads raw red, green, blue, and clear values. The software normalizes the color values by the clear channel to reduce sensitivity to absolute brightness. A reading is treated as unknown if the clear channel is too small. Otherwise, the software evaluates the relative red, green, and blue ratios and the spread between the maximum and minimum RGB values.

Yellow/orange balls are identified when the red ratio is sufficiently above the green and blue ratios and when the blue ratio remains below the selected threshold. White balls are identified when the RGB channels are close to each other and the spread ratio is small. This approach was chosen because it is simple enough to run on the Arduino Uno while still using all three color channels and the clear channel.

2.4 Control Software and Sorting Logic

At power-on, the program attaches the servos, initializes the TCS34725 sensor, sets the arm to the initial pose, and waits 5 s before entering the main loop. Each loop performs one color reading, classifies the ball, and executes the corresponding servo action. White balls are moved to the left side, yellow/orange balls are moved to the right side, and unknown readings are rejected by opening the gripper after the same standard action interval. Figure 3 summarizes the final sorting logic.

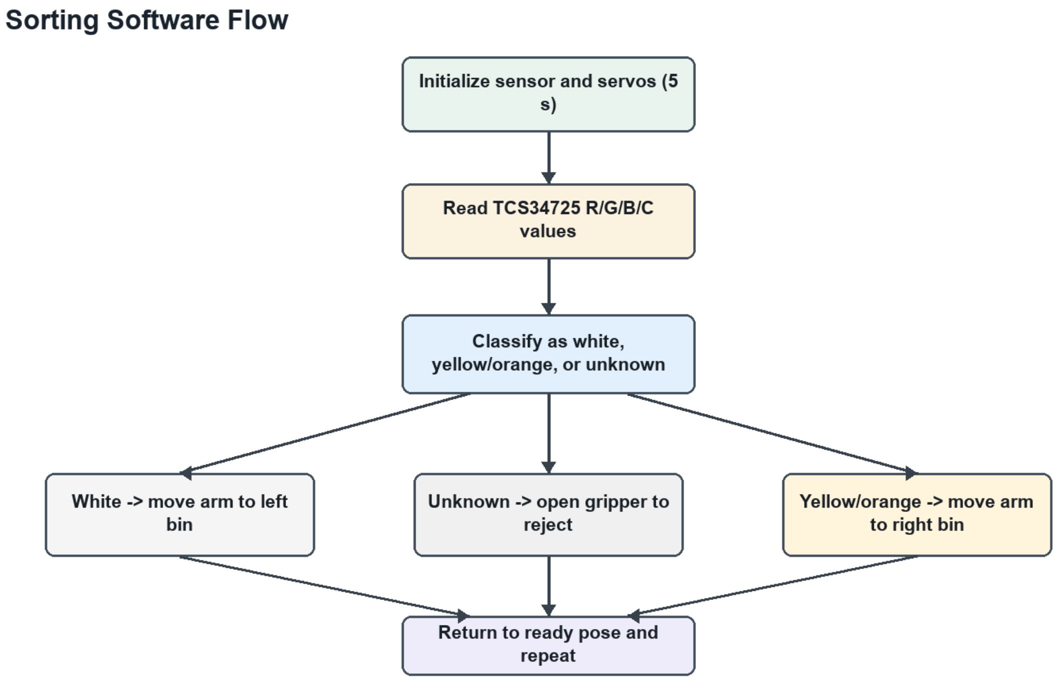


Figure 3: Final sorting software flow for white, yellow/orange, and unknown color readings.

2.5 Robotic Arm and Mechanical Operation

The robotic arm uses four MG90S servos. MG90S servos were selected because they are low-cost, compact, and provide enough torque for moving lightweight table tennis balls according to the manufacturer specifications in Table 2. The arm moves gradually in one-degree software increments to reduce sudden mechanical shocks and avoid forcing the joints into mechanical stops. Table 3 lists the calibrated servo roles.

The mechanical frame was built from acrylic sheet material with custom 3D-printed fixtures for mounting and positioning. This combination kept the structure lightweight and inexpensive while allowing the sensor position, gripper geometry, and servo mounting points to be adjusted during prototype testing.

Table 3: Final servo mapping and calibrated motion roles.

Servo pin	Joint/function	Nominal position	Action range
D3	Base rotation	90 deg center	45 deg right, 135 deg left
D4	Upper arm	90 deg ready	110 deg forward during pickup/drop
D5	Forearm	135 deg raised	100 deg lowered during pickup/drop
D6	Gripper	90 deg closed	45 deg open

2.6 Power Subsystem

The power subsystem has two practical roles: supplying the controller and supplying the servos. The Arduino Uno R3 board itself is a 5 V microcontroller platform [5], while the servos draw larger transient current when starting, stopping, or holding position. A dedicated regulator PCB is useful because it can convert the battery input to a lower regulated voltage, distribute power through clearer wiring, and reduce voltage sag caused by long jumper paths or weak breadboard connections.

The final verification build used a separate battery rail for the servo subsystem. Because the prototype was verified in this battery-powered configuration, the power test focuses on rail voltage during idle, holding, and active servo motion rather than on the regulator PCB output.

2.7 Design Tolerance Considerations

The critical tolerance in the final prototype is color sensing. The most important variable is the surface region facing the TCS34725 sensor. A white ball with its black logo facing the sensor can produce a lower-brightness, lower-saturation reading than a plain white region. This effect was observed during verification and is the most likely cause of the white-ball classification errors.

The practical tolerance strategy is therefore to normalize RGB values by the clear channel and classify white balls using channel similarity rather than absolute brightness alone. The remaining tolerance limitation is mechanical: the prototype does not control the rotational orientation of the ball when it arrives at the sensor. Future revisions should either rotate the ball before measurement or sample multiple surface regions before making a final classification decision.

3 Verification

Verification focused on the completed prototype and the revised requirements in Table 1. The primary tests measured color classification accuracy, sorting-cycle timing, and power-rail behavior under normal servo operation. Testing was performed under indoor ambient lighting without additional controlled illumination.

3.1 Color Classification Accuracy

Ten balls were used for the preliminary classification test: five yellow/orange balls and five white balls. The set was tested over three rounds, giving 30 classification trials. The results are shown in Table 4 and Figure 4.

Table 4: Color classification results over three test rounds.

Round	Yellow/orange balls	White balls	Overall
1	5/5 correct (100%)	4/5 correct (80%)	9/10 correct (90%)
2	5/5 correct (100%)	5/5 correct (100%)	10/10 correct (100%)
3	5/5 correct (100%)	3/5 correct (60%)	8/10 correct (80%)
Total	15/15 correct (100%)	12/15 correct (80%)	27/30 correct (90%)

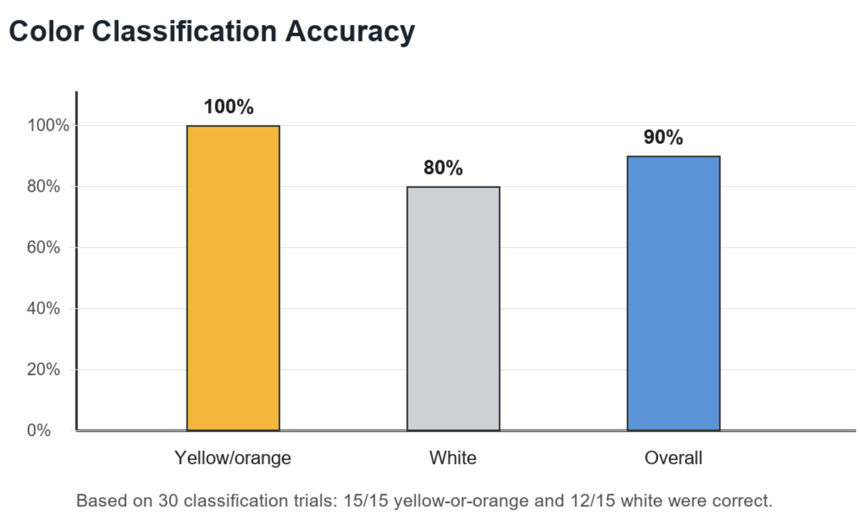


Figure 4: Classification accuracy from the 30-trial mixed-ball test.

The system met the revised overall color-classification requirement with 27 correct classifications out of 30 trials. However, the class-level results show an important limitation:

yellow/orange balls were classified correctly in all trials, while white balls were correct in only 12 of 15 trials. The error pattern is consistent with the black printed logo on the white balls affecting the sensor when the logo faces downward toward the TCS34725.

3.2 Sorting Cycle Time

The program was analyzed and timed for the standard sorting sequence after initialization. The initialization delay is 5 s. After initialization, a standard cycle consists of one color read, a three-servo motion to the sorting position, gripper opening and closing, a return motion, and a fixed delay before the next color check. Table 5 summarizes the timing result.

Table 5: Timing summary for the final control loop.

Item	**Measured/estimated time**	**Interpretation**
Power-on initialization	5.0 s	Startup overhead before repeated operation
One standard sorting cycle	Approximately 4.3 s	Normal repeated cycle time
Approximate continuous throughput	About 14 balls/min	Informational

3.3 Power Stability During Servo Operation

The final prototype was measured in three operating conditions: idle powered state, holding or hesitation state, and active servo motion. The measurements in Table 6 were taken while the servo subsystem behaved normally. No servo stall, abnormal jitter, or controller reset was observed in the measured operating cases. Figure 5 summarizes the rail-voltage measurements, and Figure 6 shows the oscilloscope photos from the tests.

Table 6: Servo supply rail voltage measurements during normal operation.

Condition	**Max (V)**	**Min (V)**	**Average (V)**	**Observed behavior**
Idle powered state	4.250	4.062	4.160	Normal
Holding/hesitation state	4.187	3.937	4.075	Normal
Active servo motion	4.125	3.812	3.989	Normal

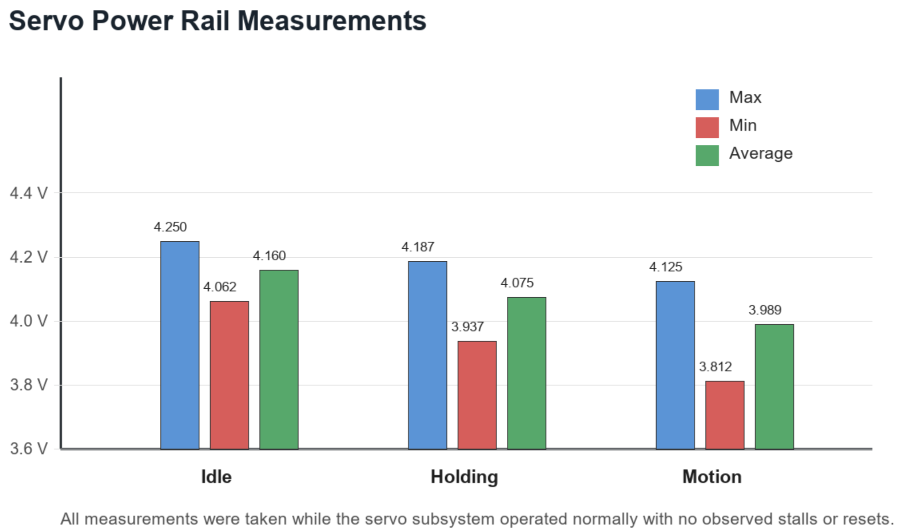
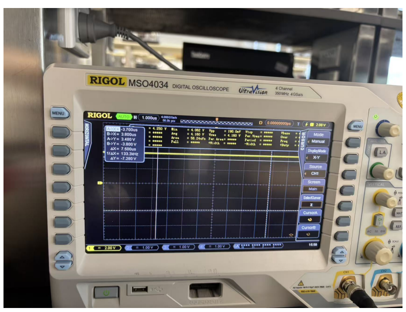
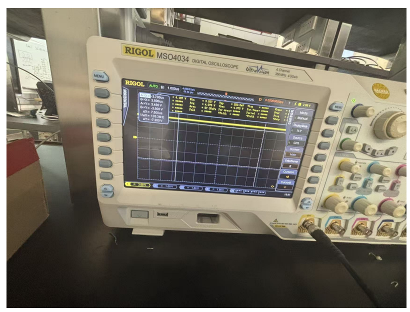


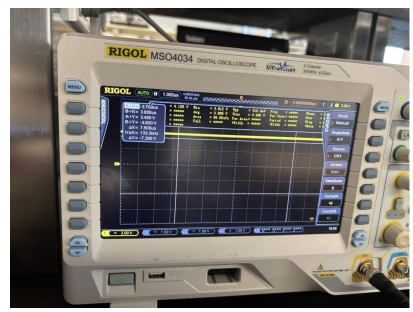
Figure 5: Measured maximum, minimum, and average servo rail voltage under three operating conditions.



(a) Idle powered state



(b) Holding/hesitation state



(c) Active servo motion

Figure 6: Oscilloscope photos from the power verification measurements.

The data show that the final prototype remained functional with the tested battery rail, including a minimum observed voltage of 3.812 V during active servo motion. This result is acceptable for describing the demonstrated prototype. A future integration of the regulator PCB or another higher-current regulated supply would likely improve repeatability, but it is not part of the verified build reported here.

3.4 Requirement Verification Summary

Table 7 summarizes the verification status of the final high-level requirements.

Table 7: Summary of final requirement verification.

Requirement	Verification method	Result	Status
Color classification	30 mixed-ball classification trials	27/30 correct overall; yellow/orange 100%, white 80%	Pass overall; class imbalance noted
Sorting action	Observed commanded left/right/reject actions during normal cycles	White directed left, yellow/orange directed right, unknown handled by reject/drop logic	Pass in implemented logic
Timing	Standard loop timing after 5 s initialization	Approximately 4.3 s per cycle	Pass
Power and motion reliability	Voltage measurement during normal servo operation	Minimum 3.812 V during active motion; no observed stall, abnormal jitter, or reset	Pass for final prototype; original regulated-5 V requirement not verified

4 Costs

Table 8 lists the major parts used or designed for the final prototype. Retail prices and actual paid prices are separated because some parts were reused from the lab or supplied without direct team cost. This table is the most meaningful cost measure for reproducing the demonstrated prototype.

Table 8: Major parts cost for the final prototype.

Component	Qty.	Retail (RMB)	Actual paid (RMB)	Notes
Arduino Uno R3	1	35.00	35.00	Controller
TCS34725 color sensor	1	28.00	28.00	Color sensing
MG90S micro servo	5	50.00	50.00	Four arm servos plus feeder servo
3-cell battery holder	1	10.00	10.00	Servo battery rail
Batteries	3	45.00	0.00	Reused from lab
Acrylic sheets	2	20.00	20.00	Mechanical base
3D printing filament	0.3 kg	15.00	15.00	Custom parts
Jumper wires and bread-board	1 set	15.00	15.00	Wiring
Screws and hardware	1 set	8.00	8.00	Assembly
Custom regulator PCB	1	10.00	0.00	Power distribution board
Parts subtotal	–	236.00	181.00	–

This cost section reports direct prototype parts only. The project was completed as academic coursework, and team members were not paid wages for their own project labor; therefore, labor is not included as an actual project expense. The demonstrated prototype had an actual direct parts cost of 181 RMB and a retail replacement parts cost of about 236 RMB.

5 Conclusions

5.1 Summary of Accomplishments

The project produced a working automatic color-sorting prototype for table tennis balls. The final system uses a TCS34725 color sensor, an Arduino Uno R3 controller, and MG90S servo actuation to classify and sort yellow/orange and white balls. The prototype achieved 90% overall color-classification accuracy in the available 30-trial test set and completed one sorting cycle in approximately 4.3 s after initialization.

5.2 Limitations and Future Work

The largest technical limitation is white-ball misclassification when the black printed logo faces the color sensor. This limitation arises because the TCS34725 measures the local surface region directly above the sensor rather than a full image of the ball. Future work should collect more raw R/G/B/C data, test controlled logo orientations, and tune the threshold values using a larger sample set. A mechanical rotation stage or a multi-sample sensing routine would also reduce sensitivity to logo orientation.

A future revision should also integrate and evaluate the custom regulator PCB or another higher-current regulated servo supply to provide more voltage margin than the direct battery-rail prototype.

5.3 Ethics, Safety, and Broader Impacts

The project follows the IEEE Code of Ethics by reporting system limitations honestly, prioritizing user safety, and avoiding exaggerated performance claims [7]. The final prototype uses only local color readings from the ball surface and does not collect identifiable user data. Electrical safety concerns are limited by the low-voltage battery-powered design, but future regulated-supply testing should still include overcurrent protection and careful wiring inspection.

Mechanical safety remains important because the robotic arm and gripper move during operation. Users should keep fingers away from the moving arm, and the system should be powered off before manual adjustment. The broader impact of the project is educational: it demonstrates how sensing, embedded software, and actuation can be combined into an accessible automation system while also showing the engineering value of adapting requirements when implementation constraints appear.

References

[1] T. Hülsdünker, M. Ostermann, and A. Mierau, "The Speed of Neural Visual Motion Perception and Processing Determines the Visuomotor Reaction Time of Young Elite Table Tennis Athletes," *Frontiers in Behavioral Neuroscience*, vol. 13, p. 165, 2019. DOI: 10.3389/fnbeh.2019.00165. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fnbeh.2019.00165/full.

[2] International Table Tennis Federation, *ITTF Statutes 2026*, [Online]. Available: https://documents.ittf.sport/sites/default/files/public/2026-02/2026_Statutes_v1_consolidated_tc.pdf, Accessed: May 15, 2026, 2026.

[3] D. J. Rogers, T. McMorris, and A. Morris, "Yellow or White? A Preliminary Investigation into the Effects of Ball Colours on the Players' Speed of Response in International Table Tennis," *International Journal of Table Tennis Sciences*, no. 1, pp. 23–26, 1992. [Online]. Available: https://sasportssience.blob.core.windows.net/ijtts/IJTTS_1_pdf%20files/IJTTS_1_23_26_Rogers_Yellow%20or%20Withe.pdf.

[4] ams OSRAM, *ams TCS34725 Color Sensor*, [Online]. Available: https://ams-osram.com/products/sensors/ambient-light-color-spectral-proximity-sensors/ams-tcs34725-color-sensor, Accessed: May 15, 2026.

[5] Arduino, *Arduino Uno Rev3*, [Online]. Available: https://store.arduino.cc/products/arduino-uno-rev3, Accessed: May 15, 2026.

[6] Tower Pro, *MG90S*, [Online]. Available: https://towerpro.com.tw/product/mg90s-3/, Accessed: May 15, 2026.

[7] IEEE, *IEEE Code of Ethics*, [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html, Accessed: May 15, 2026.

Appendix A Detailed Requirement and Verification Table

Table 9 records the final requirement-and-verification matrix used to connect the revised project requirements to the reported tests. This appendix is included so the main verification section can stay concise while preserving the detailed acceptance criteria.

Table 9: Detailed final requirement and verification matrix.

Subsystem	Requirement	Verification procedure	Status
Color sensing	Classify yellow/orange and white balls with at least 90% overall accuracy in the mixed test set.	Run three rounds with five yellow/orange balls and five white balls. Record the detected class for each trial and calculate class-level and overall accuracy.	Pass
Color sensing	Return an unknown/reject behavior when the color thresholds do not match white or yellow/orange.	Run the control loop with readings that do not satisfy either threshold and confirm that the gripper opens after the standard delay rather than sending the ball to a color bin.	Logic verified
Robotic arm	Move white balls to the left output region and yellow/orange balls to the right output region.	Observe commanded servo sequences for both color classes and verify that the base servo selects the correct side.	Pass
Timing	Complete one standard sorting cycle in approximately 4–5 s after initialization.	Measure or compute loop timing from color reading through return-to-ready.	Pass
Power	Operate from the tested battery rail without servo stall, abnormal jitter, or controller reset.	Measure servo rail voltage under idle, holding, and active-motion conditions while observing servo behavior.	Pass for final prototype

Appendix B Supplemental Test Data

Tables 10 and 11 preserve the aggregate data used in Section 3. The current data set is useful for a final prototype draft, but a larger follow-up test should record raw TCS34725 red, green, blue, and clear-channel readings for each ball and logo orientation.

Table 10: Classification test aggregate data.

Round	Yellow/orange result	White result	Overall result
1	5/5 correct	4/5 correct	9/10 correct
2	5/5 correct	5/5 correct	10/10 correct
3	5/5 correct	3/5 correct	8/10 correct
Total	15/15 correct	12/15 correct	27/30 correct

Table 11: Servo rail voltage data recorded during normal operation.

Condition	Max (V)	Min (V)	Average (V)	Servo behavior
Idle powered state	4.250	4.062	4.160	Normal
Holding/hesitation state	4.187	3.937	4.075	Normal
Active servo motion	4.125	3.812	3.989	Normal