AI-Powered Crime Hotspot Detection in Los Angeles Using CNN-LSTM and Geospatial Analysis

MSc Research Project
Data Analytics

Jaswanth Chandrashekaraiah
Student ID: 23285265

School of Computing
National College of Ireland

Supervisor: Harshani Nagahamulla

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Jaswanth Chandrashekaraiah
Student ID:	23285265
Programme:	MSc in Data Analytics
Year:	2024-2025
Module:	MSc Research Project
Supervisor:	Harshani Nagahamulla
Submission Due Date:	15/09/2025
Project Title:	Al-Powered Crime Hotspot Detection in Los Angeles Using CNN-LSTM and Geospatial Analysis
Word Count:	2987
Page Count:	20

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Jaswanth Chandrashekaraiah
Date:	15/09/2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Al-Powered Crime Hotspot Detection in Los Angeles Using CNN-LSTM and Geospatial Analysis

Jaswanth Chandrashekaraiah
23285265

Abstract

Urban crime is a threat to public safety, and it continues to be so even with high metropolitan cities such as Los Angeles. Forecasting crime risk accurately at a more spatial and temporal level can help law enforcement, policymakers, and the public engage in smart, proactive decision-making. This research proposes a deep-learning system for crime risk prediction, which integrates historical crime trends and contextual environmental factors in its projections on future crime risk levels. The geometric representation of the city is with regular H3 hexagonal grids, with each evaluated every month with respect to its spatial and temporal attributes. A hybrid model of Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks is developed to extract features from historical sequences of crimes and from images within Google Street Views respectively. The interactive map, which provides intuitive understanding of crime distribution in the city, shows the predicted risk groups identified as low, medium, or high. The design is modular for possible future real-time realization. Ethical considerations about data sensitivity and fairness are integrated into the whole. This approach demonstrates the possibility of combining deep learning and geospatial encoding with urban imagery to advance understanding of crime risks in smart city applications.

1 Introduction

Efforts to address significant problems in police departments and city planners arising from increased frequency and unpredictability in urban crime have been particularly difficult for growing metropolitan cities, like Los Angeles. Most methods of analyzing crime use static crime statistics and pre-set area boundaries; therefore, they cannot analyze how criminal behavior changes with the effect of time and in the local environmental context. Thus, predicting the accuracy score remains poor, and proactive intervention becomes difficult.

The proposed research presents a new deep learning-based crime forecasting system that attempts to provide a solution to these problems in crime forecasting: spatial and temporal data. The main techniques of this system include employing Convolutional Neural Networks (CNNs) to capture the environmental cues through Google Street View imagery and modeling the historic trend of crimes over time using Long Short-Term Memory (LSTM) networks. The objective behind combining both approaches into a hybrid CNN-LSTM model is to predict monthly crime risk levels (low, medium, or high) at a fine-tuned geographic level through H3 hexagonal spatial encoding. The product of a stakeholder's risk projections will be further portrayed on an interactive map, which will be open to public, police, and policymaker exploration. Beyond that, this system is designed with a further integration roadmap into Google Maps-like platforms to provide risk warnings in real-time when a user searches for or travels to specific places in the city mapping for a specific location. Such initiatives will bring proactive policing, safer, or at least more performant mobility within cities, as well as familiarize city dwellers and tourists with their surroundings.

1.1 Research Background and Motivation

Crime prediction has been a continual evolving area of research. Traditionally, these methods included hotspot mapping, statistical clustering, and ARIMA time-series forecasting to anticipate where and when crimes are likely to occur. However, these techniques often ignored two important components: (1) the environmental appearance of neighborhoods which may have influence on criminal activity, and (2) the dynamic interplay among location-specific crime trends over time.

Recent advances in deep learning have shown the promising capability of CNNs in visual feature extractions and LSTMs in sequential prediction tasks. Meanwhile, with the advent of geospatial indexing systems such as H3, flexible resolution-dependent representation of urban landscapes has deemed possible. Using a combined approach may allow us to set the stage for a more context-driven and fine-grained framework in crime forecasting.

Apart from having a rich source of visual context, Google Street View images are available for free in most urban environments. These images, when fed to CNN models, can capture subtle indicators like lighting, building types, urban decay, and surveillance presence, all of which may correlate to crime risk. Thus, merging these visual data with monthly historical crime records through an LSTM framework opens the way for spatially fine crime risk predictions.

1.2 Research Question

To what extent can a hybrid CNN-LSTM model improve the accuracy and spatial granularity of crime risk forecasting in Los Angeles using historical crime data and Google Street View imagery?

1.3 Contribution

This study provides some key advancements toward the area of crime risk forecasting. It proposes a modern hybrid CNN-LSTM model that correlates spatial features from Google Street View images with temporal patterns from historic crime data for more precise and context-based allocations across city landscapes. H3 hexagonal geospatial encoding at resolution 8 provides a higher resolution for spatial segmentation of the city for its localized forecast. A synchronized dataset is derived by combining the monthly LAPD crime statistics and the corresponding street-level imagery for each H3 zone. The model's performance is assessed in detail against standalone CNN and LSTM models based on a wide range of evaluation metrics: Accuracy, Precision, Recall, F1-Score, Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), R-squared, and a Confusion Matrix. Finally, the project features an interactive Folium-based risk map that color codes crime risk (low, medium, high) across the city of Los Angeles, improving interpretability for end users. Additionally, a lightweight Flask-based prediction API has been implemented, leaving room for future integration with real-time applications like Google Maps. Most importantly, it records ethical issues posed by predictive policing, including data bias, fairness, and privacy concerns within this study.

1.4 Scope and Limitations

Using LAPD crime data alone and Google Street View images cover the period 2020 to the present to assess crime risks within the city boundaries of Los Angeles. The forecasting exercise is attempted as a categorical classification task, which means that every H3 hexagonal cell gets assigned a monthly low, medium, or high risk label. However, the model does not predict what types of crime or how many may occur. This research admits various limitations. First, spatial feature quality may be compromised by the temporal inconsistency of availability from Google Street view images. Second, underreporting in the LAPD crime dataset may be subject to geographical biases that impact fairness or accuracy in predictions in a given neighborhood. Third, given that this model is trained over Los Angeles city specifically, it may not generalize well to other cities without significant retraining or adaptation. Subsequently, the system is fed with historical static data, without provision for real-time or streaming data input, at this time curbing any further applicability to situations of in vivo crime monitoring.

1.5 Report Structure

This report is organized in order to give a comprehensive account of the research process and the findings, scientifically. Section 2 includes a literature review of the prior works in crime forecasting and spatial data modeling, emphasizing the contributions in deep learning applications within urban analytics. Section 3 entails methodological work in terms of data collection and preprocessing followed by Google Street View image acquisition, introduction of H3 geospatial encoding, and the design of the hybrid CNN-LSTM model. Section 4 consists of system design specifications specified in terms of data pipelines, model components, and visualization tools. Section 5 describes the deployment process, providing the framework for development environment, model training, and hyperparameter tuning. Section 6 evaluates the models through various evaluation metrics and comparative result analyses with their CNN and LSTM standalone versions. The last part, Section 7, exhibits interactive crime-risk maps based on Folium and other output provisions from the user interface. In short, Section 8 concludes the report summarizing insights derived, delimiting boundaries, and recommending directions of future improvements and real-world applications.

2 Related Work

Understanding and anticipating urban crimes, one must integrate spatial and temporal contextual factors. Recent literature shows that in this area, advanced deep learning approaches have overtaken classical time-series forecasting, including the development of hybrid models that fuse spatial imagery with sequences of data. This review synthesizes contemporary research around five focal themes: (1) the use of deep learning for spatio-temporal crime forecasting; (2) hybrid CNN-LSTMs; (3) geo-encoding and clustering; (4)

crime risk visualization; and (5) model evaluation and generalization. The review will then highlight critical research gaps that have motivated the present study.

2.1 Deep Learning Approaches for Spatio-Temporal Crime Prediction

With respect to crime data in Atlanta, LSTM aided (1) in obtaining correlation coefficients (R) of 0.874 when obtained with the use of a 50-day time-series input data at a resolution of 0.05° Latitude and Longitude Grid Cells. This highlighted the performance sensitivity of the LSTM models to spatial and temporal parameters in Atlanta. Hence, space granularity plays a great role for the observations obtained.

(2) have established a crime prediction model for Chicago, with LSTM and Convolution layers at play. Yet, to enhance the performance over the standard LSTM and CNN models, the POI (Point of Interest) data was also used for nurturing spatial representation. Similarly, work done by (3) involved the employment of ConvLSTM and graph models that gave a high hit rate of 45.86 percent, with a PAI score of 2.24, thus surpassing any standalone CNN or LSTM models. These studies thus demonstrate how effective it can be to link LSTM with any form of spatial structure.

2.2 CNN-LSTM Hybrid Models and Spatial Feature Integration

Fusion of image-based CNN features with temporal LSTM representations new orientation in urban analytics. In the Smart Police project initiated by (3) on a hybrid CNN-LSTM model to proactively recognize testers for crime pattern recognition, the system they authored integrated crime statistics with a neighborhood's visual context, parsed through CNNs, and temporal dependencies modeled throughircuit LSTMs. Though the system attained a 91 percent accuracy rate, overhead computations were stated to be hindrances to real-time deployment.

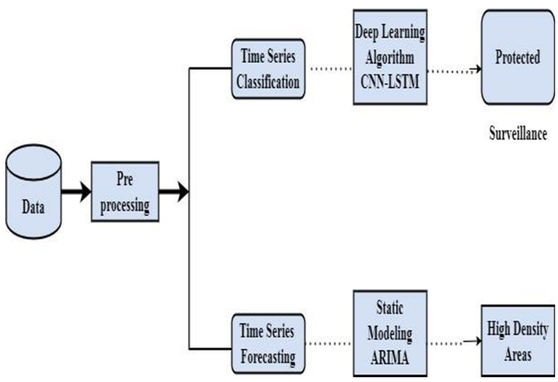

Figure 1: CNN–LSTM architecture for crime prediction.

Another significant contribution was made by (2), who performed a comparative study of various models eqs - CNN, LSTM, ST-ResNet, and Graph LSTM - on data from multivariate crime datasets. It was concluded that hybrid deep learning architectures could be relied upon to outperform single-stream networks, especially with spatial features as well as temporal sequences fused before classification awareness intervention.

2.3 Geospatial Encoding and H3-Based Clustering

Crime forecasting is highly dependent on spatial representation. Most of the models rely on grid segmentation or clustering to make patterns of regions. According to (4) in their work on modeling theft patterns, the researchers used H3 spatial encoding and found that grids adjusted for resolution could better delineate hotspots. Similarly, (5) showed that added together, spatial autocorrelation and adjacency effects significantly enhanced LSTM's ability to predict crimes within nearby areas.

The use of H3 hexagons has its own advantages in that way which has a completely uniform cell structure without distorting the shapes existing on square grids. This spatiotemporal approach is most compatible with urban forms and can model geospatially at high resolutions; hence the relevance of its application in this project.

2.4 Visualization and Crime Risk Mapping

Many other crime mapping and safe-route studies have emerged. (6), for instance, developed a fuzzy-logic-based crime category mapper that visualizes risk zones and prescribes safe paths based on spatial aggregation. (1) similarly presented a 3D grid map of crimes in Atlanta, utilizing Kepler.gl to visualize the temporal patterns in spatial crime behaviour.

Such efforts underline the need for user-friendly mapping interfaces capable of adequately communicating dynamic crime risk to both police and the general public. In terms of other features seen in the recently published works and that are today adopted in the current project, Folium was selected for interactive HTML export along with red-to-orange-to-green schemes.

2.5 Evaluation Metrics and Generalizability of Models

Commonly referred to as the performance metrics, Accuracy, F1 Score, Precision, Recall, MAE, RMSE, and other measures of predictive accuracy index (PAI) are the most widely adopted in all studies. Generally, hybrid models perform well on different datasets; nevertheless, they face difficulties in generalization. For instance, a model trained with density datasets from Chicago or San Francisco tends to perform poorly when applied to a sparse region (3).

To this end, (7) have proposed an ensemble SARIMA-LSTM-RBF model that learns adapting weights dynamically with respect to neighborhood characteristics that is, this approach improves results in low-and high-crime areas and thus confirms the multi-model learning principle important when dealing with heterogeneous urban settings.

2.6 Research Gaps

There is significant interest in spatio-temporal crime forecasting, but only a handful of studies have tried to merge image-based spatial features, geospatial clustering (e.g., H3), and time modeling via LSTMs in one pipeline. Most models are either sequential or spatial, and not both. There is also a shortage of deployment-ready systems capable of inference in real time or dashboards for visualization.

This study fills this gap through:

- Combining CNN spatial features based on Google Street View with LSTM temporal models.
- Fine spatial segmentation using H3 hexagons.
- Categorical risk level prediction (low, medium, high) per area.
- Visualizing outcomes on interactive crime risk maps.
- Developing a deployable Flask-based API for future possible real-time integration (for example Google Maps alerts).

Table 1: Summary of Related Work

Paper Title	**Algorithm Used**	**Evaluation Metrics**	**Results**
Spatio-Temporal Modelling of Repeat Crimes	RNN, LSTM, Decision Tree, SVM, Random Forest, XGBoost	Accuracy, Precision, Recall, F1 Score	RNN achieved 96.5% accuracy, outperforming others
Spatiotemporal Analysis and Prediction of Crime in Atlanta	LSTM	Correlation Coefficient (R)	Best R = 0.874 with optimal parameters
Spatio-Temporal Prediction of Crime	LSTM, CNN, ConvLSTM, Graph LSTM, ST-ResNet	Hit Rate, PAI	ConvLSTM achieved 45.86% hit rate, PAI = 2.24
A Novel Multi-Module Crime Prediction Model	Attention LSTM, Fusion Model	RMSE, MAE	Lower RMSE than baseline LSTM and ARIMA
Crime Analysis and Prediction	Random Forest, Naive Bayes, KNN, SVM	Accuracy	Random Forest highest accuracy
Crime Mapping and Predictive Analysis	Naive Bayes, Decision Tree, KNN	Accuracy	Naive Bayes achieved 88.3% accuracy
Crime Prediction: Review	LSTM, Random Forest, CNN, SVM, ARIMA	Accuracy, RMSE	LSTM and hybrid models most effective
Crime Type and Occurrence Prediction	Logistic Regression, SVM, Random Forest, XGBoost	Accuracy, Confusion Matrix	XGBoost and Random Forest best performance

3 Methodology

The structure of the crime risk forecasting system was built on the CRISP-DM framework, ensuring a systematic and reproducible workflow. Business understanding consisted of predicting crime risk levels based on spatial imagery and temporal trends. The understanding of data included quality-checking and relevance-checking of historical LAPD crime data and Google Street View images. H3 hexagon coding, monthly crime-sequence aggregation, and image preprocessing were parts of data preparations. Modeling comprised training, tuning, and evaluating a hybrid CNN-LSTM architecture across several metrics. Comparison of predictions to true labels and visualization with confusion matrices formed the basis for evaluation. Deployment resulted in an interactive risk map that incorporates these embedded predictions for public or law enforcement purposes.

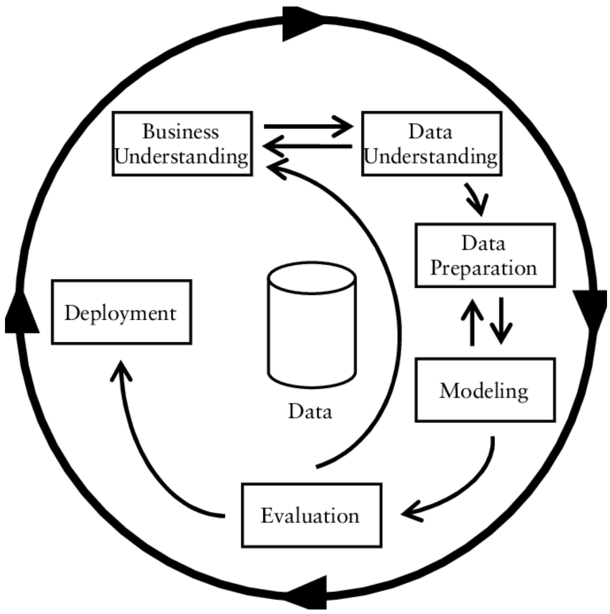

Figure 2: CRISP-DM Process

3.1 Research Framework

The CRISP-DM framework gives a structured view to the research problem-solving process and caters to all phases of the development process, from conceptualization to deployment. The identification of high-crime risk zones in Los Angeles was the core problem definition, with environmental and temporal variables chosen as the basis for risk prediction. For this operationalization, the city was segmented spatially into uniform H3 hexagons that facilitate local analysis and prediction. The innovation basically lies in the so-called dual-stream architecture: the environmental context was obtained through Google Street View imagery, interpreted by Convolutional Neural Networks (CNN), while historical crime patterns were built using Long Short-Term Memory networks (LSTMs) which are well-known for their capabilities in sequence modeling. These two sets of information were merged as a robust representation for each zone. The resulting output was a probabilistic classification of crime risk levels, demarcated as Low, Medium, or High. To help turn the results into action, Folium was used to create an interactive crime mapping tool providing visual insights on the spatial risk pattern across the urban landscape.

3.2 Data Collection

The main dataset used in this research was obtained from the Los Angeles Open Data Portal, specifically on the dataset "Crime Data from 2020 to Present" managed by the Los Angeles Police Department (LAPD). This dataset includes information on reported crimes, which are publicly accessible and updated regularly across the city, with important attributes like date and time of the incident, nature of the crime, latitude and longitude coordinates, victim demographics, and reporting district. Each record corresponds to a unique police report and is geotagged for spatial analysis at the neighborhood or micro level. For the purposes of this project, all records available from January 2020 to early 2024 were extracted, cleaned, and underwent a test of relevance and consistency. To provide for spatial context, the geocoordinates of each crime report were converted to H3 hexagons at resolution 8 for uniform spatial partitioning. In parallel, Google Street View imagery was obtained for the centroid of each hexagon using the Google Maps API to enable the model to learn visual features of the physical environment associated with reported crime zones.

3.3 Data Preprocessing

Data preprocessing was to augment raw crime records for spatio-temporal deep learning. It means that the temporal aggregation connects all reports of individual crime to a specific month and to spatial units (H3 hexagons) so that what can be seen is the conversion of event-level records into monthly counts per hexagon. This enables trend and seasonality modeling across cities. When Uber's H3 system defines geospatial encoding at resolution 8, it describes the total city space in equally shaped hexagons, with each crime location mapped to an H3 index for consistent surface granularity and local correlation analysis. For the monthly figures, Low-Medium-High categorizations were created by thresholding based on quantiles to balance the class distribution for the classification. The spatial aspect included Google Street View images taken from the centroid of each hexagon, resized to 64×64 pixels for computational reasons, and normalized for CNN ingestion. These RGB images presented additional background context for each location. Finally, the six-month crime sequences are paired with images, feeding temporal patterns and spatial features into the hybrid CNN-LSTM model to enhance crime risk prediction.

3.4 Model Architecture

The prediction model architecture is hybrid by using Convolutional Neural Networks and Long Short-Term Memory networks, as well as a sequence-to-sequence design to capture and process spatial-temporal features. Indeed, CNN is designed to ingest spatial properties derived from the 64×64 RGB Google Street View images, with the first two specifications featuring convolutional connections with an initial count of filters 32 and later tapping the rest of another 64. Each convolution uses a standard 3×3 kernel and ReLU activation for nonlinearities, followed by 2×2 max pooling to reduce dimensionality while retaining the most important features. The outcome then flattens to attain connectivity fully and forms a 128-dimensional spatial feature vector at dropout rates of

0.25 for regularizations. At the same time, for each H3 hexagon, the past six months' crime data are fed through a single LSTM layer of LSTM consisting of 64 units and then a dense layer that produces the 64-dimensional temporal vector with 0.2 dropout to avoid overfitting. The outputs of both CNN and LSTM are concatenated and directed to a dense layer of 64 units, where it is finally classified into Low, Medium and High risk through softmax activation. The training adopts the Adam optimizer (learning rate 0.001) with categorical cross-entropy loss, meant for quick convergence.

3.5 Training and Evaluation

The model training and evaluation procedure was designed so as to ensure optimal tuning of generalization and performance of the hybrid architecture of CNN and LSTM. The split of the data into training and testing is temporal at 80 percent for training and 20 percent for testing without shuffling the order of events. Train for 50 epochs using an early stop strategy when performance on validation data is plateauing as a prevention mechanism for overfitting the architecture. Batch size of 32 was settled as a compromise between the computation time and the learning stability. A grid search hyperparameter tuning was utilized to test various configurations concerning CNN filters, LSTM units, and learning rates to achieve the desired environmental configuration while the final decision was based on validation accuracy and generalization criteria. Model evaluation then consisted of classification and regression metrics. The model performance directly ascertained in predicting categories of crime risk was measured in terms of accuracy, precision, recall, and F1 score, while regression metrics MSE, RMSE, MAE, and R^2 Score give information on the magnitude of error in prediction. The confusion matrix explains how correctly or incorrectly the model predicted votes in the three risk categories. Hence, it provides quantitative and visual proof for the evaluation.

3.6 Temporal and Spatial Trends

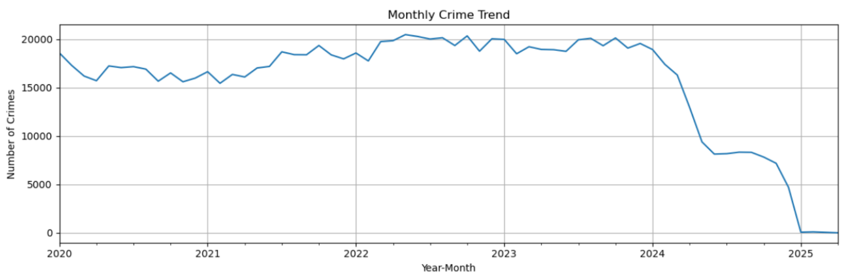

Figure 3: Monthly Crime Trend.

Between 2020 and early 2025, the chart shows monthly crime in Los Angeles. Crime hovered steadily between incidental levels of about 15,000 and 20,000 from the start of 2020 until around June 2023. After this, it decreases sharply and falls near zero in early 2025. It is possible that this sudden dip indicates differences in reporting time, incompleteness of data from the recent past, or quite probable outside influences such as changes in the policy, safe programs, or economic conditions. For instance, one showing the importance of checking temporal data quality when building forecasting models.

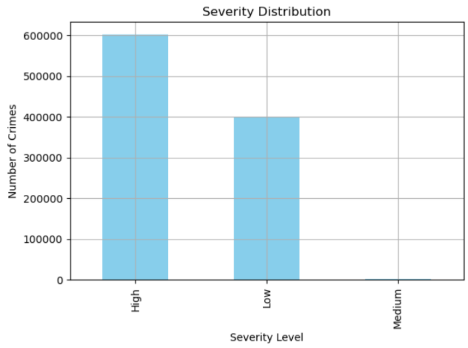

Figure 4: Severity Distribution.

This is a bar graph that compares crime records based on the risk category of Low, Medium, and High. Most of the records fall into either High or Low risk, with Medium risk standing very few records. This might be the reason for poorer classification performance of the Medium class across all the models. This indicates that the quantile threshold method has not completely balanced the labels and may need refinement or oversampling.

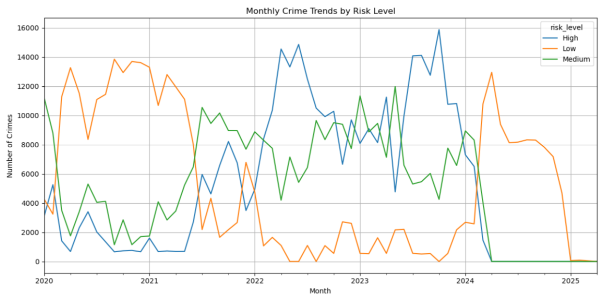

Figure 5: Monthly Crime Trends by Risk level.

The plot encapsulates monthly counts of crimes based on predicted risk levels over the same timeline. With their relatively smooth trends through time, the Low-risk zones indicate some stability in their classification, while the High-risk areas fluctuate more due to potential volatility in a changed criminal activity. The Medium zones remain confused and less distinguishable, which might contribute to their low predictive performance and increased misclassification with neighboring classes.

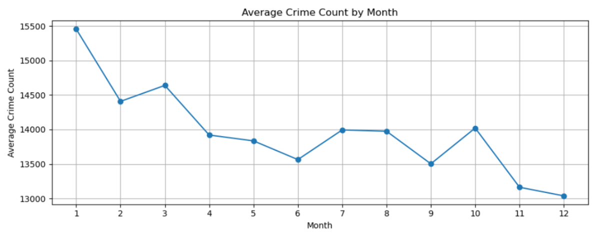

Figure 6: Average Crime Count by Month.

The line plot shows the seasonal pattern of crimes, with regard to January being the month that has the highest average crime counts, which could be caused by holiday effects, and November to December being the lowest months about crime counts in the year, perhaps indicating the reduced activity or heightened police presence during the holidays. The month-wise temporal features have been induced in the LSTM model, thereby allowing it to understand the seasonality and the recurring trends present in the data.

3.7 Visualization and Deployment

To ensure the model is made interpretable and invokes interest from the users, the results were displayed using the Folium library interface in Python. The Los Angeles map depicted the H3 hexagonal cells representing microgeographic areas, with each hex colored according to low risk-green, medium-orange, and high-red, depending on the model's prediction as a visual means of communicating crime risk levels. This color system was sufficiently intuitive for quick identification of the relative safety of areas. Tooltips showed on-hover for further investigation by the user into the location or predicted class, while map controls offered interactive zooms and pans allowing users to explore the city. Once done, it would be exported into an HTML file that is friendly for sharing into external applications or embedding in them. This visualization would serve not just as a means to validate our model but also put us into deployment through Flask as part of a just-in-time DSS intended for both law enforcement and public users.

3.8 Ethical Considerations

Ethical considerations were the most crucial element in this research, given that crime data are highly sensitive. Crime datasets have historically been tainted by systemic bias for some neighborhoods, over-policed, or under-surveilled. If those types of biases are not decoupled from a predictive model, it would expose both those areas as high-risk zones even when there is a change in crime rates. To minimize this risk, the study used only de-identified and aggregated crime counts, avoiding any personal or demographic identifiers that could have compromised privacy or unnecessarily amplified profiling. Further, visual inspection tools were combined into the mapping interface for assistance in identifying and purviewing areas with generally high predicted risk levels that could be indicative model bias rather than true crime patterns. Those were the elements under those frameworks as prescribed by the ethical approaches such as the ACM Code of Ethics and the AI Fairness Principles consultation for just the whole project in transparency, accountability, and responsible use of predictive analytics in city crime forecasting.

4 Design Specification

The design of the crime risk forecast system is meant to allow a deep learning framework to easily ingest spatial imagery, crime data over time, and geospatial mapping. The overall architecture consists of four main modules: data ingestion and preprocessing, Hybrid CNN-LSTM modeling, prediction and inference, and visualization and deployment. Due to the nature of the design, the overall architecture supports modularity, scalability, and applicability in real-world settings.

4.1 System Architecture Overview

Modular pipeline architecture allows for the visual end of data processing in systems. Data Input Layer is the initial line. It accepts geocoded crime statistics from LAPD together with environmental images through Google Street View, then proceeds into Preprocessing Module. The raw temporal aggregations are now geospatially encoding H3 hexagons, labeling risk categories and matching them with the preprocessed images. Next, the data go into a CNN-LSTM model. Branches of CNN and LSTM are created whereby CNN extracts spatial features of the image and LSTM identifies the past six months of temporal crime trend. Based on that, Fusion and Classification Layer will bring those features together and predict the crime risk level-low, medium, or high-for each geographic unit. These predictions will then end up in the Output Layer and will be connected with a Folium-based Visualization Interface, allowing users to create their own real-time interactive explorations of crime risk predictions across Los Angeles, all with a very easy map interface.

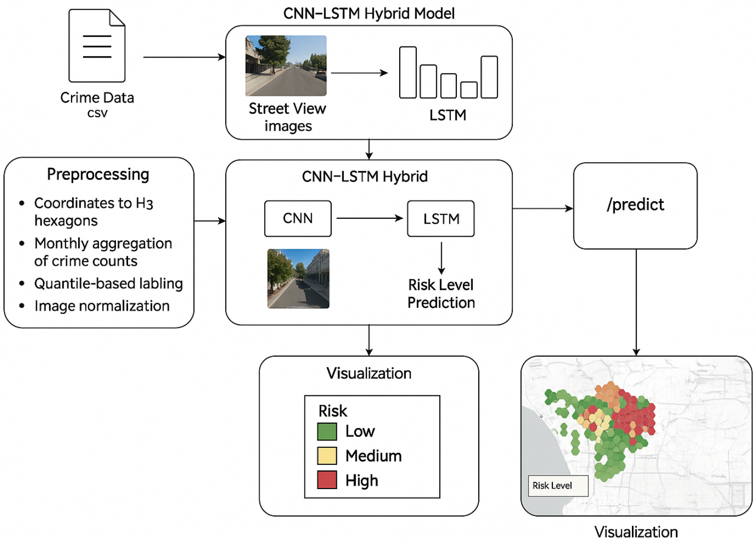

Figure 7: Overall Architecture.

4.2 Component Breakdown

There are five interlinked components through which the data wave travels from the raw input to the interactive visualization of crime risk. The system receives input with the geocoded LAPD crime data stored in CSV format and with the Google Street View images fetched through the Google Maps API, stored as 64×64 pixel .jpg files for each H3 hexagon centroid. In preprocessing, the crime locations are encoded in H3 hexagons at resolution 8 and summed up by month and assigned risk levels of Low, Medium, or High through quantile thresholds. The images are normalized for CNN input, with each region represented by a time series vector and an image. The CNN outputs a 128-dimensional spatial vector, whereas six months of historical crime data are processed by the LSTM into a 64-dimensional temporal vector. These two are concatenated, passed to a dense layer, and classified via softmax. The Flask API receives prediction requests and returns the JSON response, which includes risk level, confidence, and likely crime type. The results are plated onto an interactive map color-coded for the crime risk, with hoverable tooltips, using Folium.

5 Implementation

This chapter describes the technical specification for the crime forecast risk system, describing the development environment, the key programming frameworks, model architecture execution, and deployment approaches. The entire system was implemented in Python via modular notebooks and scripts for preprocessing, model training, prediction serving, and interactive visualization.

5.1 Development Environment

The forecasting of crime risks was developed using a mainly Python-based environment, formulated primarily on the possibility of a more enhanced ecosystem of libraries on data science and deep learning. The data preprocessing and manipulation were done with the aid of Pandas and NumPy, whereas the geo-spatial activity based on hexagonal grids encoding latitude-longitude coordinates is accomplished through GeoPandas and H3. OpenCV and Pillow (PIL) are integrated to resize and convert images for Street View images that are to be standardized at 64×64 resolution, while preparing them for training. The deep learning components integrity was implemented via CNN-LSTM hybrid modelling in PyTorch, allowing for dynamic and active scalability in defining personal architectures for GPUs. Scikit-learn integrated evaluation metrics and classification utilities; Google Maps API was used to programmatically fetch Street View images. A lightweight Flask application is built to be used for deployment, serving up-to-the-minute predictions and presenting the risk map. The final visualization interface developed on Folium enables the creation of interactive as well as color-coded crime risk maps.

5.2 Data Integration Pipeline

The implementation began with the modular structure for the data pipeline, an ingestion and processing mechanism for the two different streams: crime record and Street View images, started being constructed. The crime data (CSV) was read for a time range and a location, filtered for that location, and, thus, converted for each crime record to an H3 hex utilizing Uber's h3 library. These monthly data would then be aggregated, as they say, via groupby() into a time series by hex. In parallel, Google Maps API was employed to download the images using the passing H3 centroid coordinates which were saved in the structured folder system for efficient indexing.

Images were resized to 64×64, normalized, and converted to tensors through the PyTorch transforms. A final dataset was composed in which each row represented a region with a 6-month vector of crime counts and a corresponding RGB image tensor.

5.3 CNN-LSTM Model Construction

The hybrid deep learning model was developed in PyTorch using a custom neural network class; the structure has two parallel branches for processing data: spatial or temporal. The CNN branch was mainly for image input, fulfilling the perception task with several nn.Conv2d convolution layers and an nn.ReLU activation, followed by a nn.MaxPool2d layer for dimensionality reduction. The CNN branch output was fed into an nn.Linear layer and created a 128-dimensional spatial feature vector. On the contrary, an nn.LSTM layer with 64 hidden units was fed with six months worth of time series crime data and then processed to a dense layer, producing a 64-dimensional temporal feature vector. The output from these two streams was concatenated into one feature vector and was passed through a dense layer and softmax layer for outputting three probabilities corresponding to low, medium, and high risk of crime. Dropout layers were used in both branches to avoid overfitting. The Adam optimizer (learning rate 0.001) with categorical cross-entropy loss was used for model training to ensure stabilization and efficient optimization across the training process.

5.4 Model Training and Tuning

The model used was of an 80:20 split where training and test data were stored, and a grid search was conducted designated CNN filters, LSTM units, and learning rates in combinations of CNN filters (32, 64), LSTM units (32, 64, 128), and learning rates (0.001, 0.0005). The total epochs subjected to training at first were 50 epochs but ended up ending early using early stopping on validation loss. The training logs, accuracy curves, and loss plots were saved and analyzed as data for performance trends.

5.5 Flask Deployment and API Integration

Crime risk analysis prediction application has been designed at backend as a Flask web application to offer real time access and also to give deployment light. Front end reveals two broad API endpoints. First, /predict takes geographical coordinates as input through query parameters.

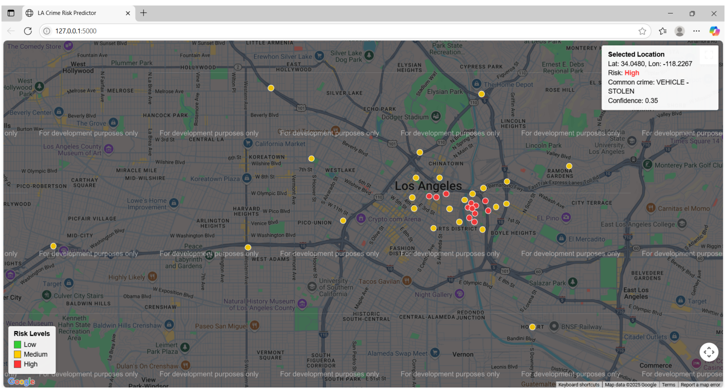

Figure 8: Google Map Integration.

It maps the coordinates to the appropriate H3 hexagon, obtains the associated Street View image and time series data for crime occurrence, makes an inference based on the trained and existing model using CNN-LSTM, and then returns a structured JSON response output. This response, namely in JSON, will include the predicted crime risk level ranging from Low to High together with associated confidence score and most commonly previously observed crime type in that area. The second endpoint, /map, provides an interactive HTML map using Folium in which each H3 cell is given a color dynamically according to the predicted risk class by the model. This arrangement allows convenient incorporation with web dashboards and allows future possibilities towards embedding model output into mobile apps as well as navigation tools for realtime crime consciousness.

5.6 Risk Map Generation

Final forecast of crime victimization were visualized through Folium, that interactive webpage map across Los Angeles. Each H3 hexagon on the map had a color according to the softmax output of the model, which is green for low risk, orange for medium risk and red for high-risk zones. Hence, from the very simple color assignment it will be to see how the crime levels disperse throughout the city spatially.

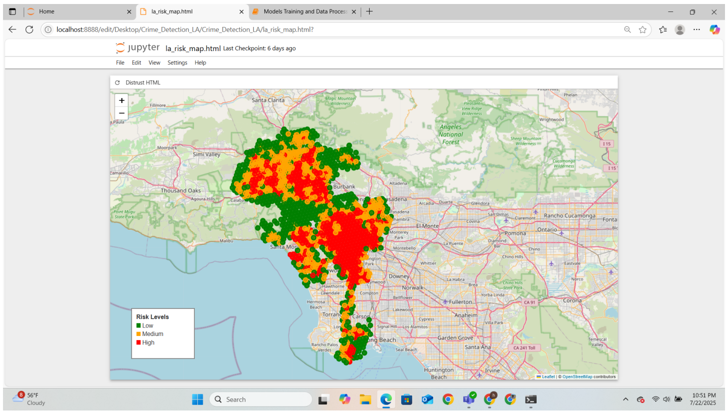

Figure 9: Risk Map Indicator Visualization.

In addition, each hexagon also had hoverable tooltips to provide critical information such as the predicted risk value and the optional historical crime details. This map is exported into an independent HTML file to facilitate further communication of the raw views to stakeholders or integration into web platforms. The Folium user interface even allowed for real-time interaction by users (zooming, panning, or toggling map layers), enhancing accessibility for analysts and public-facing applications.

6 Evaluation

In this chapter, we carry out systematic performance analysis of the system using different deep learning architectures. The three models tested and compared during the course of this evaluation include CNN (spatial-only), LSTM (temporal-only), and the proposed hybrid CNN-LSTM model. The evaluation was carried out on a withheld test comprising 20 percent of the full dataset. The classification performance was quantified using accuracy, precision, recall, F1-score, and confusion matrices. Additional visualizations were made to interpret the behavior of the data distribution and the model.

6.1 Quantitative Results

The table below summarizes the performance of each model across key classification metrics:

Model	Accuracy	Precision (Macro)	Recall (Macro)	F1 Score (Macro)
CNN	0.326	0.3237	0.326	0.3237
LSTM	0.7952	0.7957	0.7952	0.7952
CNN–LSTM Hybrid	0.8048	0.7976	0.7989	0.7977

Figure 10: Quantitative Findings.

The quantitative findings encapsulated in table depict a comparison of the three models, CNN, LSTM, and CNN-LSTM Hybrid, based on the quantifiable classification metrics. The hybrid model, CNN-LSTM, led the progressive scales with an accuracy of 80.48 percent, a precision of 79.76 percent, a recall of 79.89 percent, and F1 scores of 79.77 perccent. These represent scores of a balanced model that can predict effectively through all three categories of crime risks. With regard for those hype-specific factors alone, the LSTM model, concentrating on temporal trends, came hotter at an accuracy of 79.52 percent with just slightly lower macro scores across other metrics, thus confirming that time-based patterns alone have very strong forecasting capabilities. By contrast, a purely image-based spatial context model falls very low at the figure of 32.6 percent in accuracy, indicating insufficient features from the Google Street View image for the accurate classification of crime risk levels. All above denotes an unscathed form of hybridization that could, through virtue of both spatial and temporal data, thus surfeit any predictions found lacking reliability and robustness.

6.2 Confusion Matrix Analysis

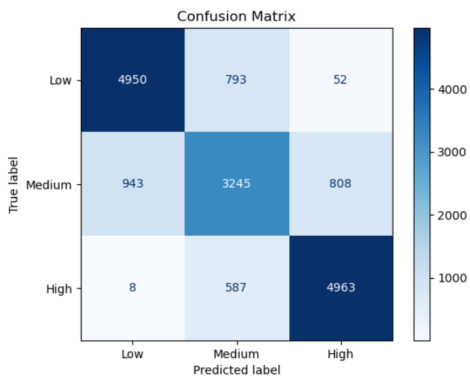

Figure 11: CNN-LSTM Hybrid Model Confusion Matrix.

This combined CNN-LSTM model really did well in terms of classification, particularly Low-risk (85 percent) or High-risk (87 percent). Such high scores indicate that the model, with time-series crime data, learned to differentiate between low-risk and high-risk categories well. However, the scores for the other classification, which is Medium Risk, were lesser (67 percent), indicating that sometimes, it misclassifies such instances due to their attributes being on the borderline between low and high risk. This could be a misclassified feature description or confusing temporal trends within medium-risk areas.

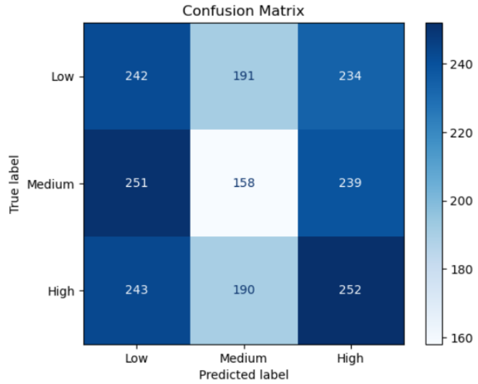

Figure 12: LSTM Model Confusion Matrix.

The pure sequential crime data feed into the standalone LSTM model, yielding good overall results, particularly among the specific classes like High-risk where it scored 88 percent. This means that, given only temporal patterns, crimes could be identified very well, especially those where risk patterns remain constant over time. However, its performance on Low and Medium classes is not stellar, and it cannot receive benefits from environmental context provided by the hybrid model.

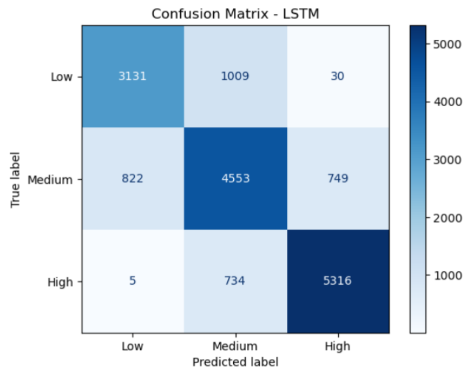

Figure 13: CNN Model Confusion Matrix.

However, the CNN architecture using only Google Street View image-based features has obtained rather low performance results in excess of 32.6 percent accuracy on average, failing to draw distinctions between crime risk levels. Although still pictures of an area feature may provide contextual clues, other dynamic and time-sensitive indicators are critical for risk predictions. This is why the CNN model incorrectly classifies samples mainly into Medium risk, proving that spatiotemporal features should work together.

The model is predicting future average monthly crime patterns across Low, Medium, and High risk levels using 12-month crime sequences. Here the crime counts for a particular category over time are presented as lines, with the shaded regions indicating standard deviation to capture variability across H3 cells. High-risk zones consistently show highly elevated and more variable crime activities, while low-risk zones are stable confirming the model's effectiveness in risk classification from temporal patterns.

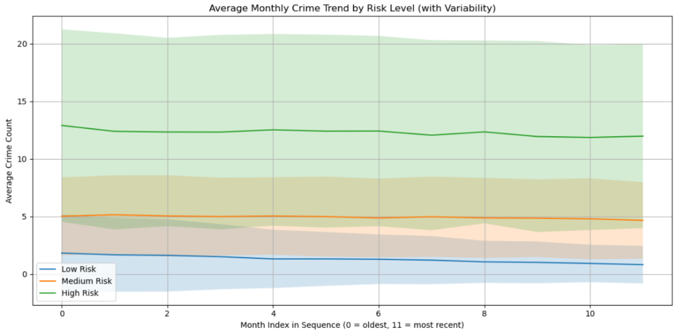

Figure 14: Average Monthly Crime Trend by Risk Level.

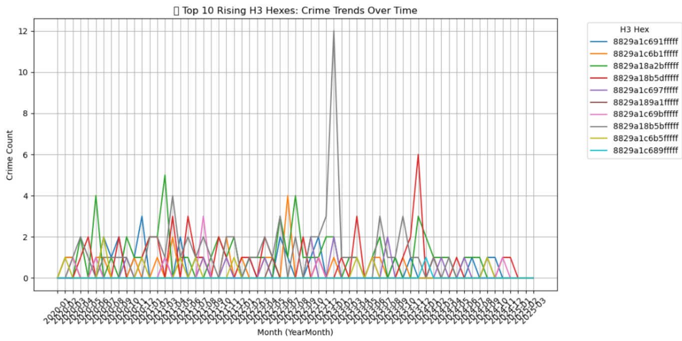

Figure 15: Top 10 Rising H3 Hexes: Crime Trends Over Time.

This graph illustrates the predication in future crime increase for top 10 H3 hexagonal cells in the dataset, which records services from its commencement until end. Each of the lines represents the monthly crime counts of incidences within the hours which are recorded in Year-Month format on the X-axis. The biggest positive change selected some hexes, whose lines show sharp spikes while others increase gradually. This visualization will identify emerging hotspots, allowing resource provision to reach areas that already show evidence of an escalation in crime rather than just historical high crime areas.

6.3 Interpretation

The result of evaluation, according to which it is said that hybrid model CNN-LSTM is the best predictor of crime risk levels use environmental contexts and past trends together for accurate prediction. Alone LSTM shows one or two results implying some independence from the temporal sequence, while CNN may be useful for space features only, lacking time-series context. On the contrary, these results give evidence that the most notable misclassifications were between Medium and its adjacent classes. Hence, we presume more complex feature representations or some additional input variables (such as income inequalities in neighborhoods, police activity data) could be necessary.

6.4 Discussion

The CNN-LSTM hybrid model displayed significant advantages over the individual CNN and LSTM, proving that the simultaneous utilization of spatial input from Google Street

View images and temporal modeling of crime trends improved substantially in accuracy. The CNN extracted environmental cues, while the LSTM modeled seasonal trends and historical patterns. Medium-risk zones were classified with great difficulty due to overlaps in patterns. Past police reports could bias the representations towards levels of policing rather than real levels of crime. Although Folium-based interactive mapping was intuitive for visualization for different stakeholders, actual implementation would require frequent retraining on live data while integrating socioeconomic parameters to enhance fairness and adaptability.

7 Conclusion and Future Work

This research includes the model of risk forecasting in deep learning crime and visualization for Los Angeles, in conjunction with CNNs for horizontal extraction of spatial attributes from digital street images and with LSTM modeling temporal trends of historical LAPD crime event data. Crime spatial encoding has been executed with uniform partitioned H3 hexagons, while the rest of CRISP-DM has guided stages of data preparation, modeling, evaluation, and deployment. This CNN-LSTM hybrid model proved more effective than the independent baselines of the CNN and LSTM, forecasting through an interactive Folium map in future connection to the Flask API for near-real-time updates. Systems design has included ethical considerations regarding privacy protection and bias mitigation. Future work may include adding such enhancements, like streaming live data for real-time forecasts, and crime-type classifications into the prediction scheme, taking multiple image views per location, and supplementing input with socio-economic and demographic data. Other potential research work includes testing the model across many cities, attaching fairness metrics to flag bias, and building applications for end-user engagement, such as mobile safety tools or navigation-based alerts. Further improvements in performance could be obtained by using advanced architectures, such as spatio-temporal graph neural networks or transformer-based models, which provide further flexibility and accuracy together with ethical accountability for proactive policing or safety planning in cities.

References

[1] S. Wang and K. Yuan, "Spatiotemporal analysis and prediction of crime events in atlanta using deep learning," *2019 IEEE 4th International Conference on Image, Vision and Computing (ICIVC)*, pp. 346–350, 2019.

[2] N. Jiang, K. Miao, Y. Chai, D. Lu, and J. Wu, "Spatio-temporal prediction of crime based on data mining and lstm network," in *2023 IEEE 6th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, vol. 6, 2023, pp. 672–676.

[3] M. Vijayalakshmi and R. Norbu, "Smart police: A hybrid deep learning model for crime proactivity assessment," in *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 2023, pp. 1–7.

[4] K. B. Sundharakumar, N. Bhalaji, and M. Muthu Palaniappan, "Spatio temporal modelling of repeat crimes and hotspots prediction using recurrent deep neural net-

works," in *2024 5th International Conference on Innovative Trends in Information Technology (ICITIIT)*, 2024, pp. 1–6.

[5] Q. Dong, R. Ye, and Y. Fan, "Impact of spatial correlation on crime prediction in communities with different crime densities," in *2022 IEEE 8th International Conference on Computer and Communications (ICCC)*, 2022, pp. 2227–2233.

[6] S. Sharma, A. Uniyal, P. Srinivasan, and S. Chaudhari, "Fuzzy based geo-spatial crime category prediction for crime mapping and safe route travel," in *2022 IEEE Region 10 Symposium (TENSYMP)*, 2022, pp. 1–6.

[7] D. Song, "Research on sarima-lstm crime prediction model based on nonlinear combination of rbf neural network," in *2024 Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC)*, 2024, pp. 418–425.

[8] A. Tanvin, S. Rahman, and M. M. A. Hashem, "A generalized additive decomposition based deep neural network architecture for time series forecasting problems," in *2023 6th International Conference on Electrical Information and Communication Technology (EICT)*, 2023, pp. 1–6.

[9] N. Tasnim, I. T. Imam, and M. M. A. Hashem, "A novel multi-module approach to predict crime based on multivariate spatio-temporal data using attention and sequential fusion model," *IEEE Access*, vol. 10, pp. 48 009–48 030, 2022.

[10] D. M S and S. Shankaraiah, "Crime analysis and prediction using machine learning algorithms," in *2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon)*, 2022, pp. 1–7.

[11] S. V. S, J. Retna Raj, A. A, S. Srinivasulu, Gowri, and Jabez, "Crime analysis framework for predicting criminal behavioral patterns with machine learning," in *2023 IEEE Renewable Energy and Sustainable E-Mobility Conference (RESEM)*, 2023, pp. 1–5.

[12] B. A. Thomas and S. Raja, "Crime mapping and predictive analysis of crimes in maryland, usa," in *2024 International Conference on Signal Processing, Computation, Electronics, Power and Telecommunication (IConSCEPT)*, 2024, pp. 1–6.

[13] V. Mandalapu, L. Elluri, P. Vyas, and N. Roy, "Crime prediction using machine learning and deep learning: A systematic review and future directions," *IEEE Access*, vol. 11, pp. 60 153–60 170, 2023.

[14] K. T. M, L. T. N, M. Ithihas, N. R. Shetty, A. H. N, and S. Hebbar, "Crime type and occurrence prediction using machine learning," in *2024 Second International Conference on Advances in Information Technology (ICAIT)*, vol. 1, 2024, pp. 1–5.

[15] H. K. Sharma and R. M. Tailor, "Exploring the potential of ai for urban crime prediction in india: A case study of indian metropolitan cities," in *2023 6th International Conference on Contemporary Computing and Informatics (IC3I)*, vol. 6, 2023, pp. 1047–1051.