Tic-Tac-Toe! Game

By Group11

September 15, 2024

1 Approach:

- We used YOLO model to our task as required, By which it depends on a dataset of labels that the model can train on it to detect the hand gesture required in the task

- As for the game implementation, it was decided to use Python's most popular games library called Pygame, the library will be used as a form of GUI while the game itself is being played through a data structure called list, basically Pygame will use the list as reference

2 Methodology:

2.1 Game implementation

2.1.1 Game initialization

The program starts off with a couple of global variables, each of which can be configured to change the GUI's design, for example the SQUARESIZE is used to configure the size of each square in the Tic-Tac-Toe grid.

Afterwards we initialize the Pygame screen with the configured parameters, create a variable that let the program know which player's turn is it, and create a board list that will hold all the information of the game's board, which will be later on used to draw the game

2.1.2 Drawing board

At the start of each move, the program will take the board list and use it as a reference to draw the game, each drawing will start off with turning the screen white then draw a 3x3 grid using horizontal and vertical lines.

The program then loops through the board seeking for the letters x and o, once found the program will take the index of the letter, multiply the index by the already established square size and add in half the size of the square, through that we get the center of the square grid.

through the center we can easily draw the o using the pygame circle function, while the x needs to be a bit more precise by finding the coordinates of the starting and finishing point of 2 lines, by getting the coordinates of the left or right upper or bottom corner of the square grid then add or subtract in a quarter of the square size

2.1.3   Game's loop

At every loop, pygame draws the current board

-
-

3   Results:

-
-

4   Challenges:

- The first challenge we faced was to collect the largest amount of photos to our dataset, which the model needed much data to be trained well

- the model needed to be more generalized, as we didn't want it for a personal use, so we varied in photos angles, brightness, and background