RITSUMEIKAN

Real-Time anomalous behavior detection system

Biligsaikhan Khurtsbayar, Zhang Qian, Hyeongeun Choi

PBL2 Project, College of Information Science and Engineering

I. Introduction

Background:
Many CCTV surveillance systems that are found in public places, such as airports, shopping centers, and streets, cannot detect unusual or anomalous behavior automatically. Meaning the footage captured by these cameras must be manually monitored to identify any potential issues or security threats. However, this process can be time-consuming and may not be able to detect all potential security risks, making it less efficient and less effective as a surveillance tool.

Study Goal:
The objective of this system is to automatically detect and recognize any unusual or anomalous behavior captured by CCTV cameras and immediately notify the relevant authorities. By providing this vital information in a timely manner, the system will help the authorities to address potential security risks and take the necessary action.

II. Proposed System

The system captures live video from surveillance cameras, breaks it down into frames, and uses InceptionV3 to extract features and classify them as anomalous or not. If anomalous behavior is detected, a notification is sent to the relevant authority. The notifications will contain the precise location of the incident on a map, along with a real-time video frame of the behavior that has been identified as anomalous.

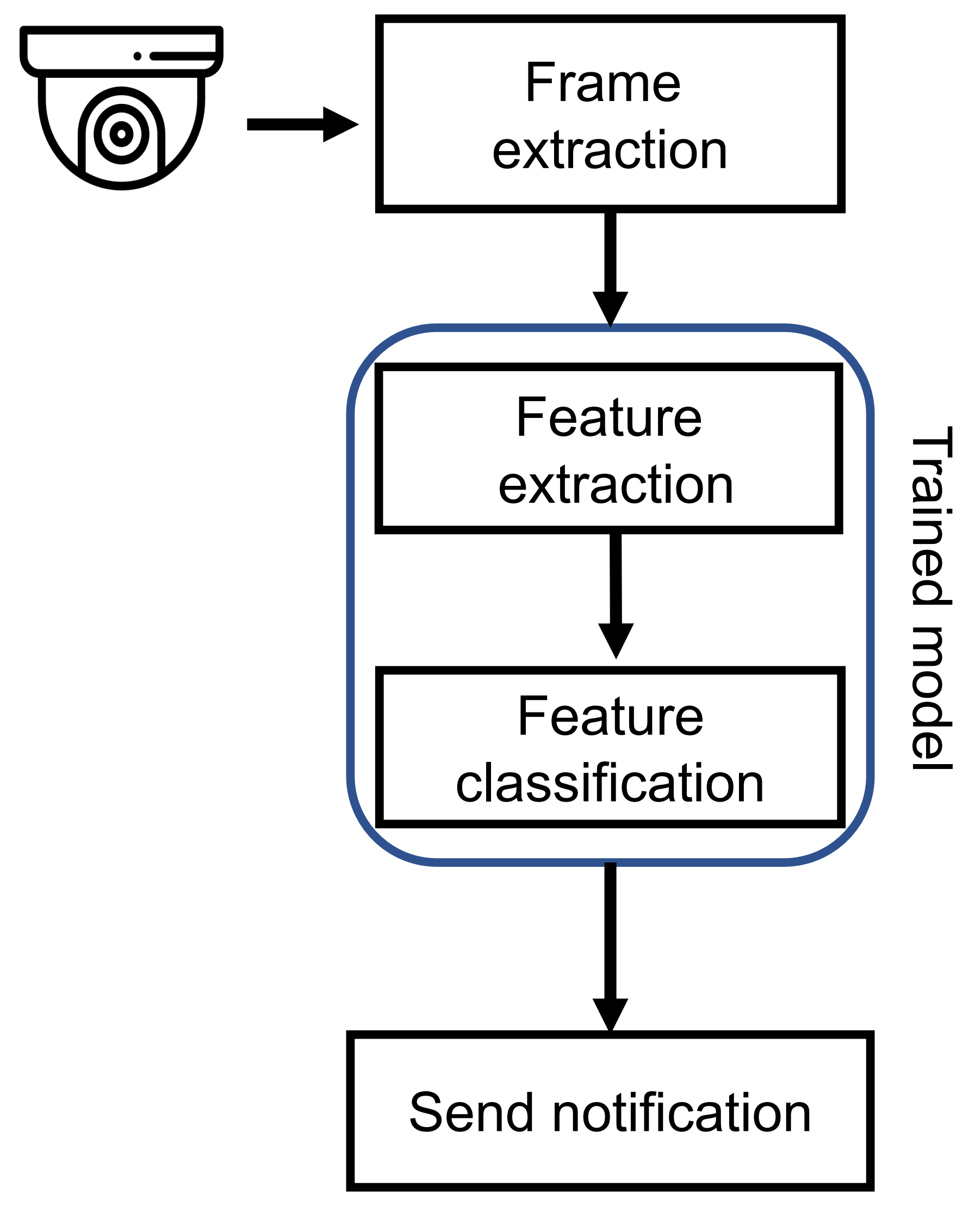


Figure 1. Proposed System

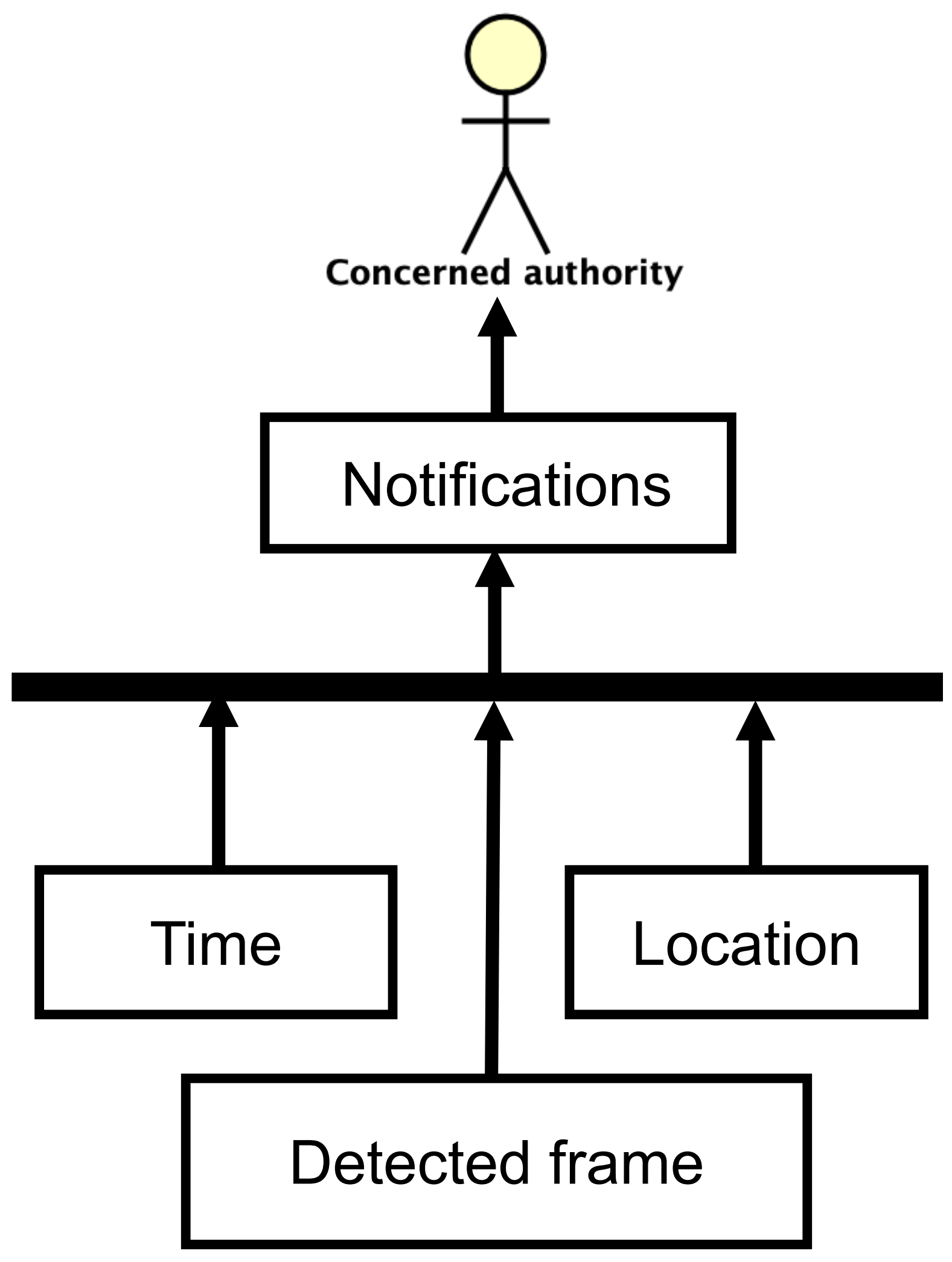
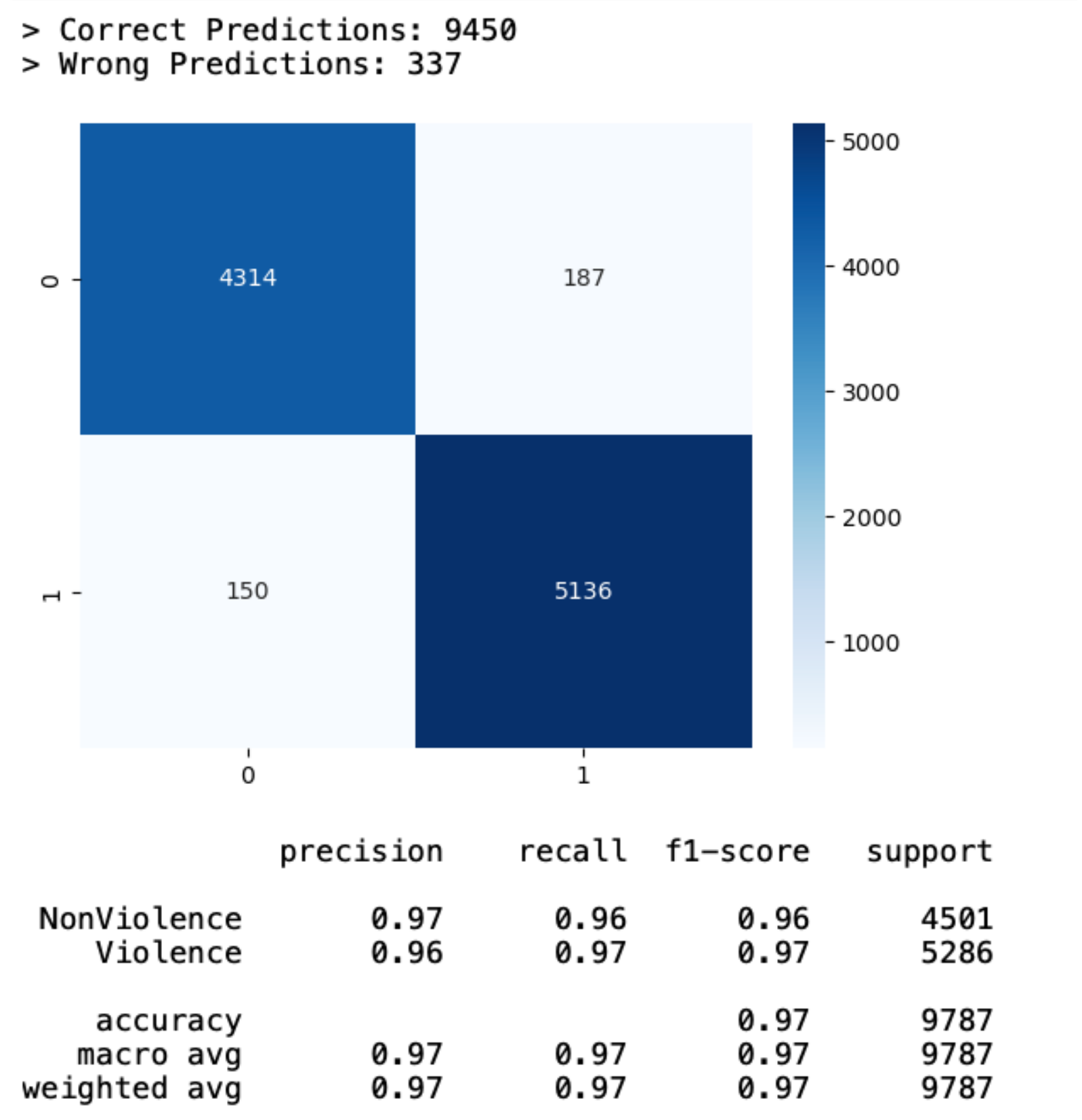


Figure 2. Notification system Design

The system includes a counter variable to prevent overloading authorities with excessive notifications that require additional investigation. The counter begins at 0 and increments each time anomalous behavior is detected.

Once the counter reaches 360 or 15 consecutive seconds of anomalous behavior is detected, the system will send the notification. The counter resets to 0 whenever the non-anomalous behavior is detected or the notification is sent to the authorities.

III. Tests



```
> Correct Predictions: 9450
> Wrong Predictions: 337
```

```
              precision    recall  f1-score   support

 NonViolence       0.97      0.96      0.96      4501
    Violence       0.96      0.97      0.97      5286

    accuracy                           0.97      9787
   macro avg       0.97      0.97      0.97      9787
weighted avg       0.97      0.97      0.97      9787
```

Figure 3. Confusion Matrix

Despite achieving high precision and recall values of 97% on the trained dataset, the model's accuracy dropped to 64% when tested manually on 100 videos not outside of the prepared dataset.

When manually tested the model also had a higher rate of false negatives than the false positives, as demonstrated by the confusion matrix in Figure 3.

IV. Developed Prototype

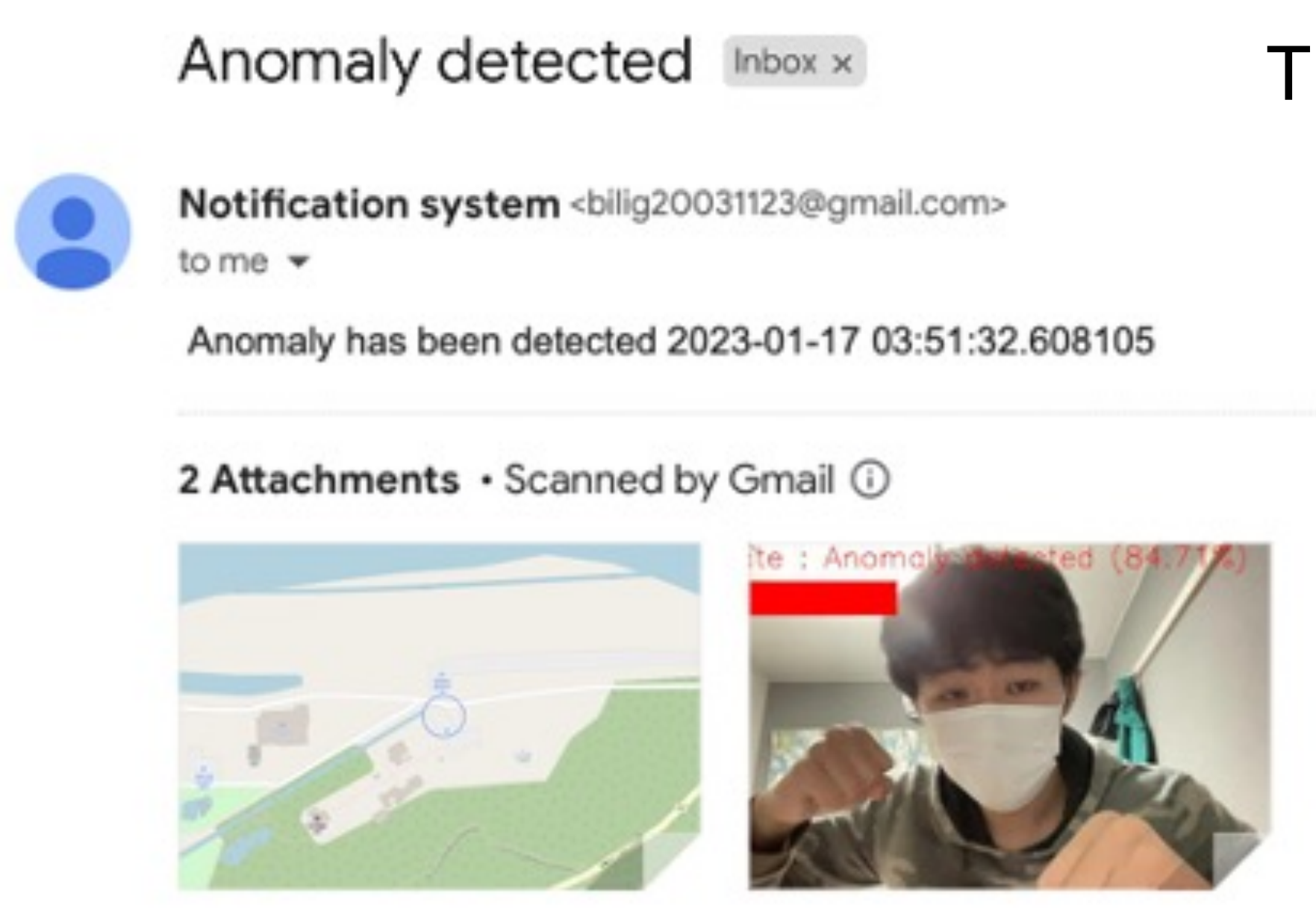
Trained model

The behavior detection model is trained on the Ubuntu operating system with a dataset including 90000 frames in total. Tensorflow and Keras packages are mainly used.

Video Capture

The video is captured using OpenCV. The program can take both live and pre-recorded videos.

Notification system



Anomaly detected Inbox ×

Notification system <bilig20031123@gmail.com>
to me

Anomaly has been detected 2023-01-17 03:51:32.608105

2 Attachments • Scanned by Gmail

The notification system uses:

- Smtplib packages to send the email
- MIME packages to attach files to the email
- Geocoder and folium package to map my location
- Selenium webdriver was used to automatically turn my map Html file to a screenshot which can be attached to the email

Used Languages – 100% Jupyter Notebook

V. Improvements

1) The model was trained on a limited dataset due to hardware constraints, which leads to inaccuracies in detecting certain scenarios. Some examples of this behavior include:

- •Incorrectly identifying a higher probability percentage in low-light areas
- •Identifying a higher probability percentage when an individual is too close to the camera
- •Failing to detect large crowd movements

To resolve these issues, it is necessary to acquire more powerful hardware to include a greater number of data points in our dataset.

2) The device's IP address was utilized to determine its geographical location. However, our system requires a higher level of precision to fulfill its purpose. To enhance the accuracy, we plan to implement the use of Google's Geolocation API.

VI. Conclusions

Most surveillance cameras only record screens and cannot detect certain behaviors. In conclusion, the convolutional neural network has made it possible for cameras to identify violence and distinguish whether it is violence or not, as well as able to send an alarm and time through email that violence was detected. Therefore, the system makes it more recognizable to people and enables rapid information delivery to security managers. Future development plans to analyze from various angles through multiple cameras and immediately send information along with location to improve it more accurately and efficiently.