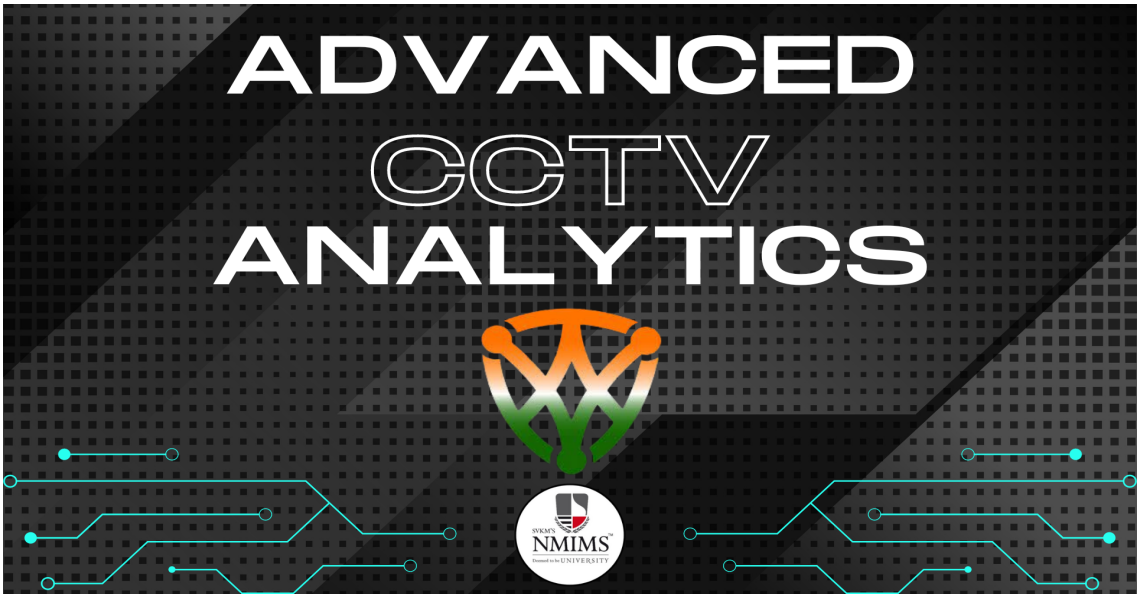


Advanced CCTV Analytics

31st July, 2023

—

Problem Statement ID: KVH 009
College Name: NMIMS
Location: Maharashtra, Mumbai
Idea ID: IDEA2547

Team Members

Sam Varghese: Team Lead, Machine Learning, MBA BTech (CE) 3rd yr

Sudhanshu Rastogi: Project Manager, MBA BTech (CE) 3rd year

Chaitanaya Kusumakar: Robotics, MBA, BTech (CE) 3rd year

Avni Bhardwaj: Website Developer, BTech (CE) 2nd year

Sanskriti Sharma: Application Developer, MBA, BTech (CE) 2nd year

Arohi Jain: Database Manager, BTech (CE) 2nd year

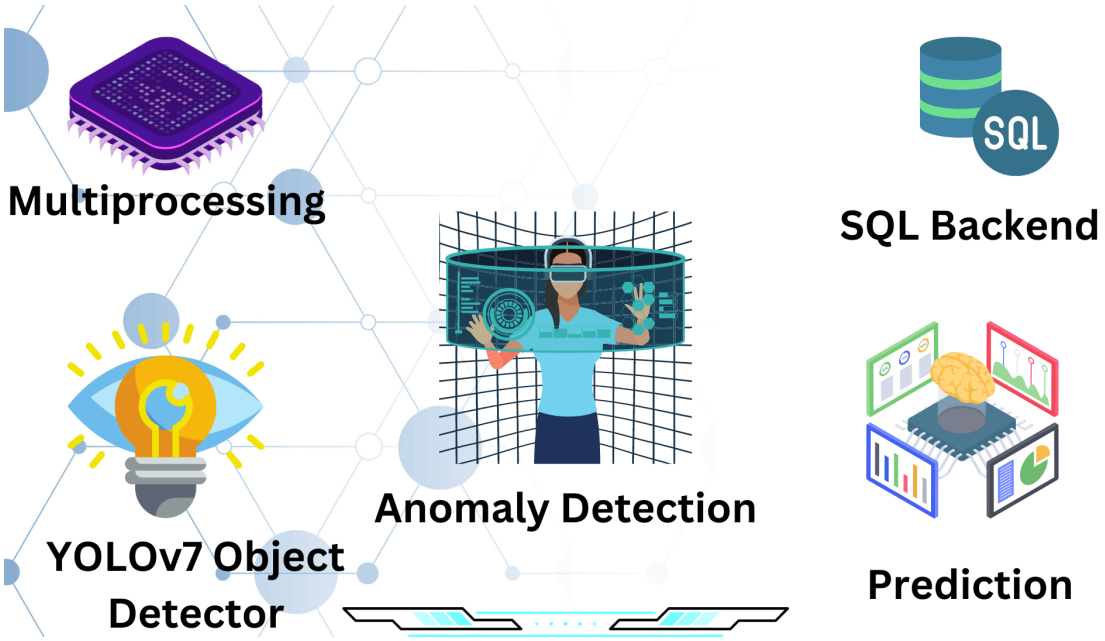
Table Of Contents

Problem Statement

Design and develop a technological solution based on live CCTV feeds, that can automatically detect incidents related to street crime, violence, burglary, theft, infiltration, unauthorized access etc. and generate alerts to the nearest Police Station. The solution should also be able to generate a report and maintain a database that includes the nature of incident/crime, location, time, level of alert (i.e., low, medium, high risk alert) etc.

Overview

We have made an Unsupervised Machine Learning algorithm along with its GUI, capable of detecting violence and accidents. It's **based on Anomaly Detection** which gives our model an added advantage of working even under new, unexplored circumstances without much training. Anomaly detection algorithms were first developed by researchers from Stanford University. Computations are enhanced by distributing tasks among multiple processors and applying the **YOLOv7 algorithm**, which surpasses all the currently known object detectors in terms of real time performance. Application also has the feature of **sending SMS messages, placing calls, etc in case of emergency alerts.**



Goals

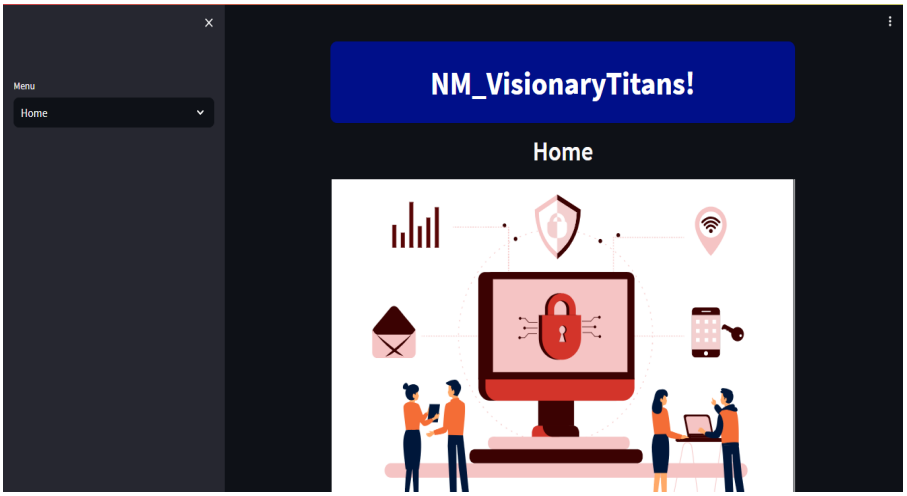
1. To develop an application that can detect crimes and accidents.
2. It should send alerts whenever any criminal activity gets detected.
3. Alerts should be sent in 3 levels: Low, Medium and High.
4. It should maintain a database of all the events detected.

Specifications

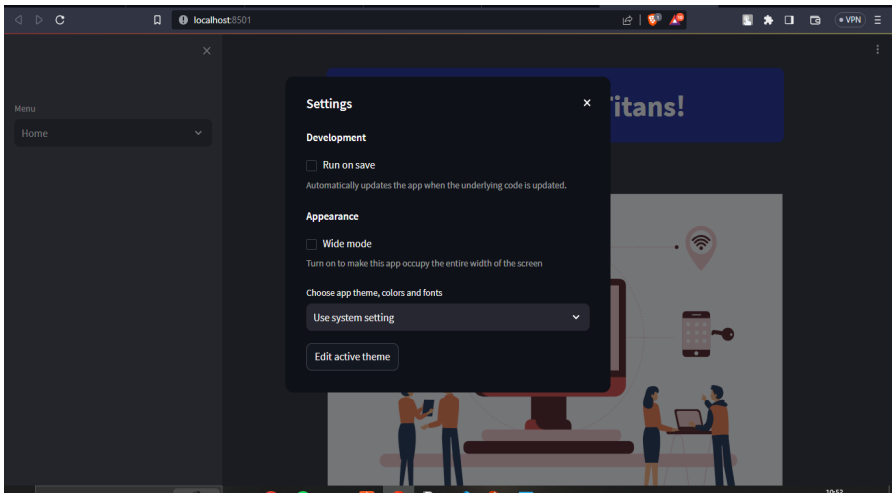
The project is made in python programming. For frontend we've used streamlit library which enables us to create a web application through python, and helps us in seamless integration with databases. MySQL has been used as a database because of its excellent speed and performance even during edge cases. For machine learning YOLOv7 object detection algorithm has been used and object tracking algorithm is custom made for excellent performance.

Streamlit: Frontend

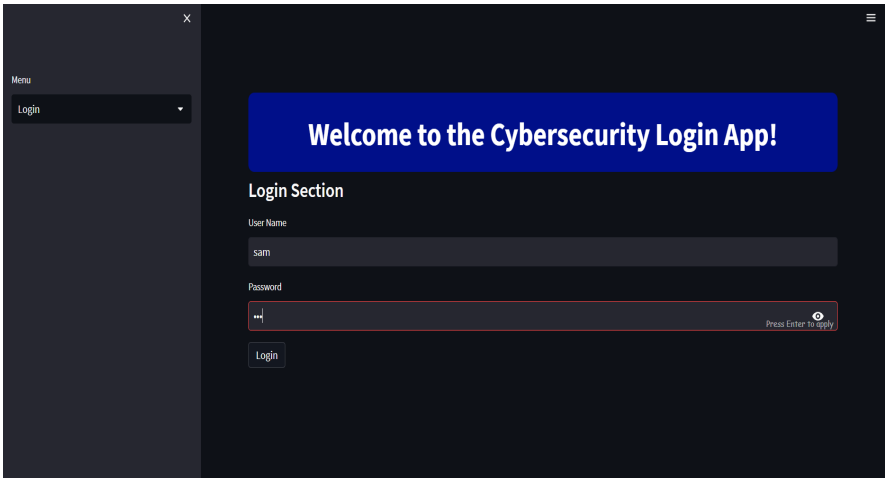
For the frontend, streamlit library has been used, which **enables seamless integration of machine learning models, and database with the web application.** The data catching also speeds up the computational pipeline.



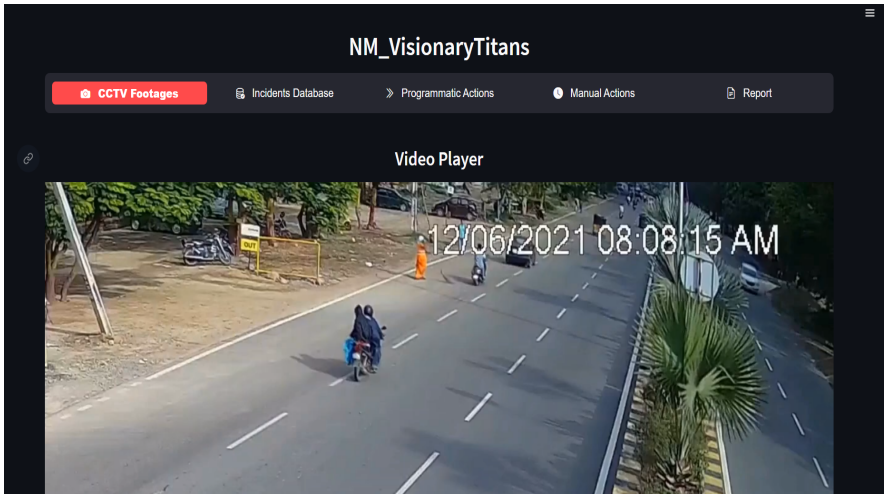
Main Page of our Application



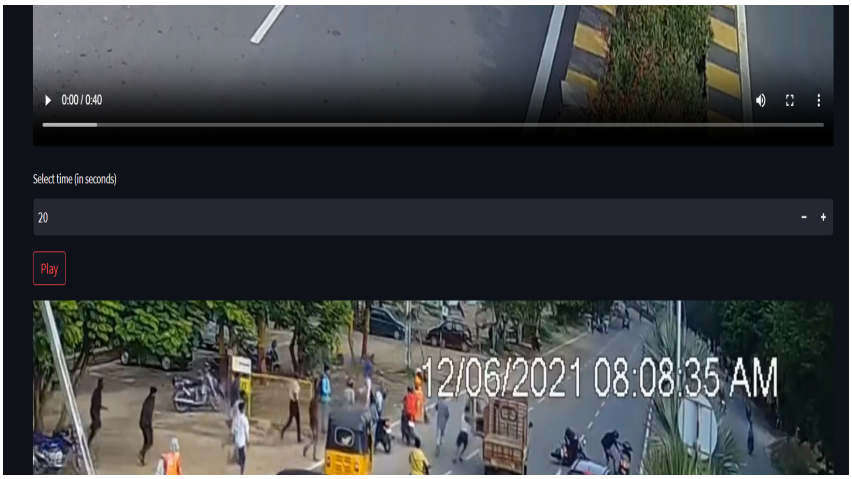
We can even change the theme, set appearance mode



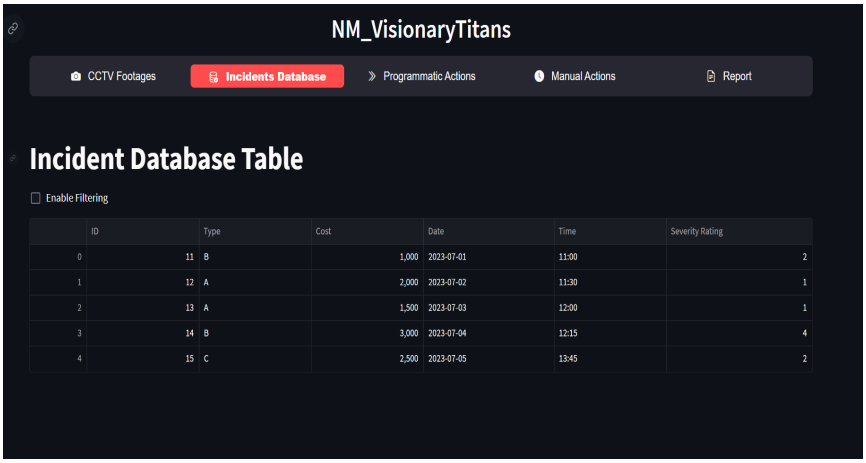
Login page



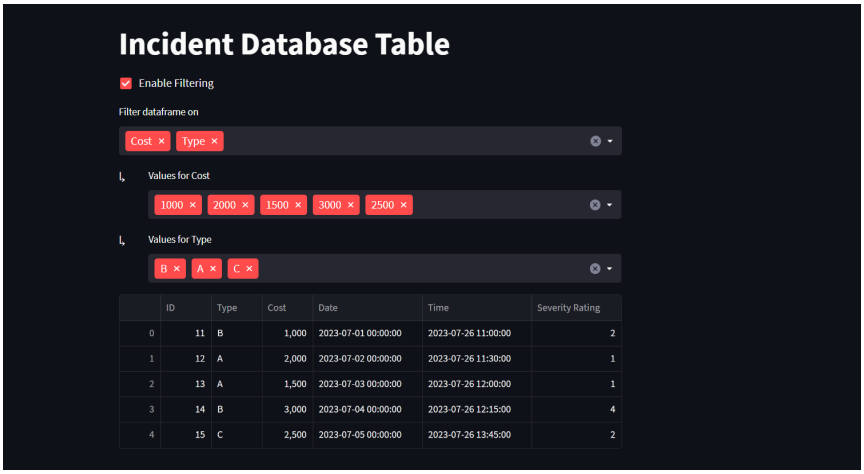
Main Page of application



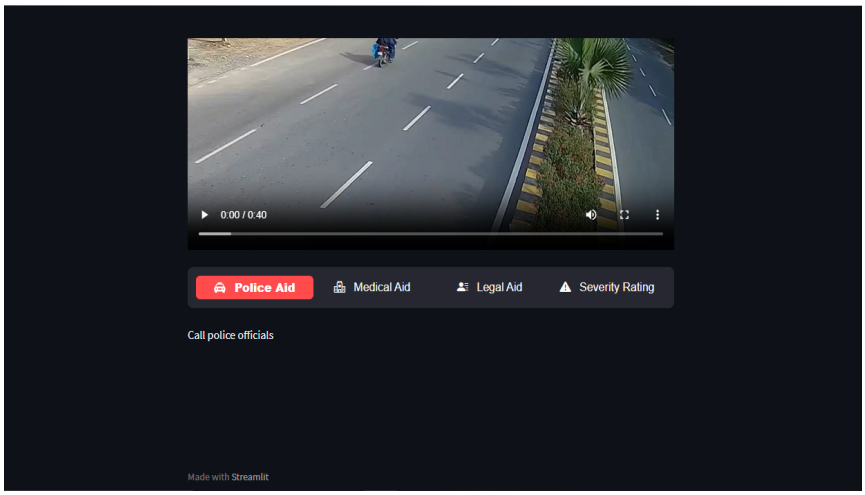
Option to view live CCTV Footage, and check recording at any given second



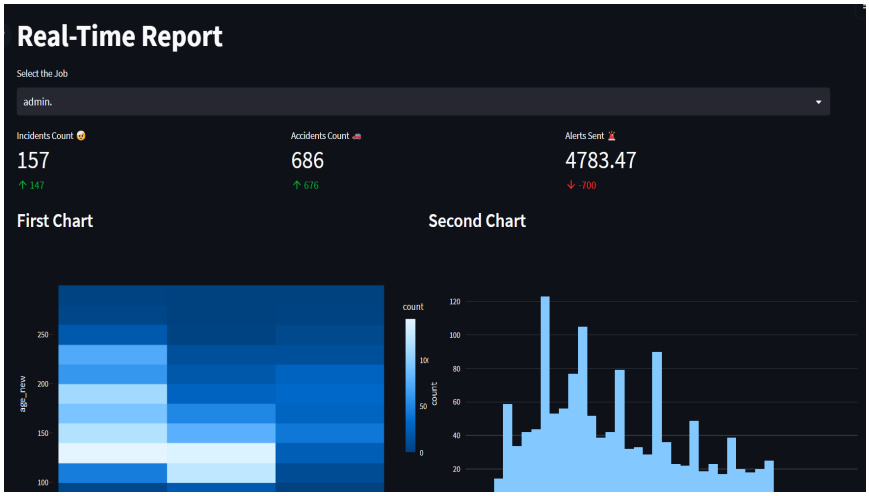
Option to view the latest database of all the incidents detected.



Feature to filter the incidents database



A manual alert triggering feature for officials in case the model fails to detect any incidents.



Real time reports, updates every second

MySQL: Backend

Using MySQL enables easy integration of this application with traditional government softwares. With its **excellent speed and ACID compilation**, it's ideal for a software targeted for use by the Government of India.

YOLOv7: Machine Learning

YOLOv7 surpasses all the known algorithms in terms of speed and performance. It has been used by Tesla, NASA, etc. Because of its excellent neural network architecture and training on a large quantity of dataset, it works quite well even in microcontrollers.

Alert System

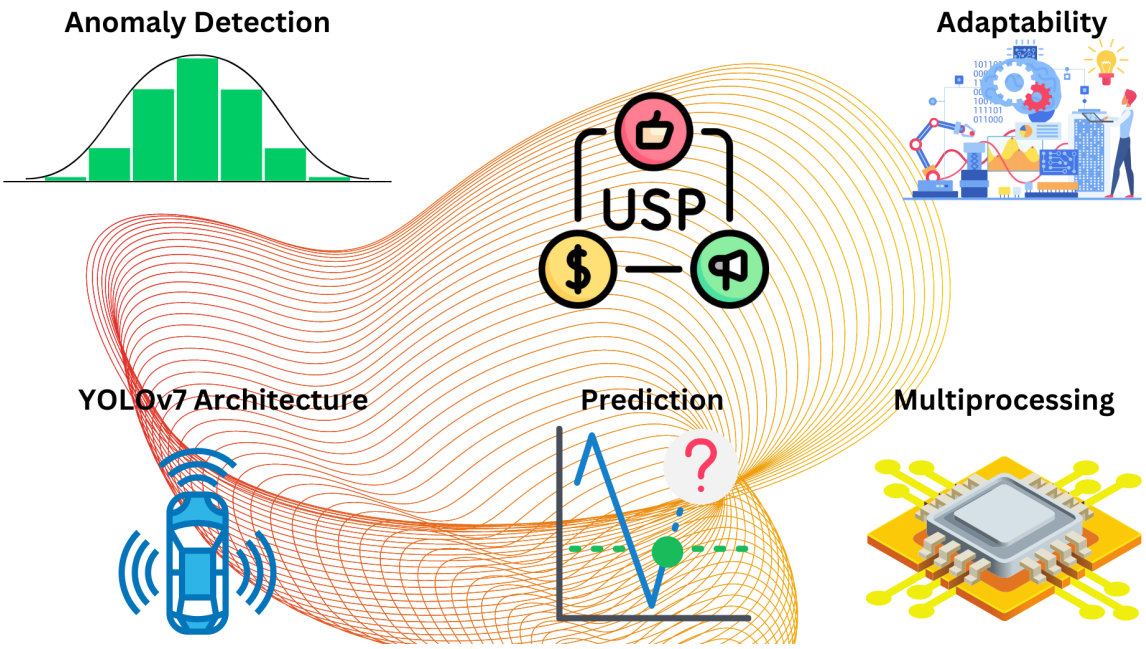
For generating instant alerts, we've utilized Twilio services which enables us to **send whatsapp messages, SMS, calls, mails, etc.** In our project we've implemented only an SMS alert system which can easily be enhanced as per the needs. The alerts are sent to the registered mobile numbers of officials and are completely secured considering that it is used by fortune companies like IBM, Airbnb, Toyota, etc.

Cyber Security

We believe that the most important aspect to be considered in a cyber security hackathon, is to analyze the overall security of the application.

1. **Authentication and Authorization**: The application only gives access to all the data and controls, once the user puts the correct username and password. This should match with the records of authenticated officials from MySQL database.
2. **Password Storage**: The password gets hashed before it's stored in the database. The hashes are compared to authorize a login instead of the original password. This gives the application an extra layer of security. Even during worst case scenarios when the data gets compromised, the original passwords do not get leaked.
3. **Third Party Components:** In order to reduce our dependence on 3rd party components, many mathematical operations for anomaly detection have been implemented from scratch. This speeds up the computation, reduces memory footprint, and secures the application.

USP



Anomaly Detection

Instead of taking the traditional supervised Machine Learning approach for tackling this complex task, we implemented Anomaly Detection algorithms for crime detection. This **helped us in reducing computational power required to perform the task, and make a flexible model capable of learning** and understanding the environment much better than other models.

Adaptability

Because of Anomaly Detection Algorithms, the model also becomes capable of understanding the trends of the area of supervision. So **if somebody tries to perform a criminal activity in a way never witnessed before, there's higher chances it will work on our model** than the traditional image classification models even without proper dataset.

YOLOv7 Architecture

Utilizing YOLOv7 architecture enables our model to be faster and more efficient than the rest of the models. This architecture was also implemented in fortune companies like Tesla, SpaceX, and even NASA. It's by far the best algorithm for real

time object detection tasks with the **ability of detecting more than 80 classes. YOLOv7 surpasses all the currently known object detectors in terms of speed and efficiency.**

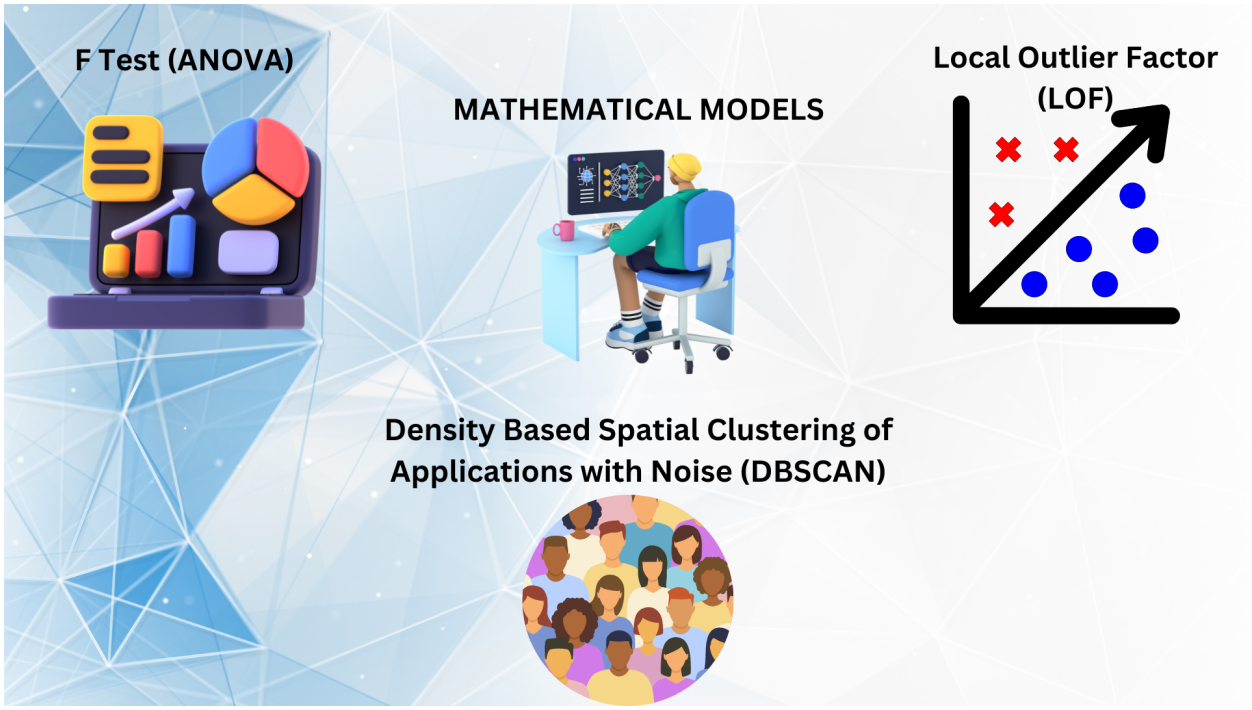
Prediction

Because it utilizes anomaly detection algorithms, hence the model also has the capability to **predict occurrence of fatal incidents beforehand based on the current trend and situation of the road.** This feature can't be achieved with image classification programs.

Multiprocessing

The Global Interpreter Lock of Python prevents the use of multiple processors which becomes a huge issue when targeting large scale applications. To overcome this issue, we used the multiprocessor modules and made the model compatible enough to **distribute the tasks among multiple processors for faster and efficient computation.**

Mathematical Models



For the algorithm to work nicely, it's important to have strong mathematical backgrounds and utilize algorithms that could do the tasks even with limited computational capabilities. For creating the best anomaly detection algorithm for our use case, we took guidance from Dr. Nidhi Asthana, who specializes in Applied Mathematics, and has over 25 research papers published by reputed international journals and conferences. For the anomaly detection, here are the mathematical concepts we applied:

F Test (ANOVA)

ANOVA (Analysis of Variance) is a collection of statistical models and their associated estimation procedures to analyze differences among means. In ANOVA, we applied a F test that compares the variances of 2 or more groups. While applying ANOVA, we assume that the variances within each group are approximately equal, the data points are independent of each other and the data within each group follows normal distribution. All these assumptions are met while taking data points from CCTV footage because the data points captured from different objects are independent, variances won't be much because of the

speed and quantity at which data gets captured, and it would approximately follow normal distribution curve.

Using **ANOVA helps us in detecting anomalies in the mean count of the crowd, vehicles, etc.** This is important because robbery generally takes place in quiet places where there's not much movement. Hence, detecting these sudden anomalies becomes crucial for our problem.

Here are the steps for implementing 1 way ANOVA

1. State the hypotheses:

   Null hypothesis (H0): Assumes that there are no significant differences between the group means. μ1 = μ2 = ... = μk

   Alternative hypothesis (Ha): Assumes that there are significant differences between at least two group means.

2. Collect data:

   Gather data from each group that you want to compare.

3. Calculate the group means and overall mean:

   Calculate the mean (Ȳ) of all data points in the entire dataset.

   Calculate the mean (Ȳ_i) for each group separately.

4. Calculate the Total Sum of Squares (SST):

   Calculate the sum of squares of deviations from the overall mean for all data points.

   $$SST = \Sigma(\Sigma(Y - \overline{Y})^2)$$

5. Calculate the Between-Group Sum of Squares (SSB):

   Calculate the sum of squares of deviations between the group means and the overall mean, weighted by the group sizes.

   $$SSB = \Sigma(n_i * (\overline{Y}_i - \overline{Y})^2)$$

6. Calculate the Within-Group Sum of Squares (SSW):

   Calculate the sum of squares of deviations within each group, summing the squared differences between individual data points and their respective group means.

   $$SSW = \Sigma\Sigma(Y_{ij} - \overline{Y}_i)^2$$

7. Calculate the degrees of freedom (df) for both the between-group and within-group variance.
8. Calculate the Mean Square (MS) for both between-group and within-group variance.

9. Calculate the F-statistic:

   Divide the Mean Square for between-group variance by the Mean Square for within-group variance.

10. Determine the critical value:

    Determine the critical value of the F-statistic from the F-distribution table based on the significance level and degrees of freedom.

11. Compare the F-statistic and critical value:

    If the calculated F-statistic is greater than the critical value, you reject the null hypothesis and conclude that there are significant differences between the groups. If the F-statistic is less than the critical value, you fail to reject the null hypothesis.

12. Perform post hoc tests (if necessary):

    If your ANOVA indicates that there are significant differences between groups, you can conduct post hoc tests to determine which specific groups are different from each other.

Local Outlier Factor (LOF)

Local Outlier Factor (LOF) is an algorithm for finding anomalous data points by measuring the local deviation of a given data point with respect to its neighbors. **It helps us in detecting outliers in coordinates, speeds and object counts in the area of supervision.** Here's the algorithm for implementing LOF:

1. Local Reachability Density (LRD):

   The Local Reachability Density of a data point, let's say data point A (denoted as LRD(A)), is the inverse of the average reachability distance between A and its k-nearest neighbors.

   Let's define k as the number of nearest neighbors used to calculate the LOF, and N_k(A) as the set of k-nearest neighbors of data point A (including A itself). The reachability distance between data points A and B is given by dist(A, B), which represents the distance between A and B.

   The formula for LRD(A) is:

   $$LRD(A) = \frac{k}{\Sigma(dist\ (A, B)\ for\ B\ in\ N\_k(A))}$$

2. Local Outlier Factor (LOF):

   The Local Outlier Factor of a data point A (denoted as LOF(A)) is a measure of how much more or less densely surrounded A is compared to its neighbors. It is computed as the average ratio of the Local Reachability Density of A to the Local Reachability Densities of its k-nearest neighbors.

   The formula for LOF(A) is:

   $$LOF(A) = \frac{\Sigma\ (LRD\ (B)\ for\ B\ in\ N\_k(A))}{LRD(A)*k}$$

The LOF(A) value close to 1 indicates that data point A has a similar density to its neighbors and is not an outlier. If LOF(A) is significantly greater than 1, it suggests that A is in a sparser region compared to its neighbors, indicating it might be an outlier.

Density Based Spatial Clustering of Applications with Noise (DBSCAN)

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a density-based clustering algorithm used for grouping data points in a dataset based on their density. **DBSCAN helped us in finding crowds in a particular region which usually forms in the case of criminal incidents.** Some parameters that affects the working of DBSCAN algorithm:

1. Epsilon (ϵ):

   Epsilon defines the neighborhood radius around a data point. All the points within this radius are the neighbors of the point at the center.

2. Minimum Points:

   Minimum points specify the least points required to call the region a cluster. This will be useful for figuring out the area cluster spans into.

Some key concepts of DBSCAN algorithm:

1. Density Based Clustering:

   The aim of the DBSCAN algorithm is to separate regions of high density from the rest. In order to do so it analyzes each and every coordinate, and computes whether there are enough points near it to call it a region of cluster.

2. Core Points:

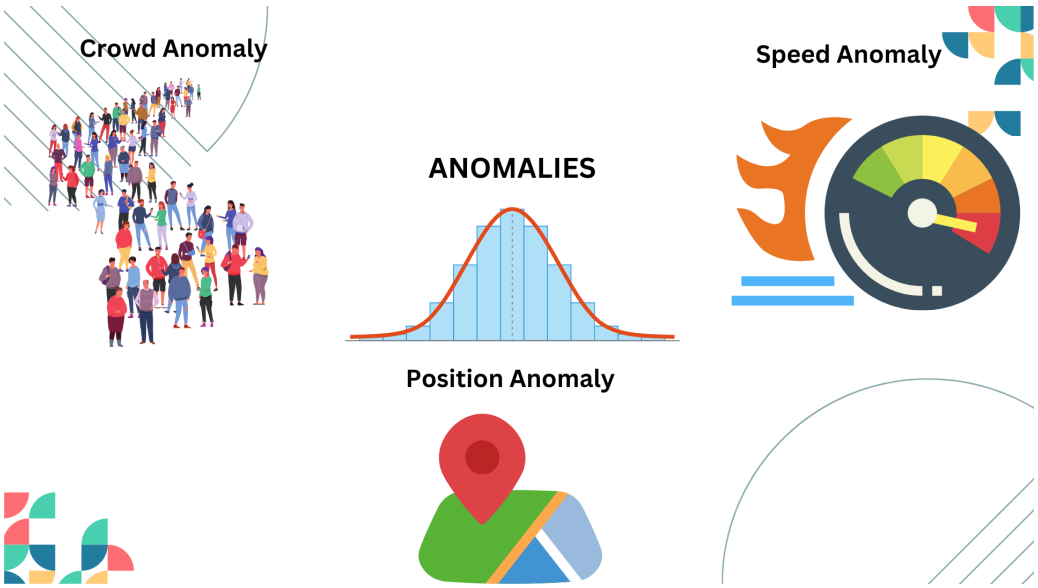
   A coordinate is given the status of core point if it has the minimum points (constant set by the programmer) in the radius of ϵ. All the core points make up the region of the cluster.

3. Border Points:

   A coordinate is given the status of border point if it is a neighbor of a core point, but does not have enough points to be called a core point. It defines the boundaries of identified clusters.

Hence in order to find the clusters, we need to analyze each and every coordinate, check if it's a core point, a boundary point or just an outlier. Based on this, finalize the clusters.

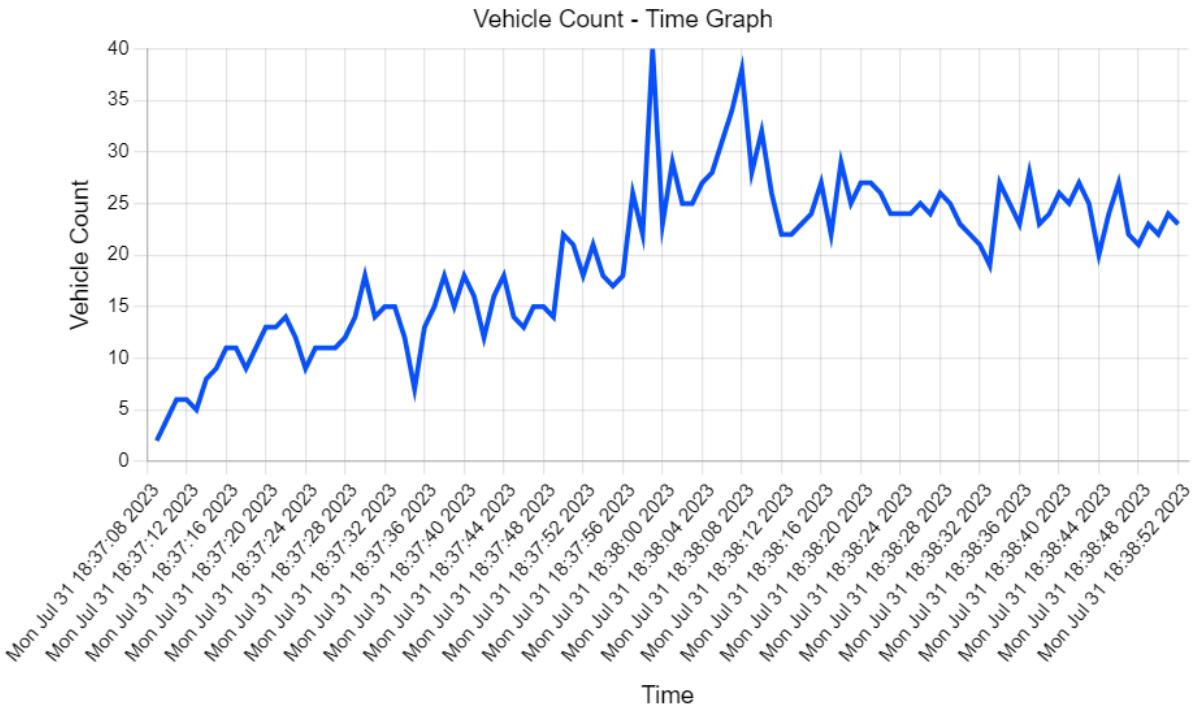
Anomalies



In order to make the anomaly detection algorithm work, one of the most important things is to make a list of possible anomalies that generally occur in case of incidents.

Crowd Anomaly

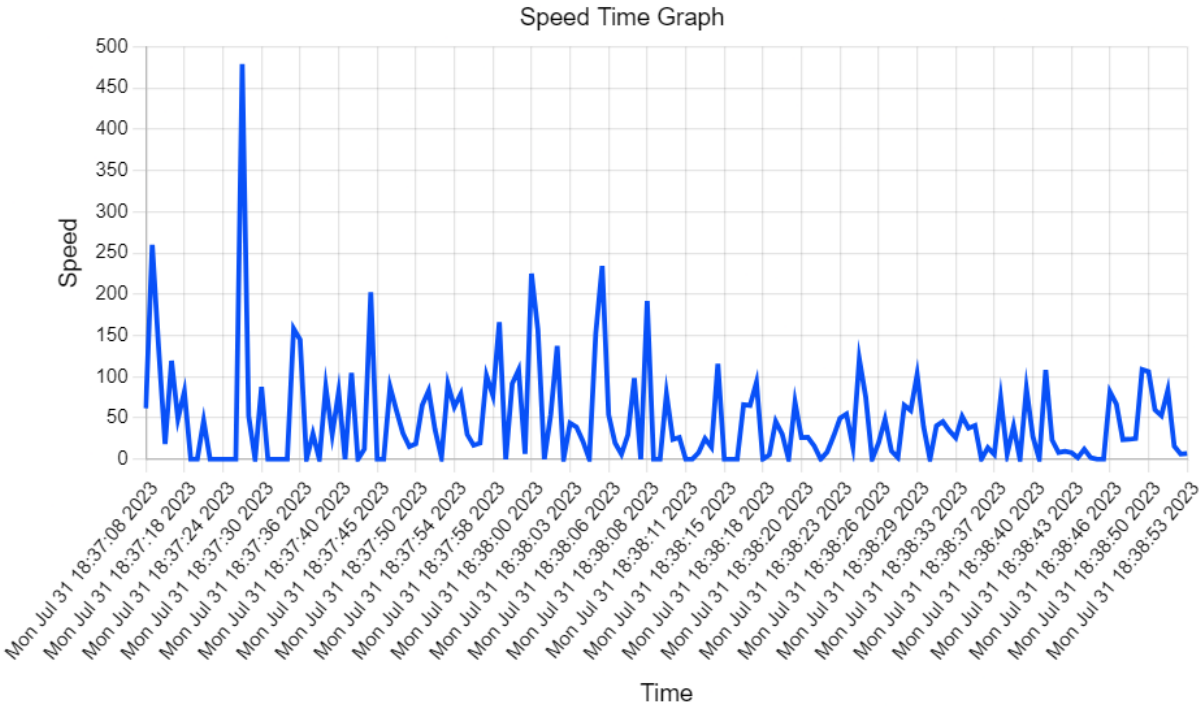
In case of any criminal activity or accidents, the count of people in that region suddenly tends to go up or come down. This anomaly can help us in detecting criminal activities in quite a short time.



Graph of the count of vehicles accumulated in an accident situation. X axis signifies time, while the Y axis denotes the count of people. Developed by our team.

Speed Anomaly

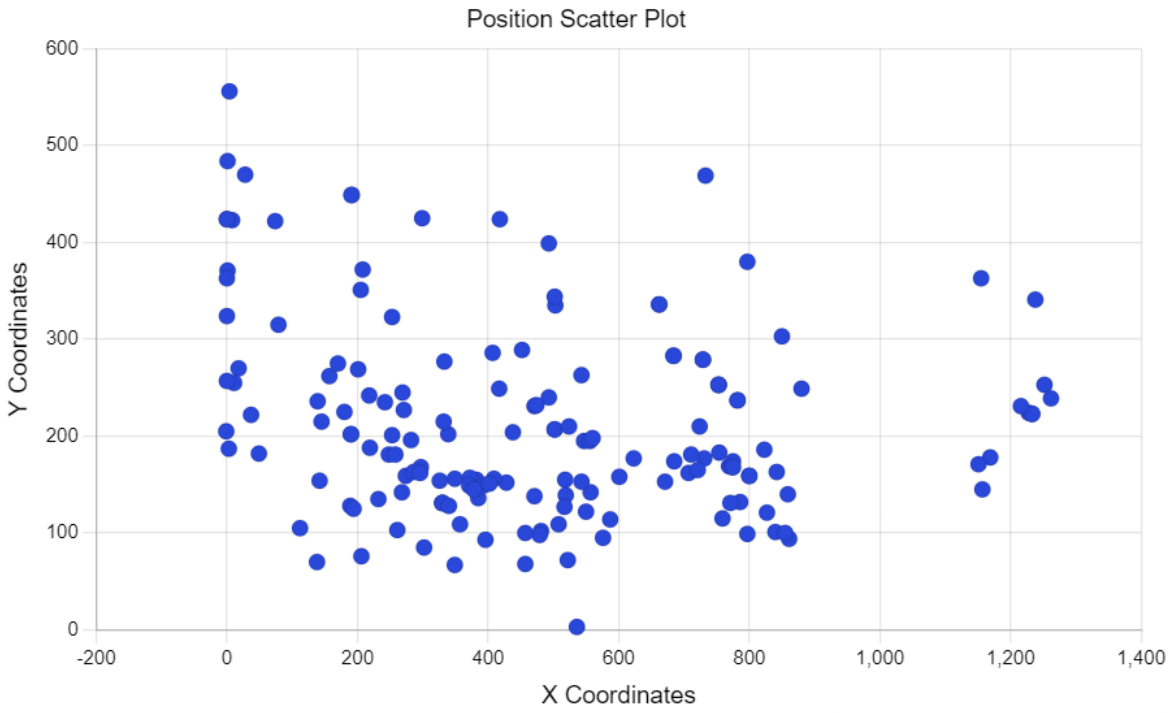
In the case of criminal activities, the walking speed of people, or the speed of vehicles undergo sudden change. It either increases or decreases swiftly. This anomaly helps us in detecting crimes, overspeeding vehicles, and even robbers trying to run away.



Graph of average speed of vehicles after an incident (Generated by the team)

Position Anomaly

During drink and drive cases, or murders, it's generally observed that their location becomes anomalous because of the inactive places they get discovered in. In this situation, keeping an eye on the position of objects helps us in detecting or even predicting possible criminal activities.



The above graph shows the spread of people in x and y coordinates during a criminal activity. Outliers in these situations become a crucial clue for our detection purpose.

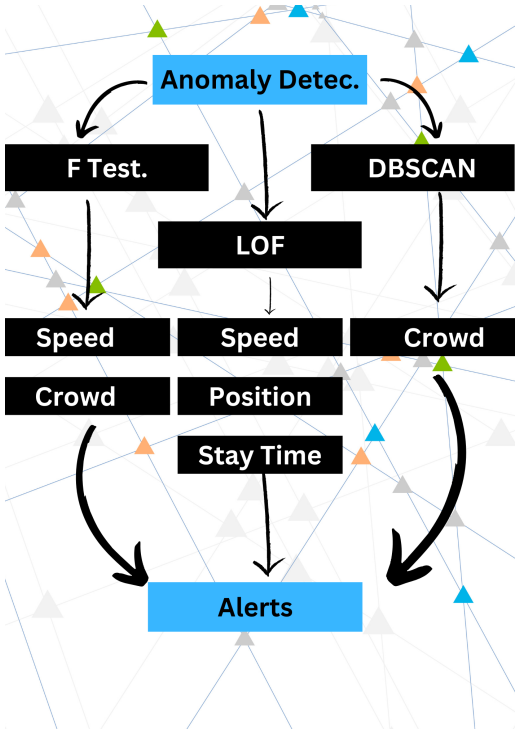
Budget

Component	Price (Per item, in ₹)	Quantity	Total Cost
Raspberry Pi	6080	1	6080
Raspberry Pi Camera Module	500	1	500
ESP 32 Camera	650	1	650
FTDI Adapter	250	1	250
Total			**7480**

Before experimenting with the Raspberry Pi System, we tried using ESP 32. For better performance, we shifted to Raspberry Pi.

Risk Assessment

Our project follows the given structure:



By the above flowchart, it can be observed that speed gets measured by both F Tests, and LOF algorithms. Similarly the crowd gets analyzed by the F Test and DBSCAN algorithm, and the rest of the parameters by LOF.

- So **there are 6 detections that are constantly working on real time data to generate alerts in case of anomalies.** In case F Tests fail to provide information on anomalies, we can expect LOF to detect those anomalies (thus we have a backup plan for all things).
- If our model fails to work at new, unexplored regions, **manually resetting hyperparameters of the anomaly detection algorithm can help it work much better.**
- Moreover, the model has the **capability to enhance itself constantly while observing the situation** of the area of surveillance because it starts learning the trends and patterns of that area.
- If the model still works unexpectedly, then we have additional options in the streamlit frontend of our application to **manually trigger alerts**. The model could correct itself according to this, and enhance itself.

Evaluation and Success Metrics

The proposed model is working on the datasets with 97% accuracy. We tested it with night vision cameras, and the results seem to be positive. In case of failure, adjusting a few hyper parameters make the model work much better for that particular area of supervision.

Sustainability and Impact

Integrating Application with Latest Technologies

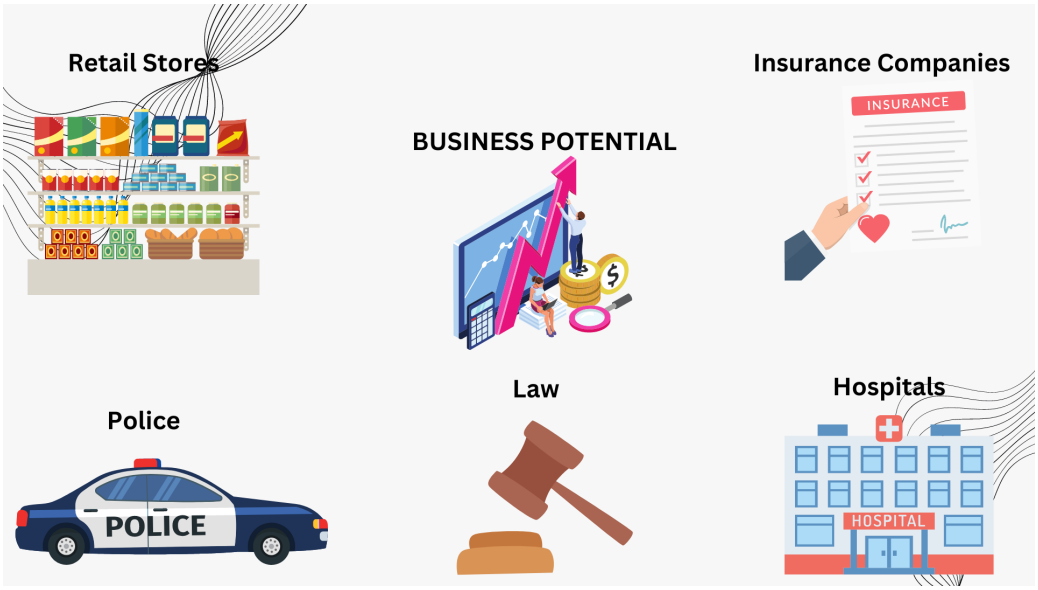
Now that we've created the application, one question you might have is whether we'll be able to integrate it with better technologies, use new programming languages, or change the database.

The answer is yes, the application is made in accordance with the principles of software development. **It follows Encapsulation, Abstraction, Composition, etc which make it easy to replace components with better technologies/ approaches.** Moreover, because we used streamlit, hence **you could even use markdown, HTML, CSS directly with Python which gives us a lot of flexibility and features.**

Adding Features in Existing Application

Adding features in the existing application is also quite simple because of the encapsulation approach taken by us. The **program also does tasks in a multi processing approach, which gives us the confidence that in the worst case of failure, it would only affect a single process**, and the rest of the processors can still continue to work normally.

Business Potential



The Government of India can also utilize this model to generate revenues. Here are the business models we propose:

Retail Store's Security

This application can be used in order to secure retail stores. With a subscription based system on our application, this can help the government with a continuous source of income, and the stores with a constant AI supervision.

Collaboration with Different Agencies

Victims of any criminal incidents/ accidents require immediate medical, legal and police help. In these cases, we can collaborate with different hospitals, legal agencies etc to give victims of incidents assistance within minutes.

Reducing Frauds in Insurance

Insurance companies can use these reports of this technology in order to detect fraudulent claims, and fasten the procedure to provide insurance to victims.

Conclusion

Some concepts used in our project are a part of cutting edge technologies developed by researchers from Stanford University. Applying concepts of diverse fields like physiology, medical science, etc into this problem statement has allowed us to tackle it better than the rest of the models. Furthermore, adjusting the hyperparameters helps the model in performing much better for a particular area of supervision. Model has the capability to work even without much training data which makes it more flexible and usable.

Some of us would like to continue working on it as our major project irrespective of the outcome of this hackathon.

Team

- **Sam Varghese**: Team Lead, Machine Learning, 3rd year MBA, BTech student
- **Sudhanshu Rastogi**: Project Manager, 3rd year MBA, BTech student
- **Chaitanaya Kusumakar**: Robotics, 3rd year, MBA, BTech student
- **Sanskriti Sharma**: Application Developer, 2nd year MBA, BTech student
- **Avni Bhardwaj**: Website Developer, 2nd year BTech student
- **Arohi Jain**: Backend Developer, 2nd year BTech student
- **Dr. Aaquil Bunglowala**: Mentor, Associate Dean of STME, NMIMS
- **Dr. Vikas Khare**: Associate Professor at NMIMS

We are fortunate to have an amazing team, with excellent professors who have excelled in their respective fields. Our team members are quite active in competitions where innovations are motivated. The **team secured AIR 6 in IIT Kanpur's Techfest**, where they proposed **advanced submarine models capable of performing tasks with 50% reduction in manufacturing costs** and **also achieved AIR 4** for making **a line following robot with obstacle detection to solve the maze in the least possible time.**

We included Dr. Aaquil Bunglowala, who's the Associate Dean of STME, NMIMS for guidance. He specializes in Electronics and Telecommunication. Due to his guidance, we were able to put up a lot of innovations in our model.

Then we have Dr. Vikas Khare who has done Data Analytics from IIT Madras and delivered lectures at IIT Jodhpur. Apart from this he's a certified energy manager at the Bureau of Energy Efficiency, India.

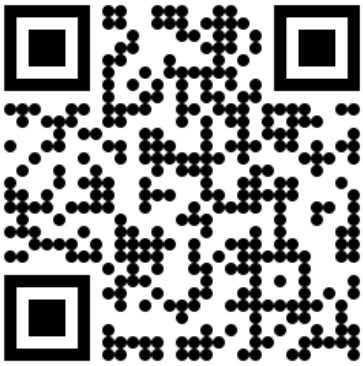
For the excellent mathematics we covered in this project proposal, we would like to thank Dr. Nidhi Asthana who has a record of publishing more than 25 renowned research papers at international journals and conferences. She was our math professor in the year 2022.

Our group photo taken at NMIMS

References

- https://online.stanford.edu/courses/cs229-machine-learning
- https://web.stanford.edu/class/cs259d/lectures/Session13.pdf
- https://www.wikipedia.com/en/Anomaly_detection
- https://www.wikipedia.com/en/Cluster_analysis

QR Codes



Website QR Code

Digital Copy of this Proposal

GitHub Repository QR Code