CodeClash

Space Station Object Detection Application Documention

Overview

CodeClash is an intelligent image-processing application where users can:

1. **Start their device camera**.
2. **Capture an image** from the live camera feed.
3. **Store** the image in **MongoDB** and **Google Drive**.
4. **Trigger image analysis** via a **Python script**.
5. **Display the analyzed image** back on the front

Frontend Interaction Flow

1. Start Camera

- Starts the webcam feed in the browser using the MediaDevices API.

2. Capture Image

- Captures a frame from the webcam feed.
- Sends the captured image (as a blob or base64) to the backend.

3. Start Analysing

- Sends a request to the backend to start the Python image analysis script.

4. Display Analysed Image

- After analysis, fetches the processed image and displays it to the user.

Feature Flow Summary

[Start Camera]
↓
[Capture Image]
↓
Upload to:
- MongoDB
- Google Drive
↓
[Start Analysing] → Runs Python script
↓
Saves Result:
- MongoDB
- Google Drive
↓
[Show Analyzed Image on Frontend]

Tech Stack

Layer	Technology
Frontend	React.js, MediaDevices API, Axios
Backend	Node.js, Express.js
Database	MongoDB
Storage	Google Drive API
Image Analysis	Python (YOLO/custom)
Deployment	Render / Vercel / Local

Problems Faced

1. Deploying Node.js and Python Together

Deploying both runtimes in a single project was challenging on platforms that support only one. I solved this by using Node.js's `child_process` to call the Python script and setting up a custom environment on Render that supports both.

---

2. Pushing Large Folders to Git

Some folders (e.g., model weights or output data) were too large to push to GitHub. This caused slow commits and push errors. I resolved it by using `.gitignore` to exclude large directories and storing heavy files externally.

---

3. Automating Best Model Selection

I wanted the system to automatically select the best-performing model for analysis. I implemented logic to load the most recent or highest-accuracy model dynamically during runtime, ensuring better results without manual intervention.

Link For Application :-

Space_Station_Object_Detection_Web_link

GitHub_Repo_Link_For_Application