Smart Poker Table

Patrick Kollman, Brandon Hung, Zongpeng Yu

Use Case

Goal: Improve quality and speed of real-life, casino poker games

Device Purpose
- Track player stacks and pot size using CV
- Provide information to all players at the table
- Maintain game state with dealer interface

ECE Areas
- Signals
- Digital and Analog hardware
- Software

Online Poker Environment

Pot size

Player's stack

Display this information for players in real life poker games

Requirements - Software

- Dealer Interface: Learnable in less than 5 minutes
    - The interface needs to be simple and usable by a non-engineer,
    - Provide simple controls for managing the game state, camera, and motor
    - Initialization requires the number of players, their location, and initial stack size
    - After initialization, the dealer only has to press one of 3 options (fold/bet/call) for each turn

- Display for Players: 100% success rate
    - Game state, stack size, and pot size needs to be displayed to all players in real time

Requirements - CV

- Accuracy: +- 10% range of the actual stack size
    - Players only need rough estimate of other people's stack to make decisions
    - 10% wiggle room lets us trade off accuracy for speed

- Computing Speed: minimum of 10 seconds
    - Comparable or even faster than human chip reading speeds

- Field of view: minimum of 60 degrees
    - 4x4 foot wide window, 75->80 degrees for safety

- Resolution: minimum of 640x480
    - FOV = 48 in x 48 in, chip = 1.53 in * 0.209 in, minimum resolution of 31 x 460

Requirements - Hardware

- CPU: 4 GB of RAM, 1 GHz, high and low level interface
    - Memory and speed for CV (1 to 5 Mbyte arrays)
    - Low-level control for motors, high-level for game state tracker

- Positioning Accuracy: +/- 7.5 degrees
    - +/- 4.5 inches of the actual reference

- Position Convergence Speed: align camera within 5 seconds.
    - Poker rounds take from 10 seconds to 5 minutes
    - Speed will outpace an average poker round

Technical Challenges

- Motor Positioning: Speed and Accuracy
    - Rotary encoder with a resolution of 1024 gets within +/- 0.22 inches of target
    - Requires finely tuned gains or good control scheme

- Software
    - Integrating UI controls to the motor and camera functions

- Stack Counting
    - Possible misalignment and noise
    - Poor lighting can influence colors

- Keeping track of a live state of the poker game
    - Will need to simulate a poker match using data from the real life game

Solution Approach

- Current approach: modify a poker mat
    - Dealer controls software and updates game state
    - Mat attached to motorized turntable with camera and monitor
    - Turntable rotates towards stacks and reads stacks using CV
    - Monitor will display stack sizes and pot size
    - Basic button-based UI for the dealer to use to control the state of the game

- Gameplay
    - Dealer starts a round, camera sweeps to the first player to act
    - Dealer enters player's action into the computer and goes to next player
    - Repeats until the round is over

Testing, Verification, and Metrics

- Hardware:
    - Motor
        - Choose designated locations, and ensure the camera can accurately rotate to +/- 4.5 inches of the actual reference
    - Camera
        - Without using the motor, verify the camera can read stack sizes within +-10% range
- Software
    - Game State
        - Simulate a poker game using fake inputs to see if we get the expected result
    - Dealer Interface
        - Use random participants and see if they can learn our software in less than 5 minutes
- Together
    - Make sure dealer software gets correct output from the CV, camera rotates to correct player based on output from dealer software, and the display's updated information match with what's happening in real life

Tasks and Division of Labor

- Team Tasks:
    - Purchase components (camera/ motor/controller/buttons)
    - Design game state representation, control method, and table layout
    - Set up camera, turntable, and other hardware
    - Implement/test game state control
    - Integration/assembling physical table final testing
- Steve
    - Design motor and connect to camera
    - Write/test firmware for the motor
- Brandon
    - Pick CV library
    - Write/test firmware for CV (one player)
- Patrick
    - Design UI and Dealer Interface
    - Write/test code for UI

Schedule

	Task	Task Owner	Brainstorm/ Designing Phase				Development Phase					Finalization	
			Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11
Brainstorm/ Designing Phase	Purchase components(camera/ motor/controller/buttons)	All	▨	▨	▨	🟩							
	pick approporaite OpenCV library	Brandon	▨	▨	▨	🟦							
	Design UI Interface	Pat	▨	▨	▨	🟨							
	Design Motor and how it hooks up to the camera	Steve	▨	▨	▨	🟧							
	Design control for game state	All	▨	▨	▨								
	Design the Physical table (connect multiple components)	All	▨	▨	▨	🟩							
Development Phase	Set up environment for Rasberry Pi	All					🟩						
	Write firmware for the motor	Steve					🟧						
	Write firmware for the CV	Brandon					🟦						
	Write code for UI	Pat					🟨						
	Write code for Game State control	All					🟩						
	Test Motor Control	Steve							🟧				
	Test CV component for one player	Brandon							🟦				
	UI testing	Pat							🟨				
	Game State control Testing	All							🟩				
	Slack	Slack						🟥		🟥	🟥		
Finalization	Connect mutiple components together	All										🟩	
	Final testing	All										🟩	
	Slack	Slack											🟥

Conclusion

- The Smart Poker Table will revolutionize the way poker is currently played at casinos
  - Players will be able to play poker the way that it is meant to be played, with table information readily available

- Our product will help bridge the gap between online poker games and real life poker games
  - We will provide real life poker players with the same advantages online poker players normally have in their games