Robot Controller and Live Video Feed Over the Internet

# Introduction

The goal of my project was to create a system where I could control a virtual character (robot) from my laptop and receive a live video feed from my smartphone, with all communication happening over the internet rather than just a local network. This setup is useful for remote monitoring and control, and it was a great way to combine cloud-based communication, real-time control, and live video streaming between devices.

# Problem Statement

Create a panel to control a virtual character and get live video feed over the internet. Prefer using cloud services for internet-based control. Do not use LAN unless you don't find any alternative. Make sure the panel is on Laptop and the camera and virtual character is on the other device (use your smartphone as device 2).

# What I Built

1. Control Panel (Laptop)  
On my laptop, I developed a graphical control panel using Python’s Tkinter library. The panel features four large, visually appealing arrow buttons (up, down, left, right) for moving a virtual robot. There’s also a “Show Webcam” button that, when clicked, displays a live video feed from my smartphone’s camera.  
  
2. Virtual Character and Camera (Smartphone)  
On my smartphone, I used the IP Webcam app to stream live video over the internet. The virtual character (robot) is visualized on a web page, which can be accessed from the phone or any device. The character’s position updates in real time based on commands sent from the laptop.  
  
3. Communication Approach  
While the ideal was to use a cloud service for all communication, I found that for real-time video and control, a combination of public web endpoints and dynamic tunneling (using tools like ngrok) was the most practical solution. Here’s how the communication works:  
- Robot Position: The laptop control panel sends the robot’s position to a Flask web server. To make this accessible over the internet, I used ngrok to expose the Flask server to a public URL. The smartphone accesses this URL to visualize the robot’s movement.  
- Live Video Feed: The IP Webcam app streams the camera feed to a public IP and port, which the laptop fetches and displays in the Tkinter panel. This allows the video to be viewed from anywhere, not just on the same WiFi.

# How It Works

1. Start the Flask server on the laptop and expose it to the internet using ngrok (or a similar service).  
2. Open the control panel (Tkinter GUI) on the laptop. Use the arrow buttons to move the robot.  
3. Start the IP Webcam app on the smartphone and begin streaming.  
4. Access the web visualization from the smartphone using the ngrok URL. The robot’s position updates in real time as you use the control panel.  
5. Click “Show Webcam” on the laptop to see the live video feed from the smartphone’s camera.

# Technologies Used

- Python (Tkinter, Flask, requests, Pillow)  
- HTML/CSS/JavaScript for the web visualization  
- IP Webcam app (Android) for live video streaming

# Challenges and Solutions

- Internet-Based Communication: Setting up real-time communication over the internet (not just LAN) required exposing my local Flask server using ngrok. This allowed devices outside my WiFi to access the control and visualization endpoints.  
- Live Video Feed: Streaming video from the phone to the laptop over the internet was made easy with the IP Webcam app, which provides a public video stream URL.  
- Synchronization: Ensuring the robot’s position was always up-to-date across devices required careful management of file paths and HTTP endpoints.

# What I Learned

- How to combine desktop GUIs, web servers, and mobile apps for cross-device projects.  
- The importance of cloud/public endpoints for true internet-based control.  
- How to use ngrok and similar tools to bridge local and global networks.  
- Practical experience with real-time data synchronization and video streaming.

# Conclusion

This project successfully demonstrates remote control and monitoring of a virtual character and live video feed using a laptop and smartphone, with all communication happening over the internet. The system is flexible, visually appealing, and can be accessed from anywhere, making it a solid foundation for more advanced IoT or robotics applications.