

ECE 445

SENIOR DESIGN LABORATORY

DESIGN DOCUMENT

Cyber Guandan AR Tabletop with Real-Time Visual Assistance and Action Monitoring

Team 43

FAN ZHANG (zhang427)
YUSHANG YANG (yy61)
ZIHAN ZHOU (zihanz15)
WENDA O YAO (wendaoy2)

TA: Zhanglong Yu

April 2026

Abstract

This design document presents the complete system design for *Cyber Guandan AR Tabletop with Real-Time Visual Assistance and Action Monitoring*. In its revised scope, the project is centered on a compact automatic card dealing prototype for the Chinese card game Guandan. The system focuses on automatic card feeding, four-way card distribution, card recognition during transfer, level-card selection through hardware buttons, and local display of the largest card in each dealt pile.

Unlike a purely manual dealing process, the proposed system integrates mechanical actuation, embedded control, image-based card recognition, and rule-dependent result computation into a single standalone prototype. Two decks of cards are loaded into the machine and are fed one by one through a constrained recognition channel, then routed into four output piles. During operation, the system records the recognized cards assigned to each pile and, after dealing is complete, determines the largest card in each pile according to the selected Guandan level-card rule. The final result is displayed locally on the device without requiring an external computer interface.

The design emphasizes mechanical feasibility, modular subsystem structure, quantitative requirements, and safe low-voltage operation. The final prototype is intended to provide a more efficient and structured card-dealing process while demonstrating a practical embedded-system solution that combines sensing, actuation, computation, and result presentation.

Contents

- 1 Introduction** **1**
 - 1.1 Visual Aid 1
 - 1.2 High-Level Requirements 1

- 2 Design** **2**
 - 2.1 Block Diagram 2
 - 2.2 Physical Design 3
 - 2.3 Card Feeding and Distribution Subsystem 3
 - 2.4 Card Recognition Subsystem 4
 - 2.5 Rule Selection and Maximum-Card Computation Subsystem 5
 - 2.6 Output Display Subsystem 5
 - 2.7 Power and Embedded Control Subsystem 6
 - 2.8 Tolerance Analysis 8

- 3 Cost and Schedule** **8**
 - 3.1 Cost Analysis 8
 - 3.1.1 Labor 8
 - 3.1.2 Parts 9
 - 3.2 Schedule 9

- 4 Discussion of Ethics and Safety** **9**
 - 4.1 Ethics 9
 - 4.2 Safety 10

- References** **12**

- Appendix A Guandan Rules Summary** **13**

1 Introduction

Guandan is a popular Chinese card game played by four players using two standard decks, for a total of 108 cards [1]. At the beginning of each round, the cards must be distributed evenly so that each player receives 27 cards. In practice, this dealing process is still typically performed by hand. Manual dealing is repetitive, time-consuming, and prone to mistakes such as uneven pile sizes, dropped cards, or accidental card mixing. For applications related to Guandan, users may also want the system to determine the strongest card in each dealt pile according to the currently selected level-card rule.

The goal of this project is to develop a compact automatic card dealing prototype for Guandan scenarios. The system focuses primarily on the dealing process itself rather than complete game management. It is designed to store and feed two decks of cards, recognize each card during transfer, distribute cards evenly into four output piles, allow the user to select the current level card through hardware buttons, and display the largest card in each pile after dealing is complete. Compared with manual dealing, this solution improves repeatability, reduces handling effort, and provides pile-level result output directly on the device.

1.1 Visual Aid

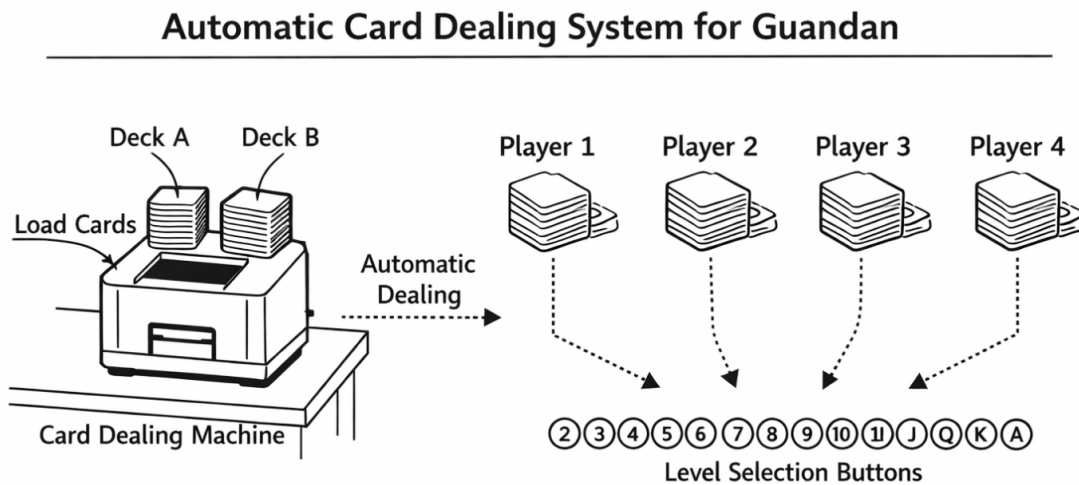


Figure 1: Use scenario of the automatic card dealing system. Two decks of cards are loaded into the machine, cards are distributed to four player piles, and the largest card in each pile is displayed locally according to the selected level-card rule.

1.2 High-Level Requirements

The project shall satisfy the following high-level requirements:

- The system shall automatically distribute two decks of playing cards (108 cards total) into four output piles, with exactly 27 cards delivered to each pile.

- The system shall recognize each card during transfer and record the recognized card identity together with its assigned output pile.
- The system shall allow the user to select the current Guandan level card through 13 hardware buttons corresponding to ranks 2 through A.
- After dealing is complete, the system shall determine and display the largest card in each output pile according to the selected level-card rule.

2 Design

2.1 Block Diagram

The revised system is divided into five main subsystems: Card Feeding and Distribution, Card Recognition, Rule Selection and Maximum-Card Computation, Output Display, and Power/Embedded Control. The card feeding and distribution subsystem is the core mechanical part of the design. The recognition subsystem identifies each card as it passes through a constrained sensing channel. The rule selection and computation subsystem keeps track of which cards belong to which pile and determines the largest card in each pile according to the currently selected level-card rule. The display subsystem presents the result locally, while the power and embedded control subsystem coordinates all sensing, actuation, and communication.

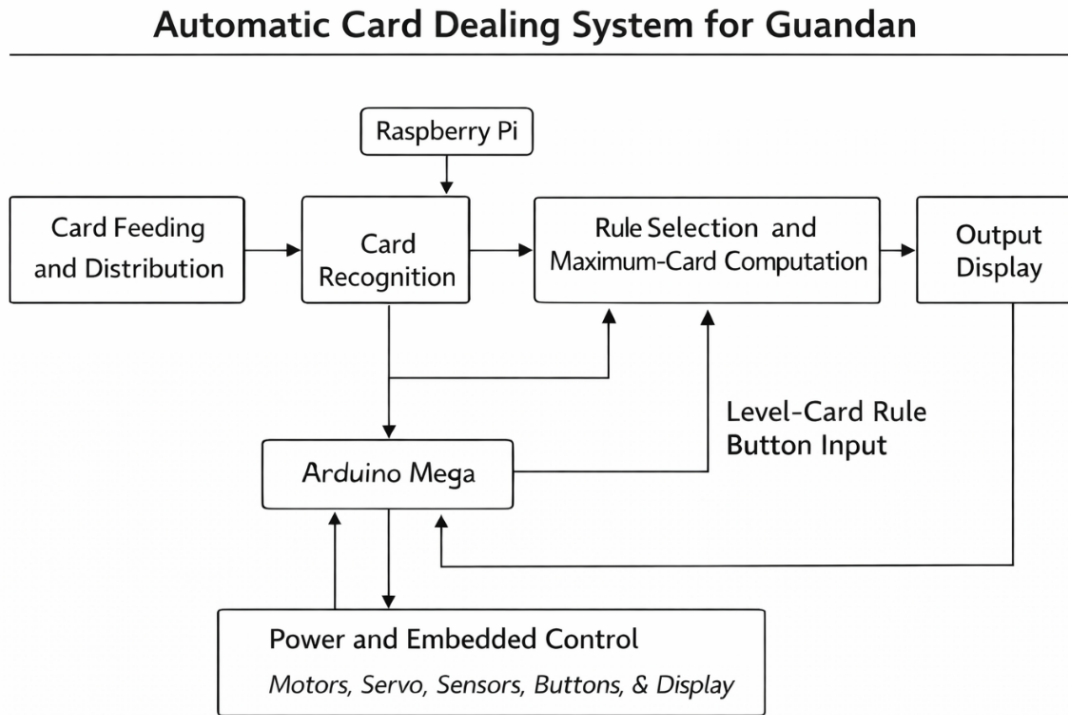


Figure 2: High-level block diagram of the automatic card dealing system.

This modular design satisfies the high-level requirements because the mechanical, sensing, computation, and display functions are separated into subsystems that can be developed and tested independently. The physical card flow is handled by the feeding mechanism, the image-based card identification is handled by the recognition subsystem, and the final pile-level result is generated by the rule engine and shown directly on local displays.

2.2 Physical Design

The prototype is installed on an acrylic baseboard of approximately $300\text{ mm} \times 200\text{ mm} \times 5\text{ mm}$. The structure is divided into three longitudinal regions: a rear storage and feeding region, a middle visual recognition region, and a front distribution and output region. A vertical hopper in the rear stores all 108 cards. Two friction wheels driven by two DC gear motors pull one card at a time from the bottom of the hopper into the recognition channel. In the middle region, the card passes through a semi-enclosed channel where an infrared trigger and an overhead camera are used for image capture. In the front region, a servo-driven rotating guide chute routes the card into one of four player piles arranged in a fan-shaped layout.

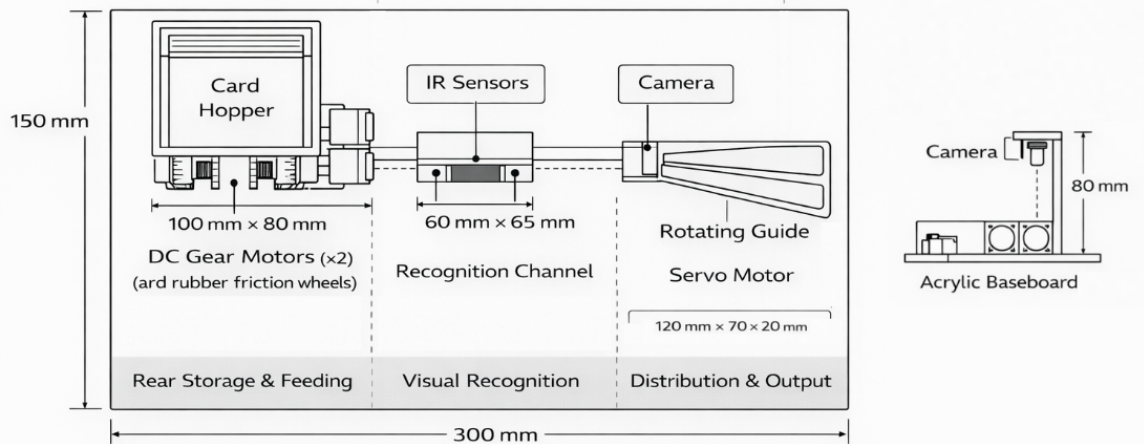


Figure 3: Physical layout of the automatic card dealing prototype.

The selected layout keeps the card path compact and mostly linear. This reduces mechanical complexity and makes it easier to maintain stable card transfer and controlled image capture. The use of a fixed recognition channel also simplifies the vision task compared with recognizing cards in an unconstrained open environment.

2.3 Card Feeding and Distribution Subsystem

The Card Feeding and Distribution Subsystem is the primary mechanical subsystem of the project. It is responsible for storing two decks of cards, feeding them one by one, and routing them into four output piles. The subsystem consists of a vertical hopper, two 12 V DC gear motors, two 40 mm rubber friction wheels, transmission shafts, couplers, bearings, a servo-driven guide chute, and four output slots.

The hopper dimensions are approximately 100 mm × 80 mm × 150 mm, which is sufficient to hold 108 cards. The hopper bottom opening is designed to be approximately 3.5–4.0 mm high so that the bottom card can be pulled forward while reducing the chance of multiple cards feeding together. The two friction wheels are driven in coordination by the two DC gear motors. After passing through the recognition channel, the card reaches a guide chute driven by an MG996R servo. The chute rotates to one of four calibrated angles, approximately 0°, 30°, −30°, and 60°, corresponding to the four player pile entrances.

This subsystem directly contributes to the first high-level requirement by physically realizing the automatic dealing function. The main design decisions are a friction-wheel feeding mechanism for simplicity and low cost, and a rotating chute for four-way routing because it minimizes the number of active actuators required.

Table 1: Requirements and Verifications for the Card Feeding and Distribution Subsystem

Requirement	Verification
The subsystem shall distribute 108 cards into four output piles, with exactly 27 cards in each pile.	Execute five full dealing trials and count the number of cards in each output slot after each trial. Pass if each pile contains exactly 27 cards in all five trials.
The subsystem shall feed cards one by one through the hopper opening under controlled indoor conditions.	Observe 100 feeding events and record the number of single-feed and double-feed outcomes. Pass if at least 95 of 100 events are successful single-card transfers.
The subsystem shall route each card to the intended output slot using the calibrated servo angles.	Perform 25 routing trials for each output slot. Pass if at least 24 of 25 cards enter the correct slot for each pile.

2.4 Card Recognition Subsystem

The Card Recognition Subsystem identifies the rank and suit of each card while it passes through the recognition channel. It consists of a 1080p USB camera, two infrared sensors, a fixed recognition channel, and image-processing software running on the Raspberry Pi 4B.

The recognition channel is approximately 60 mm long, 65 mm wide, and 10 mm high. A card entering the channel interrupts the infrared beam, which triggers image capture. The camera is mounted approximately 80 mm above the channel surface and points vertically downward. The Raspberry Pi processes the captured image using OpenCV-based operations such as grayscale conversion, thresholding, region extraction, and template matching or feature-based recognition [2].

This subsystem contributes directly to the second high-level requirement because it al-

lows the system to record which recognized card is assigned to which pile. The main design choice is to recognize cards in a constrained channel instead of a free tabletop environment. This reduces background clutter, viewpoint variation, and occlusion, thereby improving feasibility and robustness.

Table 2: Requirements and Verifications for the Card Recognition Subsystem

Requirement	Verification
The subsystem shall recognize the rank and suit of a passing card with at least 95% accuracy under controlled indoor lighting.	Feed 200 labeled cards through the recognition channel and compare the recognition output with ground truth. Pass if overall rank-and-suit recognition accuracy is at least 95%.
The subsystem shall produce one valid recognition result for each valid card-transfer event.	Record infrared triggers and recognition outputs during one full 108-card deal. Pass if every valid transfer event produces exactly one recognition record.
The subsystem shall correctly associate each recognized card with its transfer index in the dealing sequence.	Compare logged transfer indices and recognition records over a full deal. Pass if no recognition record is missing, duplicated, or misordered.

2.5 Rule Selection and Maximum-Card Computation Subsystem

This subsystem stores the recognized cards assigned to each pile, accepts the user-selected level-card input, and computes the largest card in each pile after dealing is complete. It is implemented primarily in software on the Raspberry Pi and communicates with the Arduino through a serial interface.

The system uses 13 hardware buttons corresponding to ranks 2 through A. The selected button determines the current level card. During dealing, the system records the card identities assigned to each pile. After all 108 cards have been distributed, the software compares the cards in each pile according to the selected level-card rule and determines the largest card in that pile.

This subsystem contributes to the third and fourth high-level requirements by enabling rule selection and pile-level maximum-card computation. The main design decision is to use explicit hardware rule selection instead of a software-only interface, because this is simpler to demonstrate and better matches the standalone nature of the prototype.

2.6 Output Display Subsystem

The Output Display Subsystem presents the largest card in each pile after dealing is complete. It consists of four TM1637 four-digit display modules driven by the Arduino Mega

Table 3: Requirements and Verifications for the Rule Selection and Maximum-Card Computation Subsystem

Requirement	Verification
The subsystem shall accept 13 valid button inputs corresponding to level-card choices from 2 through A.	Press each button 20 times and compare the detected selection with the expected rank identity. Pass if all 260 button presses are recorded correctly.
The subsystem shall store the recognized cards assigned to each pile throughout the full dealing process.	Run a full dealing trial and compare the software record with the actual card counts in each pile. Pass if the stored pile records contain exactly 27 cards per pile.
The subsystem shall compute the largest card in each pile correctly according to the selected level-card rule.	Use 20 prepared pile datasets with known expected answers. Pass if all computed largest-card results match the manually verified results.

2560.

Each display corresponds to one player pile. After the Raspberry Pi determines the largest card in each pile, the results are sent to the Arduino through the serial link. The Arduino then updates the four TM1637 modules so that the result is visible directly on the machine without requiring a separate computer monitor.

This subsystem contributes directly to the fourth high-level requirement by making the computed result visible to the user. The main design choice is to use simple local numerical displays rather than a more complex graphical interface, because the project is centered on the mechanical dealing system and only needs lightweight output functionality.

2.7 Power and Embedded Control Subsystem

The Power and Embedded Control Subsystem coordinates sensing, actuation, communication, and power delivery. It consists of a Raspberry Pi 4B, an Arduino Mega 2560, an L298N motor driver, two infrared sensor pairs, the servo motor, the DC gear motors, the display modules, and the wiring and power supplies.

The Raspberry Pi 4B performs high-level tasks including image capture, card recognition, pile record management, and maximum-card computation [3]. The Arduino Mega 2560 handles low-level tasks including motor control, servo control, button scanning, infrared trigger handling, and display output [4]. The two controllers communicate through a USB serial connection.

The L298N motor driver is connected to Arduino pins D8, D9, D10, and D11. The servo

Table 4: Requirements and Verifications for the Output Display Subsystem

Requirement	Verification
The subsystem shall display the largest-card result for all four piles after dealing is complete.	Run five complete dealing tests and verify that all four results are displayed. Pass if no pile result is missing in any trial.
The subsystem shall update the displayed result after receiving new result data from the Raspberry Pi.	Send 20 test result packets from the Raspberry Pi and compare the displayed values with the transmitted values. Pass if all displayed values match the transmitted results.
The displayed values shall be readable under normal indoor lighting conditions.	Ask five users to read the displayed results from a normal viewing distance. Pass if all users can correctly identify all displayed values.

is connected to D3 as a PWM output. The infrared sensors use D2 and D4 as interrupt inputs. The four TM1637 display modules use Arduino pins D22–D29, and the 13 rule-selection buttons use D30–D42. The Raspberry Pi and camera are powered by a 5 V, 3 A USB-C adapter. The motors and servo are powered by a 12 V, 2 A adapter. The two power domains share a common ground.

This subsystem enables the other four subsystems to function together as a complete prototype. The main design choice is the use of a dual-controller architecture, which separates image processing from time-critical actuator control and simplifies implementation.

Table 5: Requirements and Verifications for the Power and Embedded Control Subsystem

Requirement	Verification
The subsystem shall maintain stable controller operation during normal integrated prototype operation.	Run a 30-minute integrated test including feeding, recognition, routing, and display. Pass if neither controller resets or crashes.
The subsystem shall correctly transmit recognition and result data between the Raspberry Pi and Arduino.	Send 100 test messages in both directions and compare the received values with the transmitted values. Pass if all messages are received correctly.
The subsystem shall safely stop motor actuation when a jam or abnormal transfer condition is detected.	Create 20 jam conditions during operation and observe motor behavior. Pass if the motors stop in all 20 cases.

2.8 Tolerance Analysis

The most critical interface in the system is the mechanical card-feeding interface between the hopper and the recognition channel. If this interface fails, the system may produce double feeds, missed feeds, or unstable card motion, which would directly violate the first and second high-level requirements.

The hopper opening is designed to be approximately 3.5–4.0 mm high. A standard playing card has a thickness of roughly 0.30 mm, so the opening is much larger than a single-card thickness. Therefore, the design does not rely on thickness alone to isolate a single card. Instead, the single-card transfer behavior depends on the combination of hopper geometry, downward card loading force, and friction-wheel contact with the lowest card.

The friction wheels have a diameter of 40 mm. Assume the DC gear motor rotates the wheel at 60 rpm under load. The tangential speed of the wheel is

$$v = \pi Dn \quad (1)$$

where $D = 0.04$ m and $n = 1$ rev/s. Thus,

$$v = \pi \times 0.04 \times 1 \approx 0.126 \text{ m/s}. \quad (2)$$

A standard card length is approximately 88 mm, so the time for one card to travel through a comparable distance is

$$t = \frac{0.088}{0.126} \approx 0.70 \text{ s}. \quad (3)$$

This is slow enough to allow infrared triggering, image capture, recognition, and servo coordination in a prototype system. Even if the actual speed varies by $\pm 20\%$, the transfer time remains between about 0.56 s and 0.84 s, which is still adequate for the recognition-and-routing pipeline. This analysis shows that the chosen wheel size and feed speed are feasible for controlled single-card transfer in a low-speed prototype.

3 Cost and Schedule

3.1 Cost Analysis

The project cost includes labor and parts.

3.1.1 Labor

Assume each team member works 120 hours on the project. Assume a reasonable entry-level engineering salary of 200 RMB/hour. Following the required formula,

$$\text{Labor cost per person} = 200 \times 2.5 \times 120 = 60,000 \text{ RMB} \quad (4)$$

For a four-person team, the total labor cost is

$$\text{Total labor cost} = 4 \times 60,000 = 240,000 \text{ RMB} \quad (5)$$

3.1.2 Parts

Table 6: Estimated Prototype Parts Cost

Item	Manufacturer / Vendor	Part # / Description	Qty.	Cost (RMB)
Raspberry Pi 4B	Raspberry Pi / Generic vendor	4GB RAM version	1	350
Arduino Mega 2560	Arduino / Compatible vendor	Mega 2560 development board	1	60
USB camera	Generic vendor	1080p USB camera, autofocus	1	55
L298N motor driver	Generic vendor	L298N dual-channel driver board	1	8
TM1637 display modules	Generic vendor	4-digit TM1637 display module	4	24
MG996R servo	TowerPro / Compatible vendor	MG996R high-torque servo	1	18
12V DC gear motors	Generic vendor	12V geared DC motor	2	44
Infrared sensor pairs	Generic vendor	IR transmitter/receiver pair	2	4
Push buttons	Generic vendor	6×6 mm tactile switch	13	4
Mechanical transmission parts	Generic vendor	Friction wheels, couplers, shafts, bearings, fasteners	1 lot	54
Structural materials	Local fabrication / Generic vendor	Acrylic baseboard and 3D-printed PLA parts	1 lot	30–50
Power and wiring	Generic vendor	Adapters, jumper wires, breadboards, connectors	1 lot	53
Consumables	Generic vendor	Cards, solder, glue, heat-shrink tubing, PLA filament	1 lot	81
			Total	795–815

The estimated hardware cost is approximately 795–815 RMB, excluding labor. The single most expensive component is the Raspberry Pi 4B. No external machine shop labor is currently planned for this prototype; most structural parts will be fabricated by 3D printing and simple manual assembly.

3.2 Schedule

4 Discussion of Ethics and Safety

This project involves ethical and safety considerations because it uses camera-based recognition and electromechanical hardware near users.

4.1 Ethics

The main ethical issue is privacy. Although the system uses a camera, the sensing area is intentionally limited to the narrow card-recognition channel rather than the whole table-top or the players. The system does not perform face recognition, identity recognition, or behavioral analysis. Any captured image data should be used only for development and debugging and should not be retained unnecessarily.

Table 7: Project Schedule

Week	Wendao Yao	Yushang Yang	Fan Zhang	Zihan Zhou
Week 1	Finalize architecture and interfaces	Define rule and output format	Set up recognition framework	Draft mechanical scheme
Week 2	Implement data and communication logic	Prototype display and button input	Build and test recognition pipeline	Complete CAD and component selection
Week 3	Integrate dealing-sequence logic	Debug display and input	Optimize recognition and test accuracy	Fabricate and assemble mechanism
Week 4	Implement pile assignment and rule computation	Refine output behavior	Lead system integration and debugging	Integrate motors, servo, and sensors
Week 5	Validate logic and fault handling	Polish display and demo flow	Lead full-deal testing and optimization	Refine jam handling and mechanics
Week 6	Final documentation	Final documentation	Final integration, debugging, and report integration	Final hardware refinement and demo setup

A second ethical issue is transparency. The system computes and displays the largest card in each pile according to a user-selected rule. Therefore, the rule-selection process should be explicit and visible to the user, which is why the design uses dedicated hardware buttons.

A third ethical issue is scope. The system is only intended to automate dealing and provide pile-level result output. It does not attempt to replace the full decision-making process of the players or serve as a complete game-management system.

4.2 Safety

The main safety risk comes from the moving mechanical parts, especially the friction wheels and servo-driven guide chute. Users should not place their fingers near the feed mechanism while the machine is operating. The feed path should be partially enclosed where possible.

The second safety risk is electrical. The project uses low-voltage DC power supplies, but all wiring should still be insulated and secured properly. Power should be disconnected

before maintenance or mechanical adjustment.

The third safety risk is mechanical jamming. If a card jam occurs and the motors continue to drive, the mechanism may be damaged or may create an unsafe condition. For this reason, the system should stop operation whenever a jam or abnormal transfer condition is detected.

Overall, the design is sufficiently safe for prototype use because it limits sensing scope, uses low-voltage electronics, and includes procedures for stopping operation during abnormal mechanical behavior.

References

- [1] Pagat. "Guan Dan – Card Game Rules," Accessed: Apr. 8, 2026. [Online]. Available: https://www.pagat.com/climbing/guan_dan.html.
- [2] OpenCV. "Template Matching," Accessed: Apr. 8, 2026. [Online]. Available: https://docs.opencv.org/4.x/d4/dc6/tutorial_py_template_matching.html.
- [3] Raspberry Pi. "Raspberry Pi 4 Model B Specifications," Accessed: Apr. 8, 2026. [Online]. Available: <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/specifications/>.
- [4] Arduino. "Arduino Mega 2560 Rev3," Accessed: Apr. 8, 2026. [Online]. Available: <https://docs.arduino.cc/hardware/mega-2560/>.

Appendix A Guandan Rules Summary

Guandan is a four-player climbing card game played with two standard decks, for a total of 108 cards [1]. The players form two fixed partnerships, with teammates sitting opposite each other. In a standard round, the two decks are shuffled together and distributed evenly so that each player receives 27 cards.

The objective of each round is to play all cards in hand as early as possible. Players take turns playing legal card combinations, and a player who wishes to beat the previous play must generally play a higher combination of the same type, except for special overriding combinations such as bombs. Common legal combinations include singles, pairs, triples, three-with-two, straights, consecutive pairs, and bombs. Different local rule sets may vary slightly in the detailed definition of some special combinations.

A complete Guandan match typically progresses from level 2 to level A. The current level affects the relative ranking of cards and may also affect tribute behavior between rounds. In addition to the standard ordering of ordinary cards and jokers, the selected level card plays a special role in ranking during the round. As a result, the strongest card in a hand or pile may depend not only on face value but also on the currently selected level.

The finish order in each round is important. The first player to run out of cards is the first finisher, followed by the second, third, and last finisher. In common Guandan scoring logic, these finish positions determine how much a team advances and whether tribute exchange is required in the next round. If one team achieves a strong finishing pattern, it may advance multiple levels. Tribute is a mechanism in which the weaker side from the previous round transfers certain cards to the stronger side before the next round begins.

Although different local versions of Guandan may differ slightly in ranking details, special hands, or tribute rules, the above rule elements are sufficient to explain the game background relevant to this project. In the current project scope, only the rule elements directly related to card dealing, level-card selection, and largest-card determination are used by the prototype.