

```

import cv2
from cvzone.HandTrackingModule import HandDetector
from pynput.keyboard import Key, Controller
import pyautogui
import speech_recognition as sr

# Initialize the speech recognizer
recognizer = sr.Recognizer()

# Function to recognize speech
def recognize_speech():
    with sr.Microphone() as source:
        print("Listening for command...")
        recognizer.adjust_for_ambient_noise(source)
        audio = recognizer.listen(source)

    try:
        command = recognizer.recognize_google(audio)
        print("Command:", command)
        return command.lower()
    except sr.UnknownValueError:
        print("Could not understand audio.")
    except sr.RequestError as e:
        print("Could not request results; {0}".format(e))

# Function to click start button
def click_start_button():
    # Simulate a mouse click at a specific location on the screen for the start button
    x, y = 1466, 900
    pyautogui.click(x, y)

# Initialize video capture
cap = cv2.VideoCapture(0)
cap.set(3, 720)
cap.set(4, 420)

# Initialize hand detector
detector = HandDetector(detectionCon=0.7, maxHands=1)

# Initialize keyboard controller
keyboard = Controller()

# Main loop
while True:

```

```

_, img = cap.read()
hands, img = detector.findHands(img)
if hands:
    fingers = detector.fingersUp(hands[0])
    if fingers == [0, 0, 0, 0, 0]: # Gesture for applying brake
        keyboard.press(Key.left)
        keyboard.release(Key.right)
    elif fingers == [1, 1, 1, 1, 1]: # Gesture for applying gas
        keyboard.press(Key.right)
        keyboard.release(Key.left)
    elif fingers == [1, 1, 0, 0, 1]: # Gesture for clicking the pause symbol
        # Simulate a mouse click at a specific location on the screen for the pause symbol
        x, y = 1835, 361
        pyautogui.click(x, y)
    elif fingers == [0, 1, 0, 0, 0]: # Gesture for restart
        # Simulate a mouse click at a specific location on the screen for the edit button
        x, y = 1466, 773
        pyautogui.click(x, y)
    elif fingers == [0, 1, 1, 0, 0]: # Gesture for resuming
        # Simulate a mouse click at a specific location on the screen for the resume button
        x, y = 1466, 810
        pyautogui.click(x, y)
    elif fingers == [1, 0, 0, 0, 0]: # Gesture for exit
        # Simulate a mouse click at a specific location on the screen for the restart button
        x, y = 1466, 850
        pyautogui.click(x, y)
else:
    keyboard.release(Key.left)
    keyboard.release(Key.right)

# Check for voice command to start the game
command = recognize_speech()
if command == "start":
    click_start_button()

cv2.imshow("problem solving", img)
if cv2.waitKey(1) == ord("q"):
    break

cv2.destroyAllWindows()

```