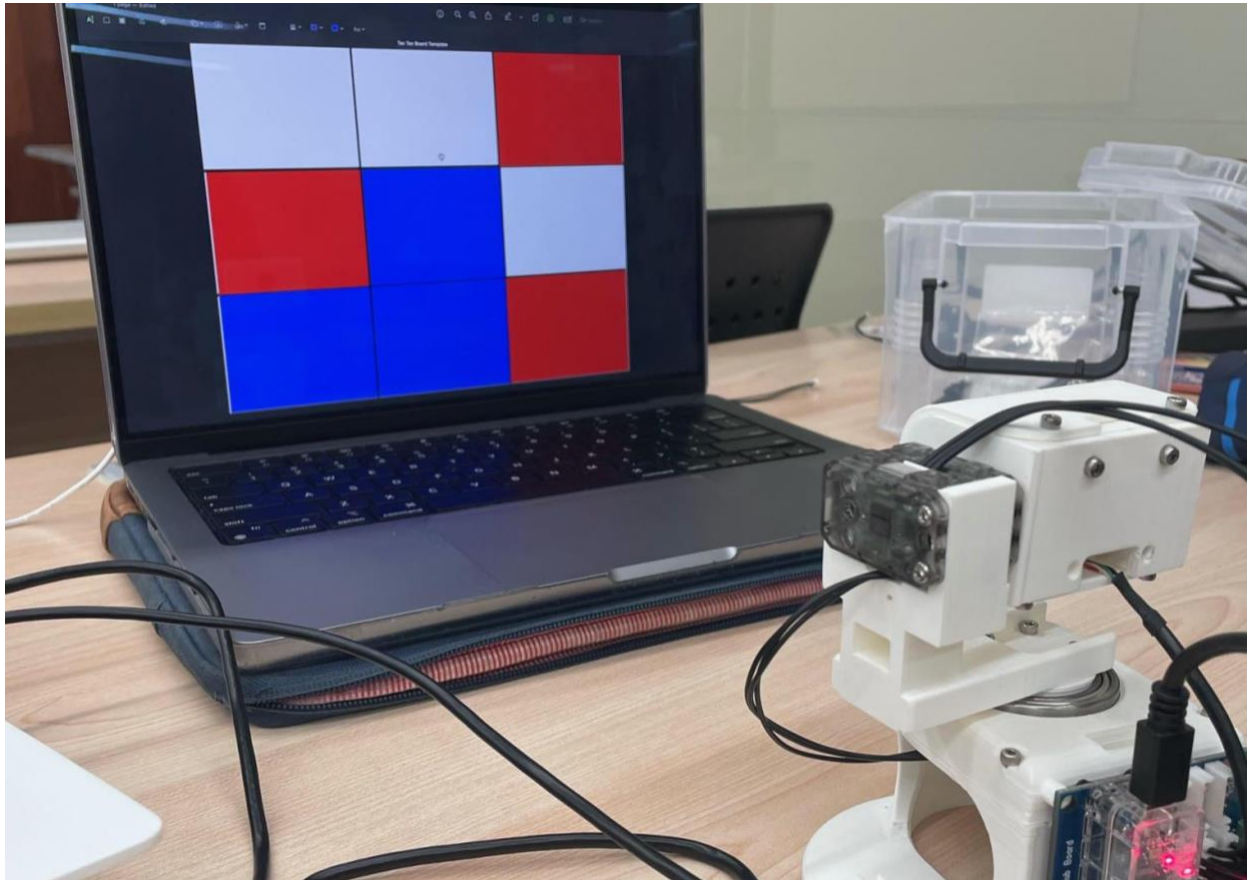


# TIC TAC TOE WITH GRETCHEN

*SNU ISP 2024*



**MIN CHOI (2024-63760)**

**JAEMIN CHO (2024-61541)**

**BYUNGHA LEE (2024-64362)**

2024.7.18

## INTRODUCTION

We explored many ideas at the start of the project, such as a guide for blind people or a system to try on clothes virtually. In the end, we decided to make Gretchen a personal game companion who can keep us company when we are bored. There were many games we were contemplating many options such as Omok and Rock Paper Scissors and after careful consideration, we gave Gretchen the ability to play Tic Tac Toe. Now Gretchen will play Tic Tac Toe with the user and determine the winner.

## ABSTRACT (INITIAL PLAN OF PROJECT)

- Blue plays the first move (User is automatically designated Blue)
- Gretchen plays the game back and forth with the user
- When a winner is found, Gretchen announces the winner and asks if you would like to play another round

## TECHNOLOGY USED/IMPLEMENTATIONS

1. Color Detection Algorithms
  - a. Color Range Filtering
    - i. Defines upper and lower bounds for each color and checks the pixel values to figure out the specific colors such as blue, red, and white.
2. Game Logic Algorithms
  - a. Winning Combinations Check
    - i. Using this algorithm Gretchen checks if the predetermined winning combinations are displayed on the board in order to announce the winner
  - b. Optimal Move Finding
    - i. This algorithm lets the robot decide which moves to make considering different conditions such as a winning combination that can be established or blocking the opponent's winning combination from forming. Also, it includes a fallback if no moves can lead to an immediate win or block.
3. Board State Detection
  - a. GridPoint Analysis
    - i. Determining the state of each cell on the board by detecting each grid point (Blue for occupied by an opponent, Red for occupied by itself/Gretchen, and white for available to place)
4. Image Annotation

- a. Drawing on Image
    - i. Lays down information such as box numbers to determine where the user or robot has placed their new move. This helps with debugging or providing feedback on code
- 5. User Interaction
  - a. User Prompts and Input Handling
    - i. These are sentences that show up to guide the user throughout the game such as “Place your first move” or “You win” to help the user input the needed information throughout the game
- 6. Image Capture and Processing
  - a. Camera Initialization and Image Capture
    - i. By using ‘camera.getImage()’ the computer creates a capture of the current state to process any changes made on the board

## DISCUSSION

Challenge	Struggle	Solution
Object detection of each box	<ol style="list-style-type: none"> <li>Initially, we planned on using O and X for placing each move like traditional Tic Tac Toe but soon we realized that it was very hard as O and X are not part of the pre-trained images in YOLO which meant we had to go through the training process which would be very time-consuming and less efficient.</li> <li>Detecting each box of the board and determining the state of each cell was very hard as it was challenging for us to divide the board into nine squares</li> </ol>	<ol style="list-style-type: none"> <li>We decided to switch to color detection instead of object detection and used RGB which allowed us to save time and distinguish between the user and Gretchen’s move.</li> <li>We used gridpoint analysis by assigning nine points on the board and naming each of them cells 1 to 9 and checking for each grid whenever analysis is needed.</li> </ol>
Interacting with user	<ol style="list-style-type: none"> <li>We planned for Gretchen to recognize the physical board in front of us and create its own replica of it and display it on the computer screen like a camera view</li> </ol>	<ol style="list-style-type: none"> <li>Instead of making Gretchen it’s platform/screen to play Tic Tac Toe on we decided that we should use the terminal to output commands for the user to replicate on the</li> </ol>

	<p>and place its moves directly on the screen so we can replicate it back on our screen. But we struggled to replicate and display a camera view, not to mention interacting with the screen.</p> <p>2. We struggled to make Gretchen recognize the new move made because live video processing is very hard and Gretchen would miss new moves made often.</p>	<p>physical board.</p> <p>2. We decided that taking snapshots after each move for analysis would be the best way for Gretchen to determine the new move made as it is straightforward and image processing is easier than video processing. After placing a move, we press enter on the terminal and prompt Gretchen to take a new photo to compare with the last one. Instead of a trained algorithm, the robot compares each cell's color with white to determine the new move made.</p>
--	--	--

## RESULT

1. Defines the color ranges for Red, Blue, and White
2. Makes a list of coordinates for each Tic Tac Toe gridpoint to create cells
3. Assign box numbers to each cell
4. Create a function that checks for a winner by analyzing the images captured for predefined winning combinations
5. Create a second function that determines the best moves by checking for a winning/blocking move and a fallback that assigns a random move
6. An image processing function that captures the board state and analyzes the color of each specific gridpoint is implemented
7. Using the functions from list number 5 and 6 Gretchen plays Tic Tac Toe with the user
8. Gretchen checks for the winner using the function on list item 4 after every move and when a winner is detected it announces and asks for you to decide whether or not to play another game

## CONCLUSION

Our code successfully implements a user vs robot Tic Tac Toe by using color detection, board state management, decision-making functions, and user interaction. It uses predefined strategies instead of machine learning which we decided was a better way to implement for a short-term project. Some next steps to improve the project would be using HSV (Hue, Saturation, Value) color space conversion for the lighting to not affect the color detection abilities and incorporating more sophisticated AI strategies to

enhance the robot's ability to play the game. Overall, our code demonstrates a functional game of Tic Tac Toe between the user and Gretchen and can make company for a bored individual like we planned.