Intruder Detection System

1. Introduction

This project is a real-time intruder detection system running on a local network. It uses an RTSP CCTV feed, performs deep-learning-based face recognition using InsightFace, detects unknown individuals with a time-based confirmation system, sends Telegram alerts, and provides a dashboard for live monitoring and user management.

2. Key Technologies

- **InsightFace**: Deep learning face recognition (ArcFace embeddings).
- **OpenCV** (cv2): RTSP reading, frame decoding, face detection, rendering.
- **Flask**: Dashboard, routing, MJPEG live stream.
- **Threading** & **Locks**: Smooth non-blocking frame sharing.
- **RTSP Stream**: Primary CCTV source.
- **Telegram Bot API**: Intruder alerts.
- **Local Storage:** User images + intruder captures.

3. System Architecture

The system consists of:

1. Detection Engine (main.py) – Reads RTSP frames, runs InsightFace recognition, handles intruder logic.
2. Dashboard Server (Flask) – Web UI for monitoring, managing users, retraining.
3. CameraStream Module– Threaded global camera reader.
4. Face Recognition Module (face_recog.py) – InsightFace embeddings for training + recognition.
5. Utils Module– Saving images, sending Telegram alerts.
6. Data Storage – User profiles & intruder snapshots.

4. InsightFace Face Recognition Pipeline

InsightFace replaces the older LBPH recognizer. The pipeline is:

1. Detect face (Haarcascade).
2. Align the face.
3. Extract 512-D embeddings using InsightFace ArcFace model.
4. Compare embeddings using cosine similarity.
5. If similarity < threshold → classified as Unknown.

This makes recognition far more accurate, consistent across angles, and robust to lighting changes.

5. Use of OpenCV (cv2)

- cv2.VideoCapture: RTSP and webcam frame reading.

- cv2.CascadeClassifier: Haarcascade face detection.

- cv2.cvtColor, cv2.resize: Preprocessing for InsightFace.

- cv2.rectangle, cv2.putText: Annotating bounding boxes.

- cv2.imencode: Streaming frames to dashboard.

- cv2.imwrite: Storing user & intruder images.

6. Threading and Locks

- CameraStream continuously reads CCTV frames in a background thread.

- Locks ensure safe shared access to the current frame.

- Processed annotated frames are also stored using a lock.

This enables real-time performance without blocking Flask or the detection loop.

7. Recognition Flow

1. Load all saved user images.

2. Extract embeddings using InsightFace.

3. For each RTSP frame:

   - Detect and crop face.

   - Generate embedding.

   - Compare with stored embeddings.

   - Determine identity.

4. Draw bounding boxes and labels.

5. Push processed frame to dashboard.

8. Intruder Detection Logic

- If the detected face is Unknown → start a timer.

- If unknown > N seconds (default 2s):

   - Save snapshot.

   - Send Telegram alert (cooldown enabled).

- If known face appears → timer resets.

This avoids false alarms from momentary recognition errors.

9. Dashboard Features

- Live video feed with bounding boxes.

- Add user using local device camera (not RTSP feed).

- View/delete users.

- Retrain InsightFace embeddings on demand.

- View intruder snapshots.

10. Adding Users (Device Camera Only)

Users are added using the laptop/PC webcam, not the RTSP stream. This ensures close, high-quality face samples which greatly improves InsightFace accuracy.

11. Folder Structure

project/

main.py

camera_stream.py

face_recog.py

utils.py

config.json

data/

users/

intruders/

dashboard/

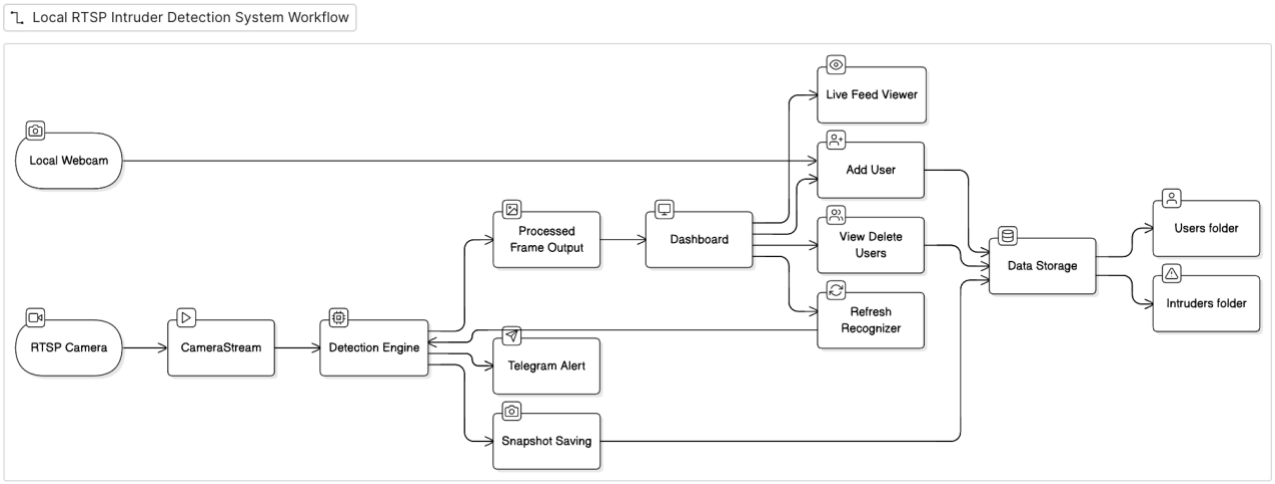
app.py

templates/

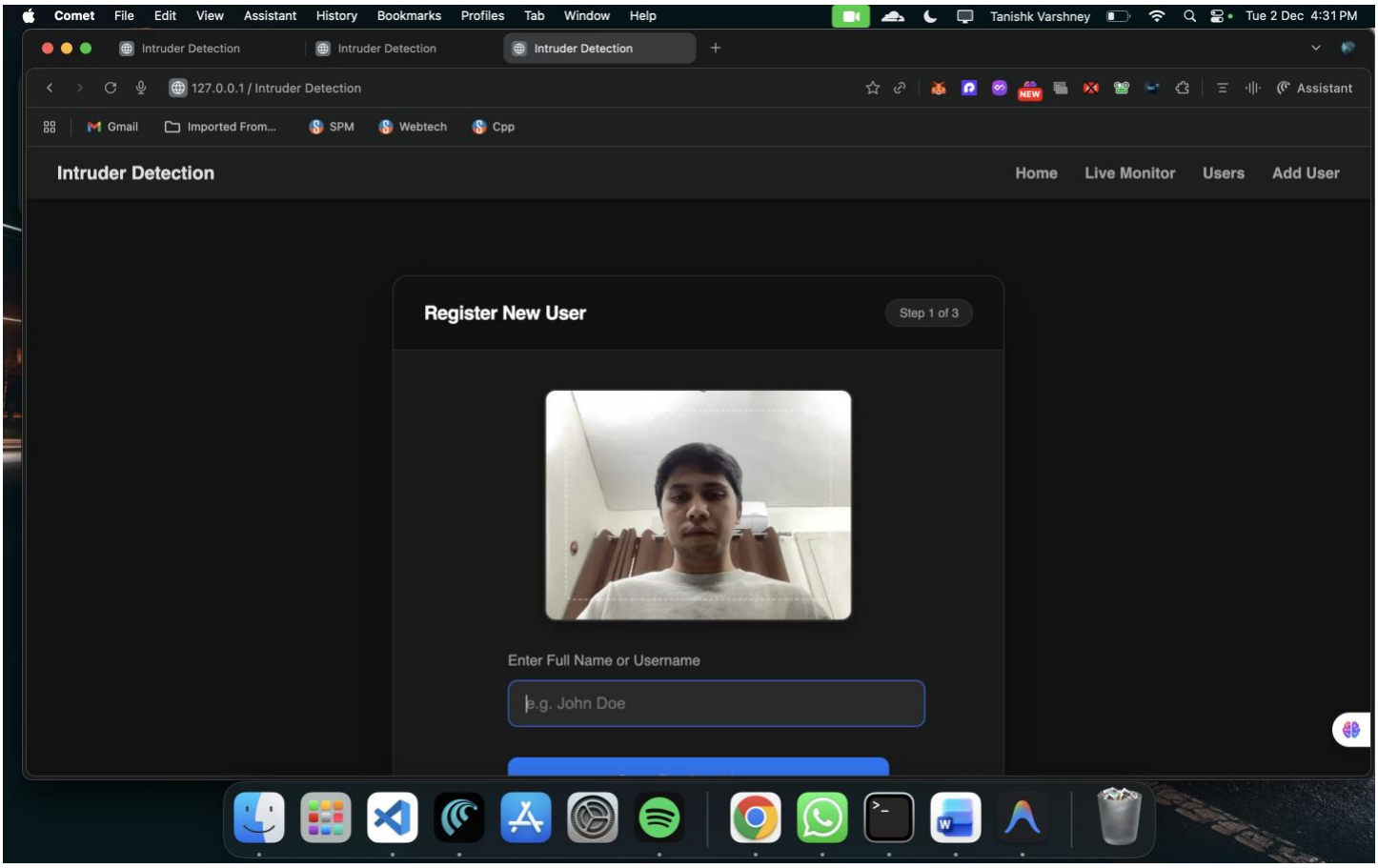
12. End-to-End Workflow

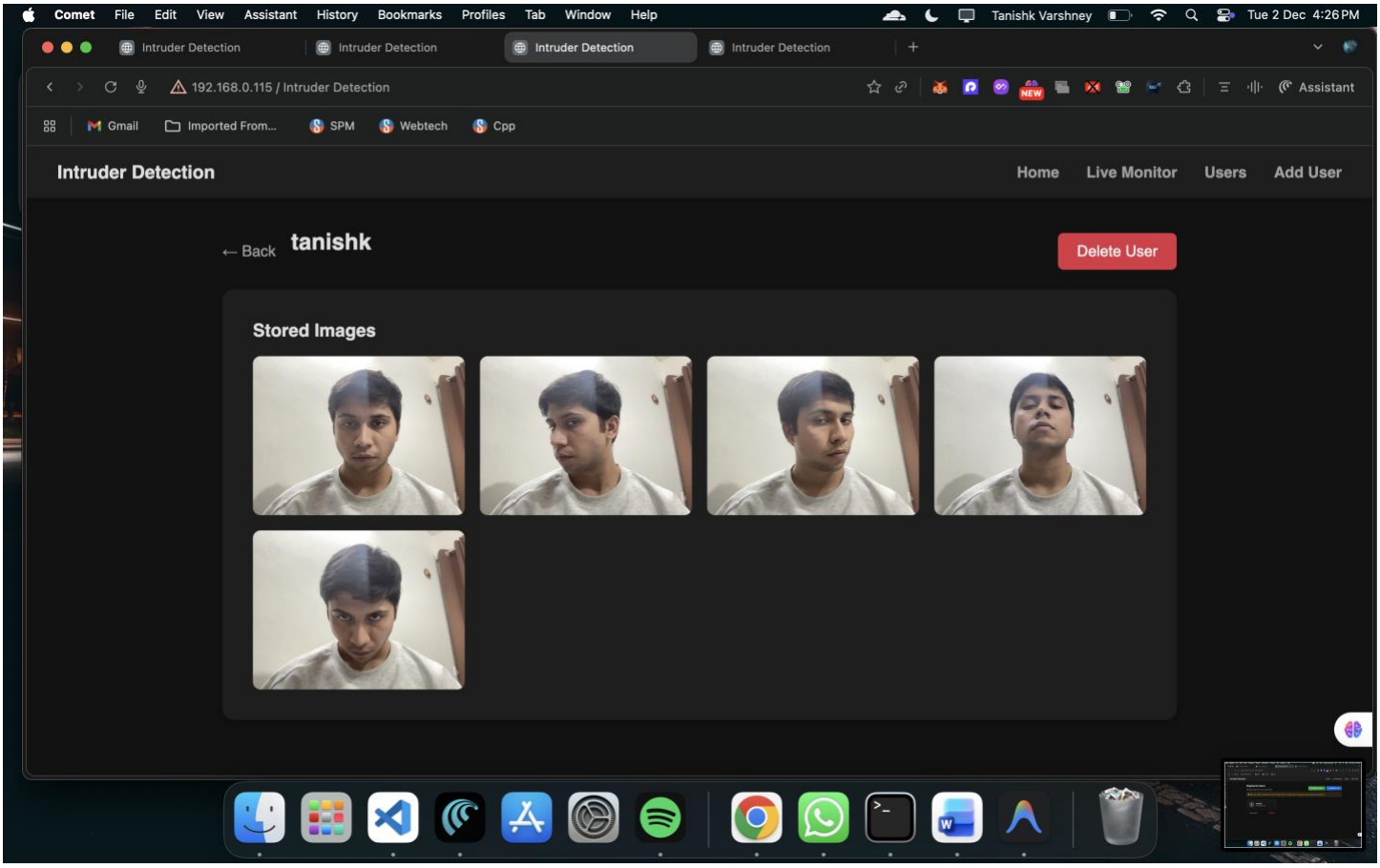
RTSP Camera → CameraStream → Detection Engine → InsightFace Recognition → Intruder Logic → Processed Frame → Dashboard.

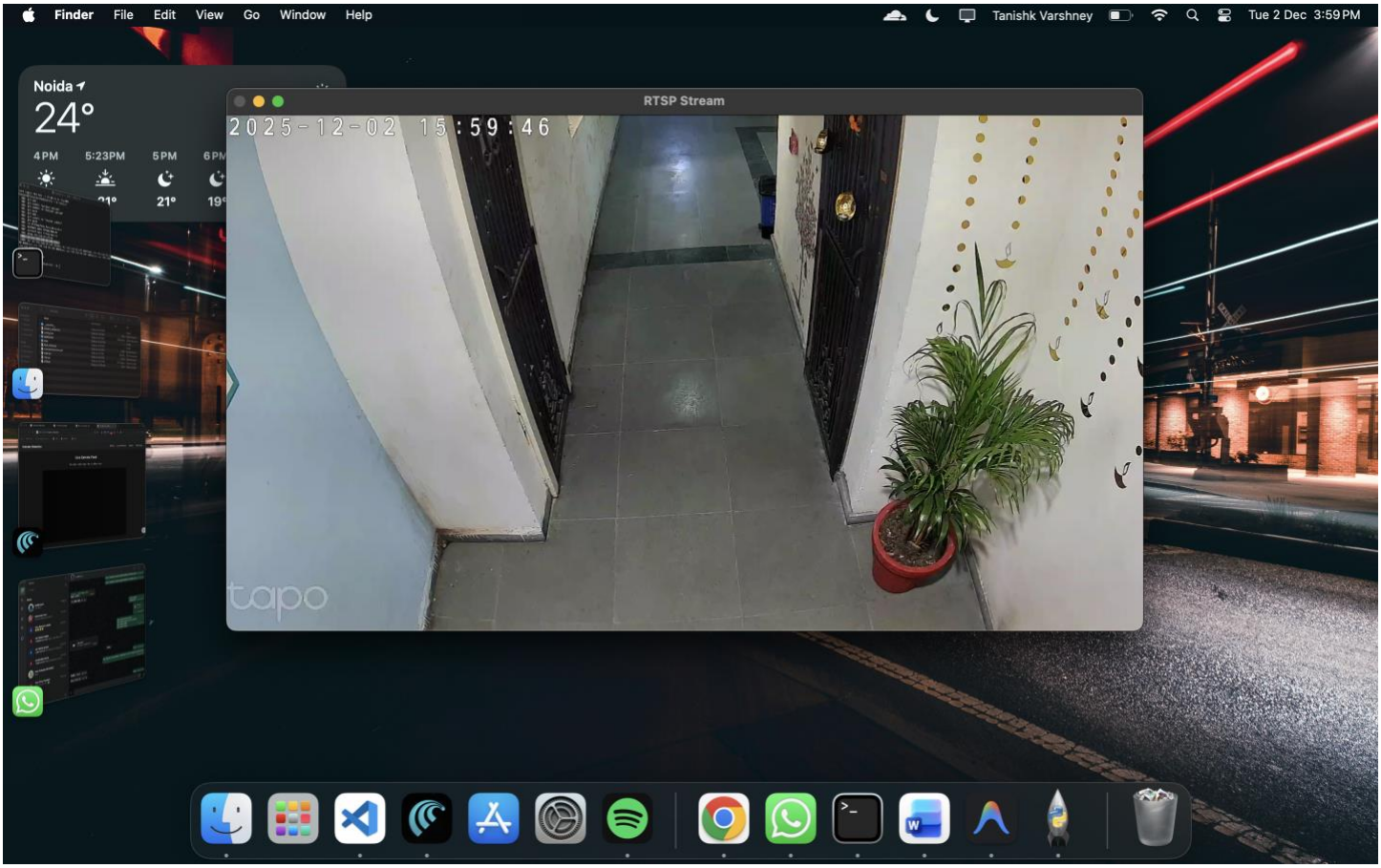
Dashboard → Add/Delete Users → Retrain → Updated Embeddings.

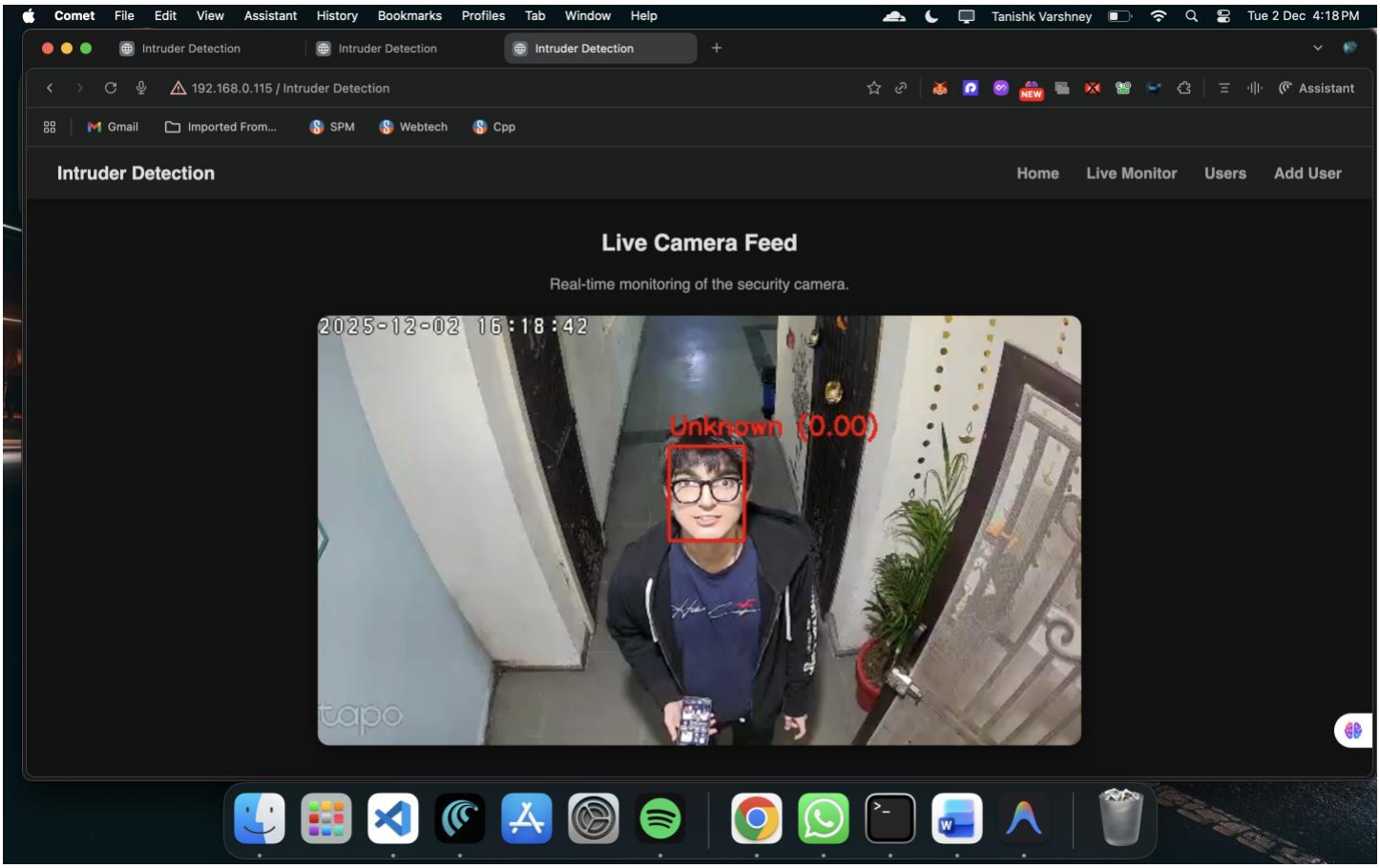


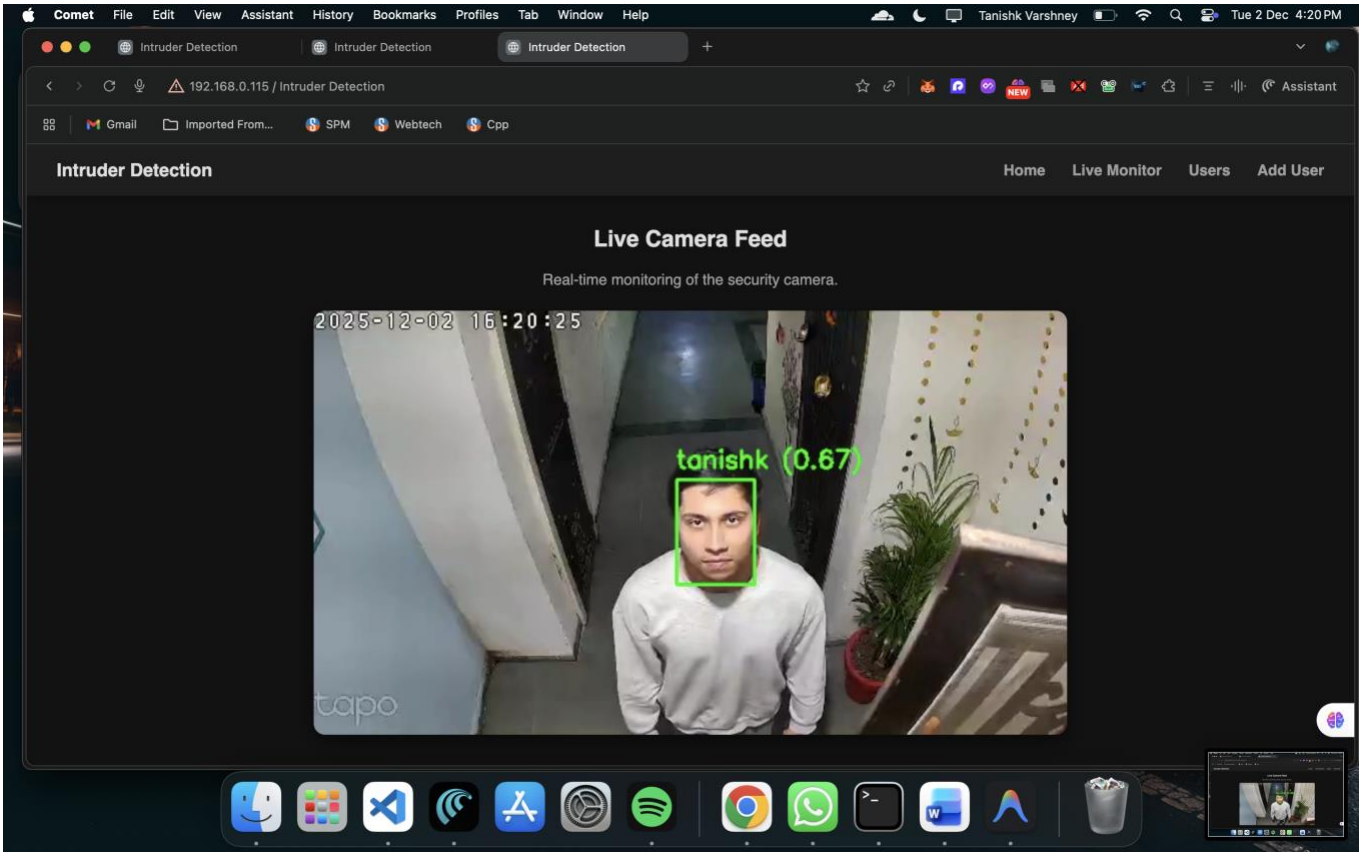
Live Demo

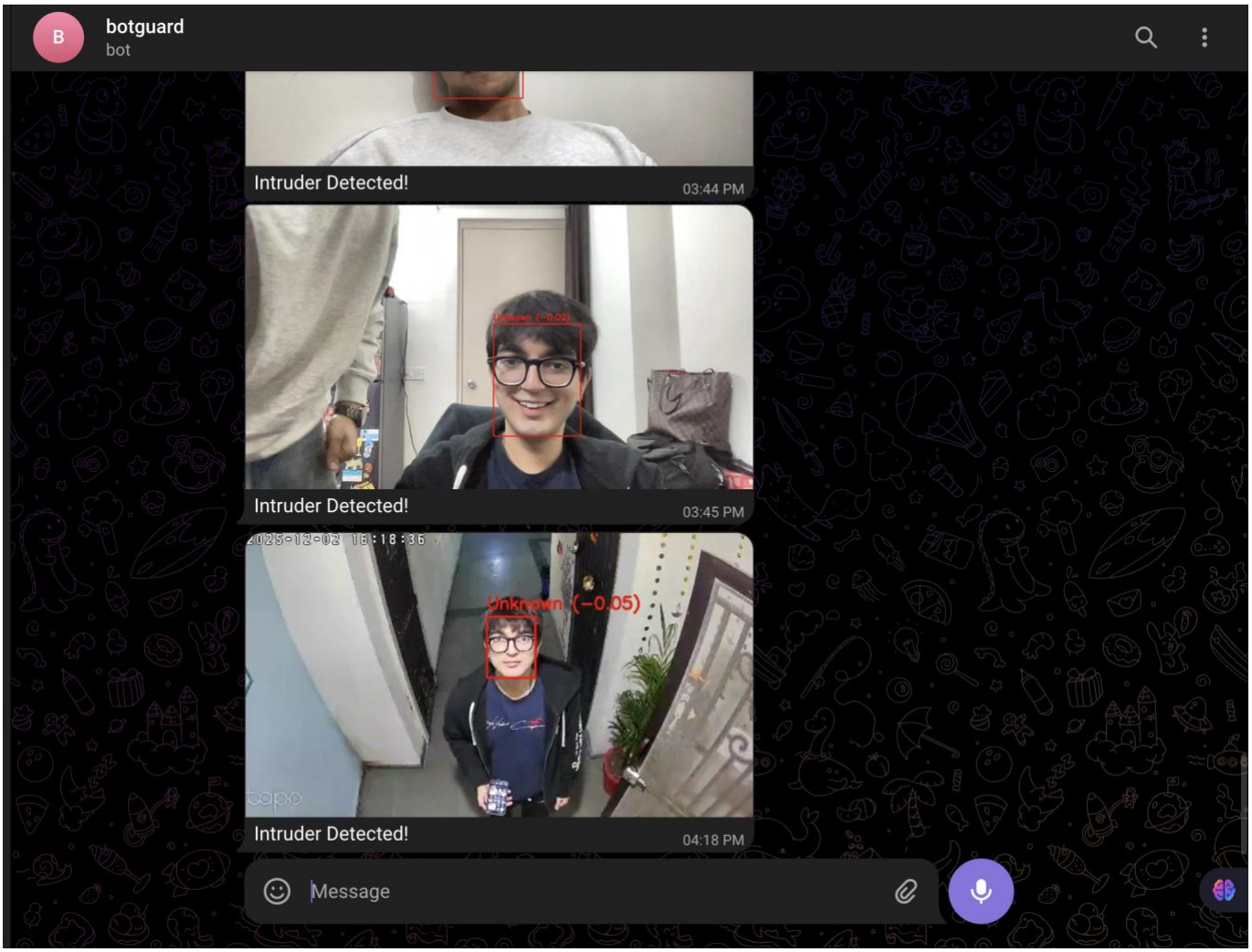

Intruder Detection
Home
Live Monitor
Users
Add User
Register New User
Step 1 of 3
Enter Full Name or Username
e.g. John Doe


Intruder Detection
Home
Live Monitor
Users
Add User
← Back
tanishk
Delete User
Stored Images


Finder File Edit View Go Window Help
Tanishk Varshney
Tue 2 Dec 3:59 PM
Noida
24°
4 PM
5:23 PM
5 PM
21°
RTSP Stream
2025-12-02 15:59:46
tapo


Comet File Edit View Assistant History Bookmarks Profiles Tab Window Help
Tanishk Varshney
Tue 2 Dec 4:18 PM
Intruder Detection
Intruder Detection
Intruder Detection
192.168.0.115 / Intruder Detection
Assistant
Gmail
Imported From...
SPM
Webtech
Cpp
Intruder Detection
Home
Live Monitor
Users
Add User
Live Camera Feed
Real-time monitoring of the security camera.
2025-12-02 16:18:42
Unknown (0.00)
tapo


Comet File Edit View Assistant History Bookmarks Profiles Tab Window Help
Tanishk Varshney Tue 2 Dec 4:20 PM
Intruder Detection
Intruder Detection
Intruder Detection
192.168.0.115 / Intruder Detection
Assistant
Gmail
Imported From...
SPM
Webtech
Cpp
Intruder Detection
Home
Live Monitor
Users
Add User
Live Camera Feed
Real-time monitoring of the security camera.
2025-12-02 16:20:25
tanishk (0.67)
tapo


botguard
bot
Intruder Detected!
03:44 PM
Intruder Detected!
03:45 PM
2025-12-02 16:18:36
Unknown (-0.05)
tapo
Intruder Detected!
04:18 PM
Message