CRIME RATE PREDICTION FROM STREET VIEW IMAGES USING CONVOLUTIONAL NEURAL NETWORKS AND TRANSFER LEARNING

SREE VENKATA SATYA PRANEETH KADIYAM
[August 2021]

SUPERVISORS:
Dr. Mingshu Wang
Dr. Claudio Persello

CRIME RATE PREDICTION FROM STREET VIEW IMAGES USING CONVOLUTIONAL NEURAL NETWORKS AND TRANSFER LEARNING

SREE VENKATA SATYA PRANEETH KADIYAM
Enschede, The Netherlands, [August 2021]

Thesis submitted to the Faculty of Geo-Information Science and Earth Observation of the University of Twente in partial fulfilment of the requirements for the degree of Master of Science in Geo-information Science and Earth Observation.
Specialization: Geo-Informatics

SUPERVISORS:
Dr. Mingshu Wang
Dr. Claudio Persello

THESIS ASSESSMENT BOARD:
Dr. R. Zurita Milla
Dr. Qunshan Zhao, School of Social & Political Sciences, University of Glasgow (UK)

DISCLAIMER

This document describes work undertaken as part of a programme of study at the Faculty of Geo-Information Science and Earth Observation of the University of Twente. All views and opinions expressed therein remain the sole responsibility of the author, and do not necessarily represent those of the Faculty.

ABSTRACT

Until recently, street view imagery is not considered a data source for scientific research. With growing interest in deep learning and computer vision, street view imagery evolved as a novel data source due to its fine resolution and rich visual scene content. They replace the tedious field surveys with virtual audits. Of late, street view imagery is used to relate visual perception of predicting non-visual attributes like building age estimation, property evaluation, walking likelihood etc. In addition to these, a few research works also used street view imagery to predict crime rates. Predicting crime rate from street view imagery is based on famous environmental theories like Broken Windows theory or Routine Activity of Places theory. They state that environmental variables influence crime occurrence. The fast-paced urbanisation and growing population can motivate criminals and encourage crime occurrences in cities. There is a need to manage the resources of the law enforcement department effectively to control the crime. This research takes the motivation from the theories mentioned above works and investigates the effect of visual variables from street view imagery on predicting crime rates.

Previous research mainly concentrated on classifying crimes based on the severity or ranked the most occurred crime in each place. This work tries to predict the crime counts of four different types from the street view imagery by solving a multi-output regression problem. Greater London is selected as the study area of research, and the crime data of one year is considered. A deep learning model is built to achieve this, taking multiple inputs, and simultaneously predicting crime counts for four different crime types. ResNet18 is used as a building block for building the model. A workflow is designed to model the crime data and prepare the labelled dataset for input to the built model. Kernel Density Estimation is used to model the crime data, and the outputs are used to extract the street view imagery and label the data. Four street view images and population density are given as inputs, and the crime rates of burglary, robbery, other thefts and vehicle crimes are predicted simultaneously. Different configurations of models are trained and compared to understand the effect of visual variables in crime rate prediction. The results obtained show a considerable relationship between visual variables of the built environment and crime rate. The R-squared value for burglary is 51%, robbery is 44%, other thefts is 50%, and vehicle crimes is 49%. However, there were no significant changes in the R-squared values, excluding population density as an explanatory variable. The scatterplots of actual and predicted crime rates are interpreted to understand and evaluate the model's performance. The inclusion of additional variables like socio-economic variables might have affected the performance of the model.

Keywords: Crime rate prediction, street view image, deep learning, multi-output regression, kernel density estimation

ACKNOWLEDGEMENTS

I take this opportunity to thank those who have be helped and encouraged me throughout my study and research.

First, I would like to express my sincere gratitude to my supervisors Dr. Mingshu Wang and Dr. Claudio Persello, for their expert guidance and valuable comments throughout my research. Their suggestions throughout the journey helped me in shaping my thesis and see to its completion. Interaction with them is always a new learning and has influenced me to look at the problem from a new perspective. Without their guidance the journey wouldn't have been the same.

I thank the faculty at ITC who have helped me learn new things constantly and explore new areas.

I would like to thank Sameer, Krishna, Satya, Abbas, and Vijay for their engaging discussions and suggestions related to my research.

I would like to thank Dhananjay, Darshana, Harsh, Sowmeya and Ranit for supporting and keeping me sane throughout my study and research time.

Finally, I would like to thank my parents for their unconditional love and support.

TABLE OF CONTENTS

1. Introduction 7
   - 1.1. Background and Motivation 7
   - 1.2. Research gap identification 8
   - 1.3. Research Objectives and Research Questions 8
   - 1.4. Thesis structure 9
2. Literature review 10
   - 2.1. Crime prediction 10
   - 2.2. Visual Scene Analysis 11
   - 2.3. Crime prediction with SVI 13
   - 2.4. Multi-output regression 14
3. Study Area and Datasets 15
   - 3.1. Study Area 15
   - 3.2. Data Description 16
4. Methodology 19
   - 4.1. Software 20
   - 4.2. Data preparation 20
   - 4.3. Model design and training 26
   - 4.4. Model Evaluation 29
5. Results and Discussion 30
   - 5.1. Results of data preparation 30
   - 5.2. Results of CNN model 34
   - 5.3. Discussion 42
   - 5.4. Limitation 44
6. Conclusion and Recommendations 45
   - 6.1. Conclusion 45
   - 6.2. Recommendations 46

LIST OF FIGURES

Figure 3.1: Study area location 15
Figure 3.2: An example of Street View Images downloaded. It can be observed that the images obtained are parallel and perpendicular to the road. 18
Figure 4.1: Workflow 19
Figure 4.2: An illustration of kernel density estimation (Hart & Zandbergen, 2014) 21
Figure 4.3: The density for the cell is estimated considering all the crime incidents inside the bandwidth distance, by weighing them using the kernel function applied. 22
Figure 4.4: The road segments intersecting the grid lines are removed and SVI collection points are generated so that they do not belong to the same road segment. It is also to make sure that two points from different cells are not too close. 23
Figure 4.5: The solid black line is the road and red dotted lines are the parallel and perpendicular directions to the road. ϴ is the bearing of the road with true north. The bearing is calculated and added to the parameters to obtain the SVI in desired direction. 24
Figure 4.6: a) The frequency distribution of crime rates of selected crime types before logarithmic transformation. The value ranges are different for each crime type. b) The frequency distribution of crime rates after logarithmic distribution. Now the values are considerably in the same range, and this helps in training process. 25
Figure 4.7: Residual block and usage of skip connections in ResNet (He et al., 2015) 26
Figure 4.8: 4-CNN model implemented using ResNet18 convolutional backbones. The designed model takes 5 inputs including population density and gives crime counts as outputs for four crime types: burglary, robbery, other thefts, and vehicle crimes. 27
Figure 5.1: Burglary crime rate map 30
Figure 5.2: Robbery crime rate map 31
Figure 5.3: Other thefts crime rate map 31
Figure 5.4: Frequency distribution of crime rates of a) burglary b) robbery c) other thefts and d) vehicle crimes 32
Figure 5.5: Vehicle crimes crime rate map 32
Figure 5.6: Map showing training, validation and testing cells split 33
Figure 5.7: Scatterplots of actual vs. predicted values (4 images) for Model 1 35
Figure 5.8: Scatterplots of actual vs. predicted values (8 images) for Model 1 35
Figure 5.9: Scatterplots of actual vs. predicted values (4 images) for Model 2 36
Figure 5.10: Scatterplots of actual vs. predicted values (8 images) for Model 2. 37
Figure 5.11: Scatterplots of actual vs. predicted total crime rates. 39
Figure 5.13: Predicted crime rates - Robbery 40
Figure 5.12: Predicted crime rate - Burglary 40
Figure 5.14: Predicted crime rate – Other thefts 41
Figure 5.15: Predicted crime rate – Vehicle crimes 41
Figure 5.16: Actual and predicted values for a set of images. The predicted values are closer to the actual values. 43
Figure 5.17: Actual and predicted values for street view images. There is a huge difference between actual and predicted values 44

LIST OF TABLES

Table 3.1: Overview of datasets ... 16
Table 3.2: Description of attributes in the crime data ... 16
Table 3.3: Parameters to download street view images ... 17
Table 4.1: Dataset split for training, validation, and testing ... 26
Table 5.1: Evaluation metrics of the testing set for Model 1. Metrics calculated are per SVI collection point (4 images) ... 34
Table 5.2: Evaluation metrics of the training set for Model 1. Metrics calculated are per SVI collection point (4 images) ... 34
Table 5.3: Evaluation metrics of the testing set for Model 1. Metrics calculated are for single-cell (8 images) ... 35
Table 5.4: Evaluation metrics of the training set for Model 1. Metrics calculated are for single-cell (8 images) ... 35
Table 5.5: Evaluation metrics of the testing set for Model 2. Metrics calculated are per SVI collection point (4 images) ... 36
Table 5.6 Evaluation metrics of the training set for Model 2. Metrics calculated are per SVI collection point (4 images) ... 36
Table 5.7: Evaluation metrics of the testing set for Model 2. Metrics calculated are per single cell (8 images) ... 37
Table 5.8: Evaluation metrics of the training set for Model 2. Metrics calculated are per single cell (8 images) ... 37
Table 5.9: Evaluation metrics of the testing set for Model 3. Metrics calculated are per SVI collection point (4 images) ... 38
Table 5.10: Evaluation metrics of the training set for Model 3. Metrics calculated are per SVI collection point (4 images) ... 38
Table 5.11: Evaluation metrics of the testing set for Model 3. Metrics calculated are per single cell (8 images) ... 38
Table 5.12: Evaluation metrics of the testing set for Model 3. Metrics calculated are per single cell (8 images) ... 38

1. INTRODUCTION

1.1. Background and Motivation

The occurrence of crime in a neighbourhood is a threat to public safety and livelihood. In major cities, with a growing population day by day, there is a scope of uncontrolled criminal activity. It is difficult and impractical for the police force to keep track of criminal activities all over the city. The crime rate per 1000 inhabitants has increased since the last decade in London (Clark, 2021). On the other hand, there is no considerable increment in the police force strength in the statistics published by London Datastore (https://data.london.gov.uk/). The disparity in crime and the police force emphasizes the need to understand the crime and its distribution, identify the hotspots of crime, and effectively manage the police force's resources. Over the years, many theories have been proposed to explain crime and criminal behaviour.

The famous Broken Windows Theory (Wilson & Kelling, 1982) suggests that the environment strongly influences the behaviour of its people. Environmental theories consider that along with the offender, spatio-temporal setting and victim also influence the crime event (Brantingham & Brantingham, 1995). The crime occurrence depends on a multitude of factors, and its distribution is non-random. These factors are categorized into crime attractors and crime generators based on their interaction with the crime event (Kinney et al., 2008). The crime occurrence varies from place to place and is regulated by appearance and perception of the built environment and other determinants. In addition to crime, built environments also influence other variables like health (Cohen et al., 2003), education (Milam et al., 2010), and mobility (Piro et al., 2006). The theories proposed in the past are formulated either by social experiments or physical audits. Understanding the built environments and their relation to crime patterns can help control and prevent crime. This research attempts to quantify the relationship between crime occurrence and built environments using Street View Imagery (SVI).

Street View Imagery (SVI) is a 360° panoramic image taken at eye level. It captures the visual scene of the built environments and can be the best substitute for human perception. Though SVI is not introduced for research originally, the potential in research is discovered in the recent past. The excellent resolution of images provides great information about the neighbourhood's appearance, which is the key to understand the built environments. A few of the data sources include Google Street View (GSV), Tencent Street View (TSV), Mapillary etc. With the exhaustive amount of SVI data in hand, now the audits for understanding built environments can be conducted virtually by trained experts (Kelly et al., 2013) and crowdsourcing (Salesses et al., 2013).

However, virtual audit on such huge data is still an unrealistic task. Computer vision and Deep Learning help handle such massive data. Computer vision deals with how a computer sees images and understands them. It can be an alternative for human vision in a cognitive understanding of the scene when trained with the data sets in a task-specific manner (Ibrahim et al., 2020). Deep learning, Convolutional Neural Networks (CNN), to be specific, made computer vision perform the tasks of classification, segmentation, feature extraction from the images more precise and accurate (Lecun et al., 2015). Machine learning techniques and computer vision advancements aid in understanding built environments and quantifying crime occurrence. As it is difficult for even an expert to understand the impact of the environment on crime for such vast amounts of data, this study will use a deep learning model to learn the association between visual variables of the scene and crime occurrence.

In recent years, due to its cost-effectiveness and ease of availability, SVI substitutes human perception in understanding the built environments. With technological advancements, computer vision and deep learning have progressed noticeably in the last decade. SVI and computer vision models are used together to relate urban built environments with physical urban change (Naik et al., 2017), safety (Dubey et al., 2016), physical activity (Kelly et al., 2013), urban mobility (F. Zhang et al., 2019), building age (Li et al., 2018), etc. Crime rate prediction and forecasting usually require fine-grained data about the crime events and their affecting factors. However, the availability of such fine-grained data is not possible in all scenarios. This research attempts to find out if SVI can explain the number of crimes for a location. It is noteworthy that the number of crimes is a non-visual attribute that model must predict. As mentioned above, if the fine-grained data about the area is not available, this model can be used to predict the crime rates and identify hotspots. The results and findings are helpful, not only to the police but also to other decision-makers like urban planners. The next section discusses the research objectives and research questions.

1.2. Research gap identification

Few researchers worked on predicting crime and crime rates from SVI. Dubey et al. (2016) tried to rank the streets on the perception of safety, from SVI, by crowdsourcing and the Convolutional Neural Networks (CNN) model. Andersson et al. (2017) used a Siamese CNN model to classify crime rates using SVI into four types based on the intensity of crimes. Kang & Kang (2017) predicted the crime rate using multi-modal analysis by taking multiple variables, including spatial, temporal, and socio-economic factors. Fu et al. (2018) developed a CNN architecture to rank crime types using a preference learning technique. Almost all the studies; treated crime prediction as a classification problem or a ranking problem. However, there is a limitation with the classification of the crime. Though we get to know the intensity of crime, a slight deviation in the threshold can change the class. At times, it is necessary to know the quantity to make clear decisions. This work looks at predicting the crime rate from SVI as a regression problem. The idea is to simultaneously predict the crime rates of four different crimes from SVI by solving a multioutput regression problem. Instead of predicting a class, this research tries to predict the crime count given the SVI.

1.3. Research Objectives and Research Questions

1.3.1. Main Objective

The main objective is to build a deep learning model to simultaneously predict crime rates of different crime types from SVI in one year. This research attempts to quantify the relationship between visual variables of the environment and crime rate by solving a multi-output regression problem.

1.3.2. Specific objectives

1. To model the crime data distribution in the study area and label SVI.
2. To choose one state-of-art architecture to learn features of SVI.
3. To implement the deep learning model to predict the crime rate.
4. To assess the performance of the developed model.

1.3.3. Research Questions

1. To model the crime data distribution in the study area and label SVI.
    1. What crime types are to be considered for the study?
    2. How to model the distribution of crime data?
    3. What is the best strategy to label the SVI?
2. To choose one state-of-art architecture to learn features of SVI.
    1. Which CNN architecture best quantifies the relationship between SVI and crime occurrence?
    2. How to achieve the multiple-output regression?
3. To implement the deep learning model to predict the crime rate.
    1. How to implement the model to handle multiple inputs and multiple outputs?
    2. How to select training, validation and test sets to train and configure the deep learning model?
4. To assess the performance of the developed model.
    1. To what extent can the visual variables from SVI explain the crime occurrence?

1.4. Thesis structure

The thesis structure is as follows
Chapter 2 reviews the literature and describes necessary theoretical principles related to the study.
Chapter 3 introduces the study area of the research and the datasets used to achieve the objective.
Chapter 4 explains the workflow and the methodology of the thesis.
Chapter 5 presents the results and ends with a discussion and critical findings.
Chapter 6 concludes the thesis with conclusions and recommendations for future work.

2. LITERATURE REVIEW

This chapter provides a comprehensive overview of the related work and the concepts used in the research.

2.1. Crime prediction

The occurrence of crime is non-random and often is affected by a multitude of factors. There is a need to analyse the non-randomness of crime occurrence. Crime prediction models are the approaches or techniques which help us to analyse and understand the crime patterns. Although it is impossible to predict the location and time of occurrence, we can understand the reason for crime patterns and lower the risk in the future. Crime prediction is made by following different methods: density estimation, machine learning and deep learning. This research mainly concentrates on the Kernel Density Estimation method (KDE), an approach of density estimation to model the crimes in the study area. So, the concepts related to KDE are reviewed and discussed in this section. Hotspots in crime analysis generally refer to places with a high concentration of crime events either in space or time or both. The concentration of crime events in a location is represented as a heatmap to easily identify regions with high and low crimes. The historical crime data of a given location in space and time is used to generate crime heatmaps. Hotspot identification is made by two approaches based on aggregated crime event locations and analysis of individual crime events (Hart & Zandbergen, 2014). The aggregated crime events techniques usually use a uniform grid or geographical boundaries to aggregate the crime counts and produce thematic maps. A wide range of methods is present in the current literature for aggregated crime events approach. KDE is one of the methods and is proven to give better results (Chainey et al., 2008).

KDE estimates the probability density function (PDF) of a random variable which is the outcome of a random process. In spatial analysis, KDE is used to smoothen the point pattern (in this case crime event locations) and create a density map. Hart & Zandbergen (2014) used KDE for hotspot mapping and crime prediction. The data used for the study is the crime data of four different crime types in the jurisdiction of the Arlington (Texas) Police Department between 2007 and 2008. The hyperparameters involved in KDE are grid cell size, kernel function and bandwidth. The authors have experimented with 12 different combinations of kernel function and bandwidth, four settings for kernel function (uniform, linear, normal, and quartic) and three settings for bandwidth. For each combination, KDE is implemented, and the crime density maps are generated. A benchmark definition for the hotspot is required to assess the prediction accuracy. For the study, authors have defined hotspots as grid cells that exceeded the sum of average density scores and 1.96 times the standard deviation of density scores. The metrics used for measuring accuracy are Hit Rate (HR), Predictive Accuracy Index (PAI) and Recapture Rate Index (RRI). The authors conclude their work by following recommendations. First, use quartic or linear functions as they performed consistently better than uniform and normal kernel functions. Second, they recommend a cell size of one-third of the block-face of the study area. Though the predictive accuracy is not improved, the generated hotspot map may have better visual quality. Third, usage of a small bandwidth to predict future crimes generally decreased with an increased search radius.

In their work, Hu et al. (2018) explored the inclusion of temporal dimension with spatial dimension in KDE for crime prediction. The method is called as Spatio-Temporal Kernel Density Estimation (STKDE). The study area for the research is the City of Baton Rouge and focuses on residential burglaries for time-period 2011. Like KDE, STKDE also has the same hyperparameters but is modified to add temporal dimension. A temporal bandwidth is also considered in addition to spatial bandwidth. For bandwidth selection a data-driven optimization approach is followed. Instead of randomly experimenting with the bandwidths, the spatial and temporal bandwidths are selected using the inferences from the recommended data in different disciplines (Horne & Garton, 2006; Z. Zhang et al., 2011). The approach minimizes the Kullback-Leibler loss which measures the distance between two PDFs (Hall, 1987). The study area is overlayed with a 100m by 100m grid and Epanechnikov kernel function (Epanechnikov, 1969) is used as a kernel function. Simulations are run to test the statistical significance of the predicted hotspots. The output is a raster with significant hotspots where crime is more likely to occur in a time window in the future. PAI curve is used as an evaluation metric for the study that gives a comprehensive overview of accuracy variation with different PAI values and factors affecting a PAI value. The performance is compared against two methods i) baseline spatial KDE (SKDE), regular KDE without time component ii) ProMap, developed by Bowers et al., (2004) and is easy to implement. The STKDE model outperforms the other two models significantly. The STKDE model identifies 14 hotspots, whereas SKDE and ProMap identified 11 hotspots each. The data-driven approach for bandwidth selection and simulations for obtaining statistically significant hotspot cells had an impact on the result. They also proposed the PAI curve as an accuracy metric rather than using the traditional PAI value. In addition to these works, there is a growing literature on using KDE for spatial crime analysis. It is one of the most chosen models for crime prediction and is used as a baseline method in few works (Kounadi et al., 2020). The current research adapts the KDE method to generate the crime density map which is used for further analysis.

2.2. Visual Scene Analysis

Deep learning and computer vision techniques are used to quantify the visual and non-visual attributes based on urban perception. SVI is used as a substitute for urban perception due to its fine resolution and neighbourhood representation. A few related works which use SVI for visual scene analysis have been reviewed and discussed in this section.

Salesses et al. (2013) used SVI to perceive the safety, uniqueness, and wealth attributes of a neighbourhood. They collected the SVI from four cities New York, Boston, Linz, and Salzburg. The data was prepared by crowdsourcing, where the participants are posed questions. The question posed was either of the three: i) Which places looks safer? ii) Which place looks more upper-class? iii) Which place looks unique? The responses for the evaluative question were either of the images which were randomly chosen from the dataset. The dataset was used to understand the relationship between visual appearance and the attributes considered. It is observed that the result is not affected by the difference in the age, gender, or location of the participant but the differences in the visual appearance in the images. Moreover, the perception is significantly different in cities in the United States of America compared to their counterparts in Europe. Naik et al. (2014) employed a computer vision algorithm to quantify the perceived safety from SVI using the dataset created by the work of Salesses et al. (2013). The preferences of images in the dataset are converted to ranked scores by Microsoft Trueskill Algorithm and are used in training the predictor. First, the features responsible for the variation in the score, like buildings, ground, trees, and sky, are extracted from the images. The extracted features were used along with scores are used to train the Support Vector Regressor (SVR). Similarly, binary classification is also performed with the same dataset by assuming a threshold in the score to classify it as high and low. The models tend to perform better for both regression and classification problems. The results obtained show that the visual appearance of the urban environment had an impact on the neighbourhood perception. However, the work has few limitations as it cannot be generalized to all cities because the dataset contains images from only four cities.

Later Dubey et al. (2016) extended the work to a global scale by collecting the data from 56 cities from different countries spread across all the continents. They created a web interface similar to the game created by Salesses et al. (2013) to prepare the data. They call the dataset Place Pulse 2.0. The dataset contained 1.17 million pairwise comparisons for 110,998 images. This dataset is prepared for the preference of different perception attributes like safety, healthy, lively, depressing, boring, and beautiful. The preferences are converted to ranked scores using the Trueskill algorithm to train the deep learning model. The authors propose a Siamese (Chopra et al., 2005) like network that shares the parameters to learn the pairwise comparisons.

The same network is extended, and a ranking sub-network is added to learn the pairwise comparisons and to rank simultaneously. The proposed CNN performed better compared to other pre-trained models. The research was mainly related to ranking and comparing street view images and studies the connection between urban appearance and visual perception. Similarly, Naik et al. (2017) used SVI and computer vision to measure the changes in urban appearance. In this study, time-series street-level imagery is used to assess and quantify the built environments' changes. The images captured in 2007 and 2014 were compared to observe the change in physical appearance. The streetscore is estimated using the algorithm built by Naik et al. (2014). The street scores are then compared to observe the changes in appearance. The measured changes are then correlated with neighbourhood characteristics of built environments to predict the variables responsible for the physical change. The results of the research concluded that education and population density affect the changes in the neighbourhood. Though the study was restricted to few cities in the north-eastern United States, they quantified the visual attributes accurately. It was observed that neighbourhoods with good socio-economic status and education tend to improve over time compared to the neighbourhoods with poor education and population density.

Another such remarkable work by Khosla et al., 2014 explored the SVI's capability and deep learning to predict the distances of fast-food restaurants and hospitals from the visual scenes of establishments. The idea is to predict the closest establishment based on visual cues from the SVI. The authors experimented with different descriptors like GIST, texture, colour, and the deep learning descriptor (the layer before the fully connected layer (FCN)) as features from the SVI. The distance to the closest establishments is calculated and considered as labels for the SVI extracted. The four images extracted from a point are considered as individual inputs with the same labels. The features and labels are used to train the SVR to predict the distances to establishments. In addition to finding the closest establishments, the authors also experimented to find out the crime rate in the area based on the visual features extracted from SVI. The deep learning model's accuracy seems to be better than the human tests in both cases. Likewise, Li et al. (2018) used SVI to estimate the building age. The authors treated the building age estimation as a regression problem and used pre-trained architectures to estimate the building age. The study area for the research is the North and West Metropolitan Region of Victoria, Australia. CNN architectures AlexNet (Krizhevsky et al., 2012), ResNet18, ResNet50 (He et al., 2015), DenseNet161 (Huang et al., 2017) pre-trained on places365 dataset (Zhou et al., 2018) are used for feature extraction from the images. The extracted features are input to the SVR to estimate the building ages. The inferences from the results obtained are that the appearances of the building impacted the building age, and the deeper models performed better compared to the shallower models. It is evident from the above-mentioned research works that the image regression of SVI gives better results for the estimation of non-visual attributes.

2.3. Crime prediction with SVI

The work mentioned above is closely related to understanding urban built environments based on appearance and perception. A few works explored the possibility of predicting crime from SVI. Doersch et al. (2012) proposed a methodology to find the visual elements from GSV images geographically distinctive to specific cities. Using the Nearest Neighbour algorithm, the authors used a Histogram of Gradient (HOG) and colour components as feature descriptors and clustered the data into positive and negative data. Later trained a Support Vector Machine (SVM) detector for classification and achieved appreciable accuracy for the selected cities. Arietta et al. (2014) adapted the model by Doersch et al. (2012) to predict the relationships between visual elements in the SVI and non-visual attributes for the neighbourhood like theft rates, population density, housing prices, and perception of danger. The non-visual attributes are interpolated over the city. The features responsible for the corresponding attribute values are identified, and then the SVR is applied to estimate the non-visual attribute. The authors compared HOG and colour descriptors with Caffe's ImageNet CNN model and concluded that the latter captured the city semantics more effectively than the former descriptor. These works laid the foundation to study and understand the crime from SVI.

Andersson et al. 2017 proposed a 4-Cardinal Siamese CNN (4-CSCNN) inspired by the work of Dubey et al. (2016) to classify visual scenes into four categories, from low to high crime rates. The authors labelled the data by dividing the Chicago city area into a grid of 2500 equal squares. The squares are given a label according to the intensity of crimes in a unit grid element. Later GSV images are collected at locations along the roads at a predefined interval in cardinal directions. The CNN architecture used for the 4-CSCNN is AlexNet pre-trained on ImageNet dataset. The weights of the CNN architecture are frozen to leverage the knowledge of model learning on ImageNet dataset. The outputs from the four CNNs are concatenated to form a one-dimensional vector which is then input to a Multi-Layer Perceptron (MLP). The final layer of the network has four nodes, and a softmax activation function is used. The proposed CNN takes four images corresponding to a location and predicts a class of crime. The overall accuracy obtained is 54.3% and per class average accuracy is 77%.

Fu et al. (2018)'s work predicts the crime rankings of multiple crimes from the GSV. The authors developed a new CNN StreetNet to predict the rankings. The proposed CNN is based on the preference learning framework. The model takes the SVI as input and gives the preference of crime that can happen for the given location. The study areas of the work were New York and Washington DC. They followed a new approach for retrieving SVI at a location. Instead of acquiring the images in cardinal directions, the images are obtained perpendicular to the road to capture the context of built environments. A new procedure has been proposed for data labelling. Images are labelled according to local crime density estimated within a time window and a distance of 1k and 2k feet from the street view sample points. They follow a data-driven approach to label the images to reduce the bias in the labelling. The results are compared with the other benchmark architectures AlexNet (Krizhevsky et al., 2012), VGGNet (Simonyan & Zisserman, 2015), PlacesNet (Zhou et al., 2014). The results obtained from StreetNet are better compared to the legacy architecture models. Similarly, H. W. Kang & Kang (2017) predicted crime occurrence using multi-modal data. The authors considered features extracted from SVI as visual features, socio-economic variables, and weather variables to build a deep neural network (DNN) to predict crime. Compared to Support Vector Machines (SVM) and KDE, the proposed data fusion DNN performed better in predicting crime.

2.4. Multi-output regression

As the research uses multi-output regression to predict the crime rates of four crime types simultaneously, the concept of multi-output regression is explained briefly in this section.

Multi-output regression, also called multi-target regression, is an extension of single-target regression. In single-target regression, input is a N dimensional vector, and output is a scalar value, whereas in multi-output regression, both input and output are N and M dimensional vectors, respectively (Watt et al., 2016). Consider a dataset D with a vector inputs and vector outputs. For a datapoint $(\boldsymbol{x_p}, \boldsymbol{y_p})$ of D, $\boldsymbol{x_p}$ is a vector of inputs and represented as a column vector and $\boldsymbol{y_p}$ is a vector of outputs. In matrix notation $\boldsymbol{x_p}$ can be considered as a column matrix and $\boldsymbol{y_p}$ can be represented as a row matrix as follows.

$$\boldsymbol{x_p} = \begin{bmatrix} 1 \\ x_{1,p} \\ \vdots \\ x_{N,p} \end{bmatrix} \qquad \boldsymbol{y_p} = [y_{0,p} \quad y_{2,p} \quad \cdots \quad y_{M-1,p}] \tag{1}$$

Suppose if a linear relationship exists between $\boldsymbol{x_p}$ and $y_{i,p}$, i^{th} element of $\boldsymbol{y_p}$ then it is a single-target regression, and the weights can be represented by $\boldsymbol{w_i}$ and the equation to estimate the output can be given by the following formula.

$$\boldsymbol{w_i} = \begin{bmatrix} w_{0,i} \\ w_{1,i} \\ \vdots \\ w_{N,i} \end{bmatrix} \qquad \boldsymbol{x_p} = \begin{bmatrix} 1 \\ x_{1,p} \\ \vdots \\ x_{N,p} \end{bmatrix} \tag{2}$$

$$\boldsymbol{x_p^T w_i} = y_{i,p} \tag{3}$$

Here the weights matrix $\boldsymbol{w_i}$ needs to be estimated and appropriately fine-tuned, assuming that there is a linear relationship. Similarly, in place of $y_{i,p}$ if we have a vector of outputs $\boldsymbol{y_p}$, then the weight matrix of such equation can be represented as a row matrix of M vectors as shown below.

$$\boldsymbol{W} = \begin{bmatrix} w_{0,0} & w_{0,1} & \cdots & w_{0,M-1} \\ w_{1,0} & w_{1,1} & \cdots & w_{1,M-1} \\ \vdots & \vdots & \cdots & \vdots \\ w_{N,0} & w_{N,1} & \cdots & w_{N,M-1} \end{bmatrix} \tag{4}$$

And the relationship of $\boldsymbol{x_p}$ and $\boldsymbol{y_p}$ is given by the formula,

$$\boldsymbol{x_p^T W} = \boldsymbol{y_p} \tag{5}$$

where $\boldsymbol{W}$ weights matrix of dimensions $(N + 1) \times M$, $\boldsymbol{x_p}$ is an input vector and $\boldsymbol{y_p}$ is an output vector. The weights need to be estimated to solve the multi-output regression problem. To fine-tune the weights, the cost function needs to be optimized. We can extend and use the least-squares cost function like linear regression and single-target regression. As the output is a vector, the average squared deviations are calculated and considered to optimize the weights.

There is no significant change in the implementation of multi-output regression in deep learning models. The loss functions and activation functions can be used the same as single output regression models but with additional outputs. Or the traditional regressors like Support Vector Regressors (SVR) or Random Forest Regressors can be used on the features extracted from the images.

3. STUDY AREA AND DATASETS

This chapter explains the study area and the datasets used for the research in different sections. Various types of datasets are used throughout the research. Each of them will be discussed in detail in the below sections.

3.1. Study Area

The study area of this research is in London, England. London is the capital and the largest city in the England and United Kingdom, situated in the southeast, on the banks of the River Thames. It covers 1572 square kilometres and consists of two major regions, Greater London and the City of London. It is divided into 33 administrative districts, one of which is the City of London, and others are referred to as the London Boroughs, which collectively fall under Greater London. According to the Office of National Statistics (ONS), the estimated population of London increased from 8.1 million in 2011 to approximately 9 million by mid-2018. The growth in different sectors of industry and education is a factor in attracting immigrants, making it the most populous city in England and the United Kingdom. It stands second to New York in terms of the immigrant population in the world. The London population is very diverse, with different ethnic groups and religions. On the flip side, the statistics by ONS show a rise in recorded crimes over the years. It is noteworthy to mention that the crime rates have been different across different areas of London. The dynamic situation of London and its demographic distribution made it a better choice to study crime. The whole London region is considered for the study. The study area is divided into equal units of 250 meters x 250 meters for the convenience of analysis. The choice of spatial resolution and the process followed is discussed in detail in 4.2.2.

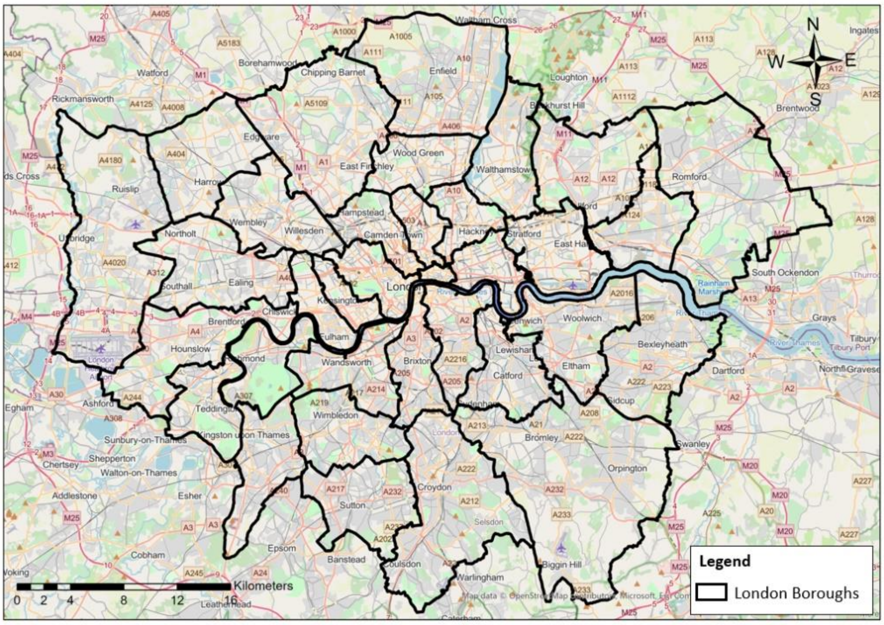


Figure 3.1: Study area location

The policing for London is provided by three forces, namely The Metropolitan Police, the City of London Police, and the British Transport Police. The Metropolitan Police is responsible for services in Greater London, whereas the City of London Police serves only the City of London. Moreover, The British Transport Police looks after National Rail, London Underground, Docklands Light Railway, and Tramlink services. The crimes reported to and reported by the City of London Police, and the Metropolitan Police are considered for the research. The details of the datasets used will be discussed in the following section.

3.2. Data Description

In this research, four types of data are used to estimate the crime rate. The data and the sources are mentioned in the following Table 3.1.

Data	Time	Source
Crime data	2018	https://data.police.uk/
Road Network	2020	https://osdatahub.os.uk/downloads/open/OpenRoads
Population Density	2011	https://data.london.gov.uk/
Geographical Boundaries	2011	https://data.london.gov.uk/
Street View Images	2018-2019	https://developers.google.com/maps/documentation/streetview/overview

Table 3.1: Overview of datasets

3.2.1. Crime data

It is the recorded crime data in London. The recorded crime data is the crime incidents reported to and reported by various police forces across the country. The data for the whole United Kingdom is available in data.uk.police website. The data is organized in months per year and is available in the CSV (comma-separated-values) file format. The data is filtered by the police forces that serve the London region. The main attributes of the data records include crime id, crime type, jurisdiction, longitude, latitude, place, area code, and time period. A brief description of each attribute is described in Table 3.2.

Attribute	Description
Crime ID	Unique identifier for each crime.
Crime type	It refers to one of the 16 crime categories defined by UK police.
Jurisdiction	The name of the police force to/by which the incident is reported.
Longitude and Latitude	The anonymised coordinates, where the incident occurred in WGS84 coordinate system (EPSG: 4326).
Place	It refers to the landmark where the incident occurred.
Area Code and Name	The Lower layer Super Output Area (LSOA) code and the name in which the incident occurred. An LSOA is a geographic hierarchy designed for better organization of small areas in England and Wales.
Time period	The month in which the incident is reported.

Table 3.2: Description of attributes in the crime data

The crime data is processed and used for further analysis. The details will be clearly discussed in sections 4.2.1 and 4.2.2

3.2.2. Road Network

The road network for the study area is acquired from the website of Ordnance Survey, the national mapping agency of Great Britain. The data is available for the whole United Kingdom in shapefile format and British National Grid (EPSG:27700) coordinate system. It is updated twice a year in April and November. The London data is available in tile TQ. The shapefile is processed in ArcGIS to select the roads to the extent of London. A Road network is required to extract the street view images. The street view images are downloaded in perpendicular and parallel directions of the road instead of cardinal directions. So, the bearing of the road is required to extract the images in the desired direction. The processing and extraction for street view images are discussed in detail in section 0. Some of the important attributes of the road network data are road identifier, class of the road, name of the road, length of the road and function of the road.

3.2.3. Street View Images

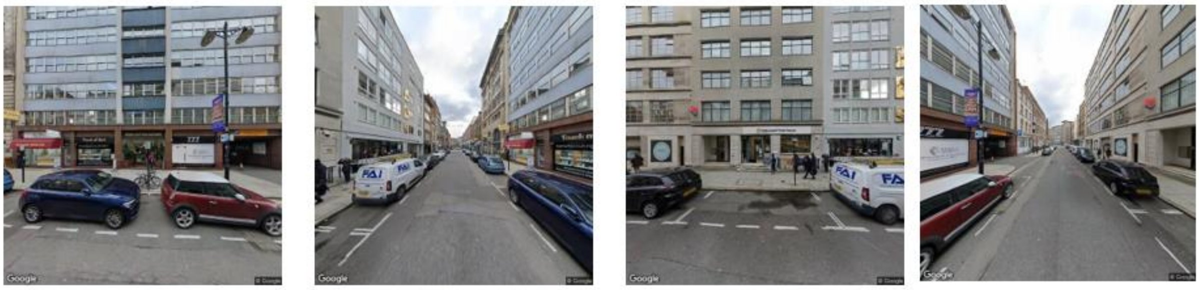
Street View Images at preferred locations are downloaded using the Google Street View Static API. The bearings of roads are calculated and used to download the images in the desired direction. A bearing of the road is the angle made by the road with the true North. Similarly, other parameters can be adjusted to obtain the desired view of the street view images. A brief description of the parameters used is mentioned in Table 3.3.

Parameter	Description
Location	It is the required location in terms of latitude/longitude values in the WGS84 coordinate system. For example, location=52.22,6.89, where 52.22 is latitude and 6.89 is longitude.
Size	It is the required size of the image and is specified as **width** x **height**. Width and Height are measured in pixels. For this work, the size of the images downloaded is 512 x 512.
Heading	It defines the direction of the image. It takes values in the range of 0 to 360. Usually, 0 indicates North, 90 East, 180 South and 270 West for cardinal directions. In this work, the bearing of the road is added additionally to acquire the images perpendicular and parallel to the roads. For example, Final heading value = cardinal directions heading + bearing. If the bearing is 2°, then heading for North is 0° + 2° = 2°. The same applies to other directions also.
FOV (Field of View)	It defines the horizontal field of view of the image. It is expressed in degrees, and the maximum allowed value is 120. In this work, the maximum allowed value of 120 is used to obtain images.

Table 3.3: Parameters to download street view images

In addition to these, there are other parameters, pitch and radius. Pitch defines the vertical field of view, and radius is the distance in meters to search for a panorama. The default values of pitch and radius are 0° and 50 meters, respectively, which are left unaltered. The street view images are downloaded using the parameter mentioned above values. After eliminating the images with no data, 148,704 images in total are selected for the analysis

The images are named after the coordinate identification number, and 0,1,2 and 3 indicate the directions North, East, South and West. The images are obtained by adding the bearing values to heading values of cardinal directions, as mentioned in Table 3.3



17874_0 17874_1 17874_2 17874_3

Figure 3.2: An example of Street View Images downloaded. It can be observed that the images obtained are parallel and perpendicular to the road.

3.2.4. Population density and Geographical Boundaries

The population density of an area is the number of inhabitants per square kilometre. The population density data is downloaded from the website of London Datastore. The population density data is available for the Output Areas (OA) in the shapefile format with geographical boundaries of OAs. According to ONS, OA is the lowest geographical level at which census data is available. An OA should have at least 40 resident households or 100 resident people. With the changing population size, the boundaries of OAs are redesigned, i.e., they are split up or merged. In addition to OAs, there are Lower layer Super Output Areas (LSOAs) and Middle layer Super Output Areas (MSOAs), formed by grouping OAs. As the name suggests, the limits of population size differ for both the Super Output Areas. In this work, the population density of OA is considered, as it is the smallest geographical unit, and it best suits the unit of analysis. The boundary files of Boroughs, MSOAs, LSOAs and Wards of London are also downloaded from London Datastore. The coordinate system of all the shapefiles is British National Grid (EPSG: 27700).

4. METHODOLOGY

This chapter explains the methodology and methods followed to perform image regression and predict crime rates from SVI. First, a brief description of the software and tools used for the analysis are mentioned, and then the methodology followed is explained. The workflow of the research is shown in Figure 4.1. The methodology can be divided into three parts for logical explanation: i) Data Preparation, ii) CNN model design, iii) Model training and evaluation. Each of them will be discussed in detail in the following sections.

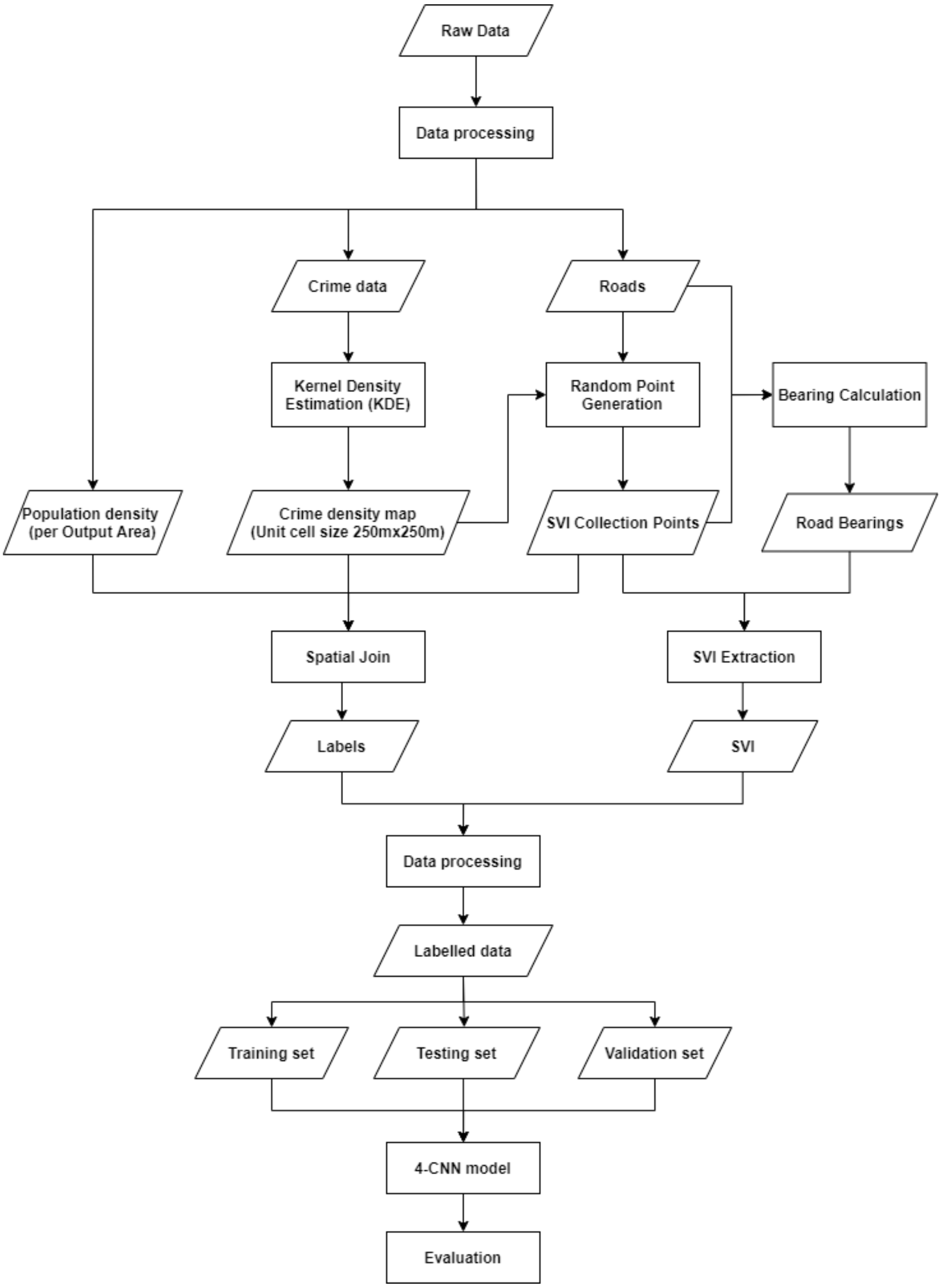


Figure 4.1: Workflow

4.1. Software

The spatial analysis part is carried out using ArcGIS software to leverage the existing tools. Postgres, database management systems are used for data handling and data cleaning in the initial stages. Once the data is cleaned and spatial analysis is performed, all the data preparation and model configuration are entirely handled in the python programming language. The Pytorch and Keras frameworks are used to build and configure the deep learning models in python. Jupyter notebook and Jupyter lab are the preferred python IDEs (Integrated Development Environment). Experiments are performed on a Windows 64-bit machine that runs on Intel Core i7-8750H with 16GB RAM and a GPU with 6GB VRAM. Once the models are finalized, the actual models are run on Geospatial Computing Platform (CRIB) by ITC, which offers GPU 32GB VRAM. The computing power is leveraged to train the models simultaneously.

4.2. Data preparation

The steps involved in data cleaning and data processing to prepare data as an input to the model are discussed in detail in this section. The dataset is not a legacy dataset; the input data must be prepared carefully for the image regression task. As the crime rates are being predicted in this work, the data is prepared to assign a crime rate to 4 images obtained per point.

4.2.1. Data pre-processing

The crime data obtained is the crime records reported to/reported by police forces all over the UK and are compiled month-wise from January to December for the year 2018. So, all the CSV files are input into the database for ease of data handling. The data is then filtered for the crimes reported to police forces serving Greater London, i.e., the Metropolitan Police and the City of London police. The reported crimes are categorised into different crime types. From different crime types, burglary, robbery, other thefts, and vehicle crimes are selected for the analysis, referred to as street crimes. Street crimes are the crime which often happens in public spaces and involves offence against people or property in a violent manner. The hypothesis is that street crimes have a better correlation with the environment compared to other crimes. So, the four crime types that fall under the street crime category are considered for analysis. Crime types like bicycle theft, violent and sexual offences also fall under street crimes but are not selected due to ambiguity in the data. Crime types like forgery, perjury etc., which are considered white-collar crimes, are excluded. Once the data is filtered with the above-stated conditions, the records with missing attributes are removed. Mainly latitude, longitude, unique identifier of crime, and the police jurisdiction reported to, are checked, and the data is cleaned accordingly to address the uncertainty. After the data cleaning, we are left with criminal records of four crime types selected recorded under the jurisdiction of Greater London police. Now, the criminal records of different crime types are segregated into four different files for further analysis.

The latitude and longitude information of the records is used to map each record as an event (point) layer using ArcGIS software. A boundary shapefile is used to check if all the points mapped are within the boundary. If there are any errors in the location information, such points are removed from the data. The same process is carried out for all four crime types. This step is to avoid the data with erroneous location information for the analysis. Then the prepared point shapefiles are used for Kernel Density Estimation.

4.2.2. Kernel Density Estimation

The point data obtained from the data pre-processing step is used to prepare crime density maps using KDE. In spatial analysis, KDE is used to smoothen the point pattern (in this case, crime event locations) and create a density map. The underlying mechanics of KDE is straightforward. Initially, a grid of equal cell size is overlayed on the study area. Then for each cell, densities are estimated based on the known crime locations. The density is estimated by a weighted distance of crime locations from the cell centre (based on a kernel function) and the search radius (bandwidth). In other words, a kernel is moved cell by cell, and density is estimated and assigned to the cell based on the kernel function and the bandwidth. A simple illustration of the KDE is shown in Figure 4.2.

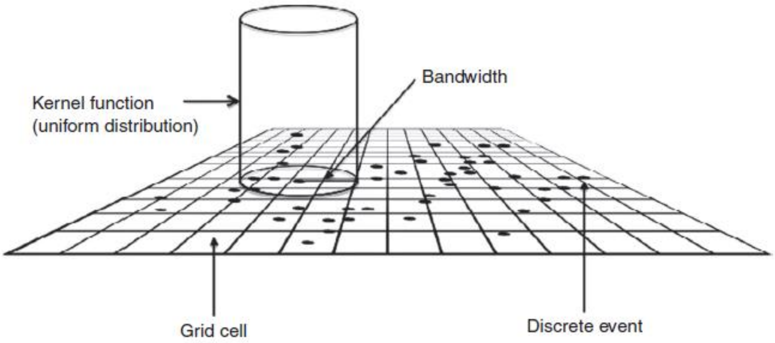


Figure 4.2: An illustration of kernel density estimation (Hart & Zandbergen, 2014)

The hyperparameters involved in the process of KDE are grid cell size, kernel function, and bandwidth. First, the grid cell size is the size of grid cells with which the study area is overlayed. The cell size affects the resolution of the resulting heatmap. Larger the cell size, the coarser the resolution, and the smaller the cell size, the finer the resolution. Additionally, it impacts the density values estimated. If the cell size is too large, there is a risk that local crime patterns or local hotspots cannot be identified. Second, the kernel function is used for interpolating the density of crime events. There are different kernel functions, namely uniform, quartic, triangle, Gaussian, etc., that are used to interpolate the weighted distances of crime locations from the centre of the cell. Uniform distribution is a flat distribution that gives equal weight to all the crime locations that fall in the search radius. Third, bandwidth to look for the crime locations. Bandwidth can also be understood as the search radius for the kernel function used. The kernel function considers the crime locations that fall within the bandwidth distance to interpolate the density value. With a change in bandwidth distance, there is a risk of under-representing the density of crimes in each cell. Therefore, bandwidth value needs to be chosen carefully.

The grid cell size and the kernel function do not impact the KDE result (Hart & Zandbergen, 2014). However, the authors suggested using linear or quartic kernels as kernel functions as they performed consistently in most of the situations. So, the quartic kernel is selected as a kernel function used by default in ArcGIS. Experimentation is carried out to select the grid cell size and bandwidth. Though the cell size does not affect the result performance-wise for KDE, it is the key for data preparation. Grid cell size is uniform over the study area and has equal length and width. Cell sizes of 100m, 250m, 500m and 1000m are considered, and the analysis is performed. A range of bandwidths is used for the analysis of each cell size. For 100m, cell size bandwidths of 150m, 200m, and 250m are considered. Similarly, for other cell sizes, bandwidth sizes within intervals of 50m of respective cell size are considered. As the density estimated is assigned to the centre of the cell, the hotspots identified in the case of 500m and 1000m cell sizes are huge and are noticed to ignore the local clustering of crime events. For 100m cell size, the hotspots identified have sharp boundaries. The bandwidths have a significant effect on the result. It is observed that the result is spiky for smaller bandwidths, and for larger bandwidths, the result is smoothened. Another major concern is the generation of random points for SVI collection. As the cell size is small, random point generation and collection of SVI, which represent the whole cell area, is a challenge. The details of the random point generation and challenges involved are discussed in detail in the next section. Due to this challenge, 100m cell size is not considered for the analysis. The remaining cell size is 250m, which addresses the challenge mentioned above and gives a reasonable output.

As mentioned earlier, the variation in bandwidths affected the result. Larger bandwidth distances smoothened the output. However, after fine-tuning, a 275m bandwidth is considered for the analysis. A 275m bandwidth covers all the eight neighbourhood cells for density estimation from the centre of a cell, as shown in Figure 4.3. The output is neither spiky nor smooth, representing the crime incidents better than the other cell sizes and bandwidths. Though the distribution of crimes is different in different crime types, the same hyperparameters chosen are the same to maintain uniformity. As the model performs multi-output regression analysis, it is necessary to input uniform data and labels for all crimes.

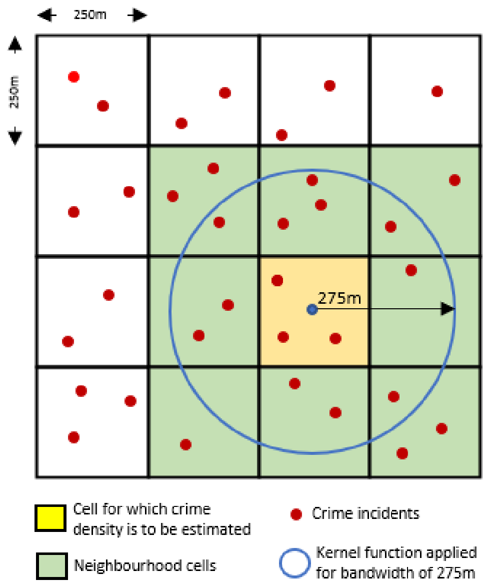

250m
250m
275m
Cell for which crime density is to be estimated
Crime incidents
Neighbourhood cells
Kernel function applied for bandwidth of 275m

Figure 4.3: The density for the cell is estimated considering all the crime incidents inside the bandwidth distance, by weighing them using the kernel function applied.

Finally, the hyperparameters chosen for KDE are 250m for grid cell size, 275m for bandwidth distance and quartic kernel function. The KDE is performed for the crime types, and the corresponding crime density maps are generated. The crime density maps generated are rasters. The value of a cell is density and not yet crime count. The density is multiplied by the area of the cell (density x 250m x 250m here) to obtain the crime count of each cell. It is to be noticed that the crime count obtained by KDE is different from the crime count obtained by overlaying the uniform grid on the study area. The crime count obtained by KDE also considers the neighbourhood effect, which is accounted by bandwidth in KDE. The result generated by KDE is the key to prepare labels for the SVI.

4.2.3. Random Point Generation

Now that the crime density maps are generated, and the crime counts per cell are obtained, the next step is to generate random points for the acquisition of SVI. As the output generated in the previous step is a raster, a grid with a cell size of 250m x 250m is generated and overlayed on the KDE output. The individual cells are the unit of analysis. The road network is required for random point generation and acquisition of SVI. Few roads are not single segments and have multiple line segments in a single road. As the SVI is acquired in directions parallel and perpendicular to roads, this setup affects the calculation of the bearing of the road, which takes the start and end node of the road to calculate the bearing.

To handle the above-mentioned challenge, the roads are split into multiple road segments. For long straight segments, it does not affect the bearing result. The change can be observed for roads that are not appropriately digitized and have one or more deviations in the road. For the obtained new road network, two random points are generated per cell with an average distance of 100m. The distance is kept at 100m to ensure that points generated are well apart from each other and represent the whole cell. However, the points generated are not as expected. There were many points within a very close distance and on the same road but in different cells. As the roads are split, both roads are different road segments but part of the same road. The points within such proximity can confuse the model to learn the features and associate them with the crime count.

So, the road segments intersecting the grid cells are removed, and the remaining road segments inside the cell are combined to form a single object. Then two random points are generated per object inside the cells, and we call them SVI collection points. An illustration is shown in Figure 4.4. This method helped to overcome the challenges stated above. Now the points generated are far apart and not in proximity with points from neighbouring cells. However, only one point is generated in few cells near the study area boundary due to fewer roads in those cells. Even if the points are generated, they are within a very close distance. So, only one SVI collection point represents the crime count of such cells. This situation is mainly observed towards the study area boundary, where the road network is sparse. The random points generated are used for SVI acquisition and label preparation. As the unit of analysis is 250m x 250m cell, the outputs obtained per point are averaged per cell to get the crime count per cell and used in the evaluation.

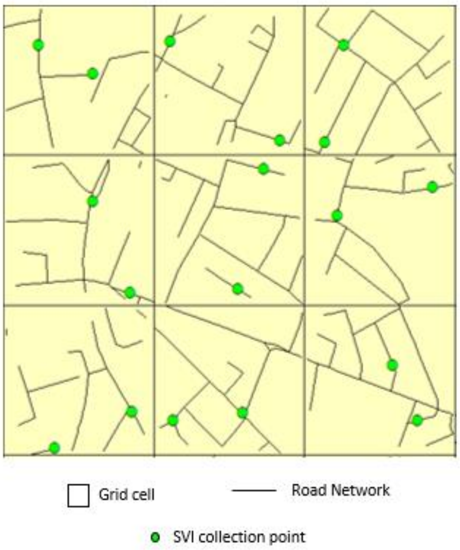


Figure 4.4: The road segments intersecting the grid lines are removed and SVI collection points are generated so that they do not belong to the same road segment. It is also to make sure that two points from different cells are not too close.

4.2.4. SVI Extraction

The random points are generated, and the road network is used to calculate the bearings of the road segments. The bearing value is between 0° and 90°. The heading values used for acquiring images in cardinal directions are 0, 90, 180, and 270. The bearing value obtained is added to these values to account for the deviation in roads. In doing so, the images are acquired in directions parallel and perpendicular to the roads. A simple illustration of SVI acquisition is shown in Figure 4.5. The images acquired tend to capture the context of the built environment information much better than the images acquired in cardinal directions (Fu et al., 2018). As mentioned in chapter 3, FOV of 120 and size of 512x512 are considered as the parameters for downloading SVI.

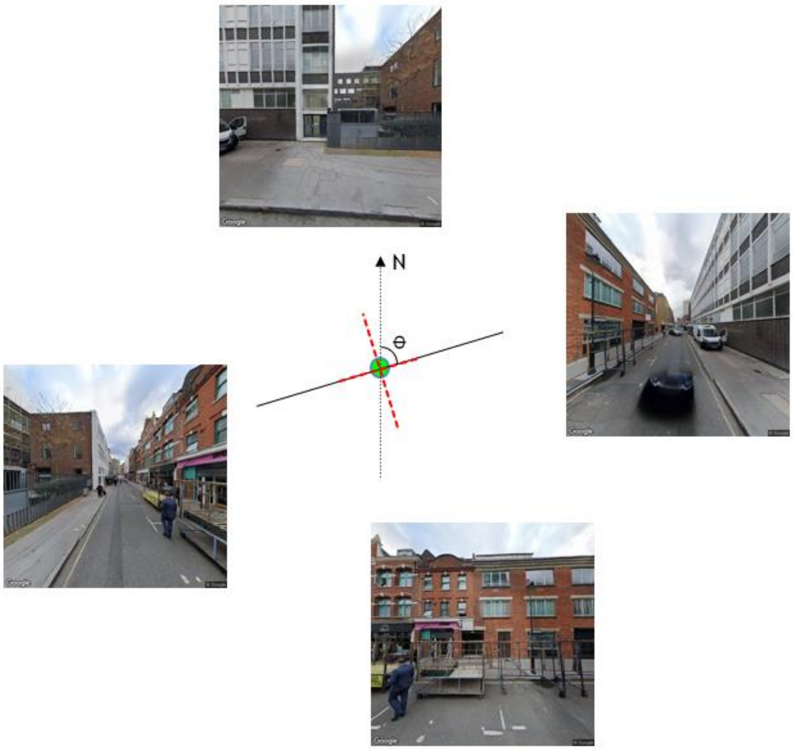


Figure 4.5: The solid black line is the road and red dotted lines are the parallel and perpendicular directions to the road. θ is the bearing of the road with true north. The bearing is calculated and added to the parameters to obtain the SVI in desired direction.

Every random point generated has a unique id, which is used to label the acquired images. As shown in Figure 3.2, the images are labelled by appending a number to the unique id of the SVI collection point. If the image is not available, a blank image is downloaded to a folder. Similarly, few images are captured in the building interiors. These images are removed before preparing the labelled dataset. The blank images are sorted for less size and removed. However, it is challenging to remove the images which are captured in the interiors of the building. As the dataset is huge, the best effort is put to search and remove such images manually. Once the data is cleaned, 148,704 images are corresponding to 37,176 SVI collection points spread across 20,398 cells. The images and the SVI collection points are further used to prepare labelled data set for the CNN model.

4.2.5. Labelled data preparation

The values of the crime density raster obtained by KDE are extracted to SVI collection points using the Extract values to points tool in ArcGIS. First, the values are extracted for individual crime types and are spatially joined to get the values of all crime types in a single table. The attributes of the table include crime counts of all crime types, i.e., burglary, robbery, other thefts and vehicle crimes, the unique id of SVI collection point, and unique id of the uniform grid cell to which the SVI collection points belong. The unique id of the SVI collection point is used as an identifier for the SVI also. The method followed to input the data into the model will be discussed in detail in section 4.3.2. Finally, the population density values from the OA shapefile are spatially joined to respective points. The population density is not generated as a raster. The grid cells are accommodating more than two OAs. So, the implementation of KDE or rasterizing shapefile is resulting in a loss of data. Therefore, the population density values in which the SVI collection point falls is directly joined to the existing table. Now all the data required for the data labelling is in a single table. The explanatory variables are population density value along with SVI, and the target variables are the crime counts of four crimes, i.e., burglary, robbery, other thefts, and vehicle crimes.

The data distribution of the variables for input and output is as shown in Figure 4.6 (a). They are highly skewed and distributed across various orders of values. As the data for different crime types is on different scales, it is a potential problem for the model to learn and adjust weights. The higher values may have much more impact on the model in the training process than the lower values. So, the values of all variables are transformed using logarithmic transformation. After logarithmic transformation, the data distribution is as shown in Figure 4.6(b). The data table obtained will be used to input data to the CNN model. Once the data transformation is performed, the data is ready to be split into training, validation, and testing datasets.

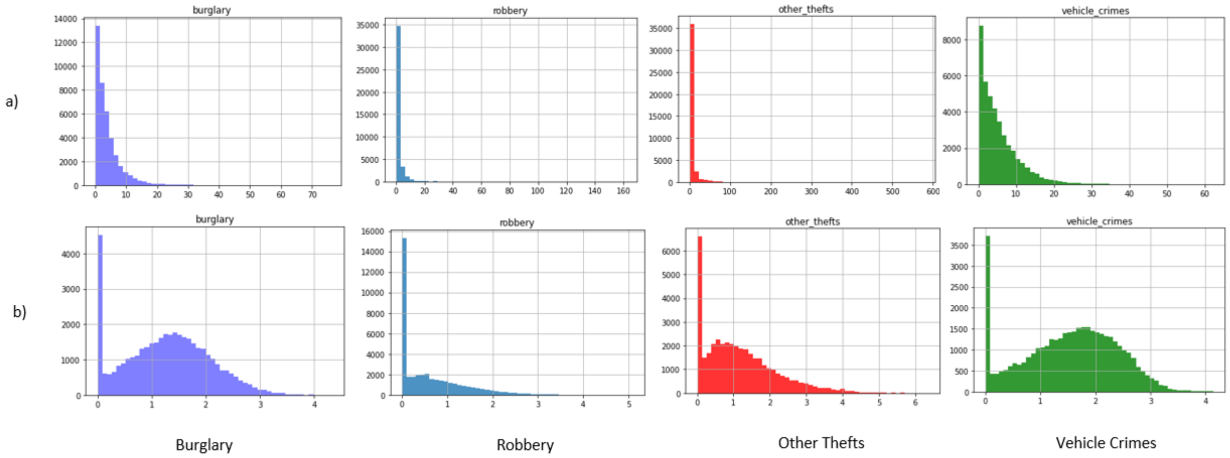


Figure 4.6: a) The frequency distribution of crime rates of selected crime types before logarithmic transformation. The value ranges are different for each crime type. b) The frequency distribution of crime rates after logarithmic distribution. Now the values are considerably in the same range, and this helps in training process.

4.2.6. Training, Validation and Testing data split

Each record corresponds to a SVI collection point in labelled data and is to be split into training, validation, and testing datasets. The training and validation sets are used for training and parameter tuning, while the testing set is used to evaluate the model. Initially, a uniform grid with a cell size larger than the unit of analysis is considered for splitting the data. The alternate grid cells are considered for training and testing split in a checkered manner. One set is used for training, and the other is used for validation and testing by random selection. In this type of data selection, there is a risk that the neighbourhoods that fall in the training set may be completely different from the validation and testing split as the area is large. This may affect model performance and generalisation.

Hence, the uniform grid which is overlayed on the KDE map is used as a basis for splitting the data. The points for the data split are taken so that two points from the same cell fall into the same split. All the SVI collection points are grouped by the unique id of the cell, and then the data records are split between the ratio of 60, 20 and 20 for training, validation, and testing datasets, respectively, by random selection. Table 4.1 below shows the number of points and number of images per split. This is the final step in the data preparation, and the following sections discuss the model design and model training.

Dataset	**SVI collection points**	**SVI**
Training	22,278	89,112
Validation	7,424	29,696
Testing	7,474	29,896
Total	37,176	148,704

Table 4.1: Dataset split for training, validation, and testing

4.3. Model design and training

4.3.1. Configuring CNN model

The ongoing research in DL helped build great CNN architectures for feature extraction from images used for different tasks like image classification, image regression, semantic segmentation, image recognition, etc. Architectures like AlexNet (Krizhevsky et al., 2012) and VGG-Net (Simonyan & Zisserman, 2015) are considered legacy architectures and the basis for all modern architectures. The modern architectures are deep compared to legacy architectures and have proven to perform better than them with a reduced number of parameters. ResNet (He et al., 2015) is one such architecture that is a deep residual network trained on the Imagenet dataset. It uses skip connections to handle the vanishing gradient problem. The example of skip connections between the ResNet layers is shown in Figure 4.7.

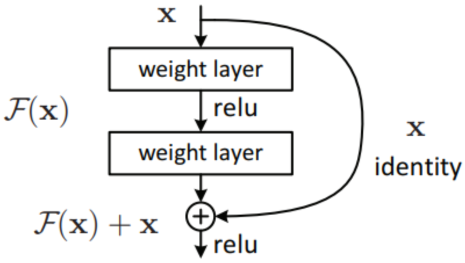

x
weight layer
$\mathcal{F}(\mathbf{x})$
relu
weight layer
x
identity
$\mathcal{F}(\mathbf{x}) + \mathbf{x}$
relu

Figure 4.7: Residual block and usage of skip connections in ResNet (He et al., 2015)

The residuals from the previous blocks are added to the current block output to address the vanishing gradient problem. However, repeated convolutions if the dimensions of the output are reduced in the current layers, the same convolutions are applied to the residuals before adding to the current output to match the dimensions. In Figure 4.6, the identity block is replaced by a convolution and added to the further layers. Though the network is deep compared to the legacy architectures, it has fewer parameters and is easy to optimize. The ResNet18 architecture is chosen to build the 4-CNN model to handle this research's muti-output image regression problem.

ResNet18 belongs to the ResNet family and, as the name suggests, is 18 layers deep. It is much shallower to than the other ResNet family variants like ResNet50 and ResNet101, which are most used. The original ResNet18 trained on the Imagenet dataset has around 11 million parameters, much less than the legacy architectures. Though the network is shallow, it leverages the skip connections to perform better. The 4-CNN model built in our research work is inspired by Dubey et al. (2016) and Andersson et al. (2017), where the authors used the model to input 4 SVI simultaneously to perform the ranking and classification tasks, respectively. A pre-trained model of Resnet18 on the Places365 dataset (Zhou et al., 2018) is configured to build the model's CNN blocks. Before the output layer Global Average Pooling is performed to produce a vector of length 512, which is then fully connected to the output layer with 365 neurons.

The convolutional backbones of the ResNet18 pre-trained on the places365 dataset are taken as individual blocks of 4-CNN. The output of each block is a feature vector of length 512. The four images corresponding to a point are passed through one CNN block each simultaneously, and the obtained outputs are concatenated to form a feature vector of length 2048. The corresponding population density is also concatenated to the current output. The final feature vector of length 2049 is fully connected to the output layer with four neurons, where each neuron is corresponding to a crime type. The linear activation is used in the final layer as the problem is regression. The final model designed is shown in Figure 4.8, which takes five inputs and gives four outputs.

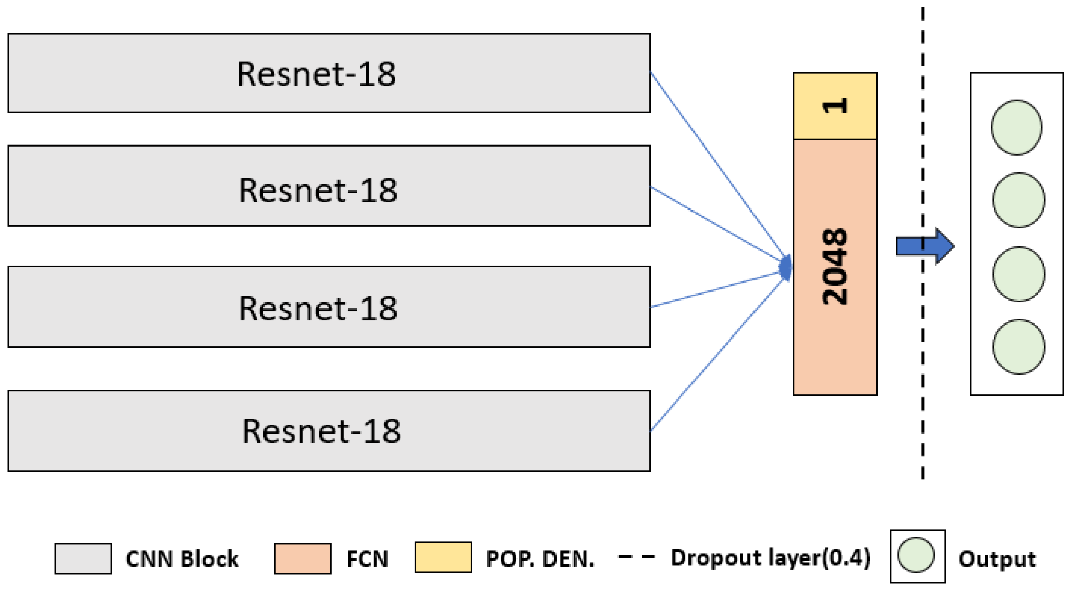


Figure 4.8: 4-CNN model implemented using ResNet18 convolutional backbones. The designed model takes 5 inputs including population density and gives crime counts as outputs for four crime types: burglary, robbery, other thefts, and vehicle crimes.

The pre-trained model with places365 weights is available in the Pytorch framework. The ResNet18 is transformed from Pytorch to Keras, and the rest of the model is built using Keras functional API. As the problem is multi-output regression, Mean Square Error (MSE) is used as a loss function. MSE is the average of squared differences between the actual values and the predicted values. The formula for MSE is

$$MSE = \frac{\sum_{i=1}^{n}(y_i - y_i')^2}{n}$$

Where n is the total number of observations, y_i is the actual value and y_i' is the predicted value. For a single output regression model, the MSE is calculated directly from the actual and predicted values. However, in the case of multi-output regression, the MSE value is calculated for individual outputs and then averaged to get the single value used for optimization. So, the MSE, considered for optimization, is the average MSE value of all the outputs. The adam optimizer is used as the optimizer for the model.

4.3.2. Model training

The input data table is to be processed to load input to the model. The data generators are already present for processing the tables to prepare the data and input the model. However, they are designed to handle single input and single data types, i.e., image or numeric/categorical data. They generally take the image identifier, preprocess the image as the model requires, and prepare the input. Similarly, take the columns with outputs and prepare output for training. However, in our model, there are multiple inputs of different data types and multiple outputs. Therefore, a custom data generator is built to prepare the inputs and outputs for the model. The custom data generator takes the inputs and outputs columns from the table prepared in the labelled data preparation step to prepare the data for input. As mentioned earlier, the unique id of the SVI collection point is used to fetch all the four images and the population density from the table and prepare a tensor to input to the model.

Similarly, the columns with output values are used to prepare the output data. As the dataset is huge, mini-batches of data are used for training the model. The batch size is 8 for training the model, which implies that eight records in the labelled dataset table are processed. The custom data generator generates 32 images, their corresponding population densities, and outputs. If larger batch size is taken, the system requirements are insufficient to carry out the training process. The training, validation and testing datasets are processed in the same way. The training and validation datasets are used to train and finetune the model, where the testing dataset is used to evaluate the model.

As we know that the weights are initialised from the ResNet18 network pre-trained on the Places365 dataset, finetuning of the model is done by freezing the top half of the original network weights. The dataset used in our work is not so huge, and training the model from scratch is not viable. Hence, transfer learning is used to leverage the network trained on the places365 dataset, a scene-centric database. The experimentation is carried out by changing the trainable parameters. Different models are trained by changing the combinations of inputs and outputs.

The initial learning rate is set to 0.0001 and is decreased by order of 0.1 for every five epochs. Early stopping is used to stop the model if the loss flattens. The validation loss is monitored to stop the model and to avoid overfitting. As the model is pre-trained and the dataset is similar to the original dataset, the model reaches minimum validation loss in fewer epochs. However, the model is run for 30 epochs without early stopping, and the weights are saved for every epoch. Once the model is trained, the best weights with minimum validation loss are considered for evaluation of the model. Now that the model is trained and ready, the next step is the evaluation of the model.

4.4. Model Evaluation

Model evaluation is to evaluate the built model's performance and estimate how better the model can generalize on unseen data. The testing dataset is used to evaluate the model. As the model is a regression model, the following statistics are used to evaluate the model.

1. Mean Squared Error (MSE)

Mean Squared Error (MSE) is the average squared differences (errors) between actual and predicted values. It is sensitive to outliers and always positive as the differences are squared. The formula for MSE is given by

$$MSE = \frac{\sum_{i=1}^{n}(y_i - y_i')^2}{n}$$

Where n is the total number of observations, y_i is the actual value and y_i' is the predicted value.

2. Root Mean Squared Error (RMSE)

Root Mean Squared Error (RMSE) is the standard deviation of the errors in predicted values. It is obtained by finding the square root of MSE. The formula for RMSE is given by

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - y_i')^2}{n}}$$

3. Mean Absolute Error (MAE)

Mean Absolute Error (MAE) is the measure of absolute differences in the actual and predicted values. It does not consider the direction of the error and represents how close the predicted value is to the actual value. The formula for MAE is given by

$$MAE = \frac{\sum_{i=1}^{n}|y_i - y_i'|}{n}$$

4. R-squared

R-squared (R^2) also known as the coefficient of determination is a statistical measure that explain the proportion of variance in the target variable explained by the explanatory variables. The value of R^2 ranges between 0 to 1 (0%-100%). The higher the value better the model predicts the target variable. The formula for R^2 is given by

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - y_i')^2}{\sum_{i=1}^{n}(y_i - \overline{y})^2}$$

Where $\overline{y}$ is the mean of the all observations.

5. RESULTS AND DISCUSSION

This chapter explains the results obtained from data preparation and the evaluation of the 4-CNN model built. Section 5.1 shows the results obtained throughout the research. Performance and predictions of the model are discussed in section 5.2. section 5.3 discusses the results of the research. Finally, the limitation of the work is mentioned in section 5.4.

5.1. Results of data preparation

5.1.1. Results of KDE

The crime data points of four crime types: burglary, robbery, other thefts, vehicle crimes are modelled using KDE, and the respective crime density maps are created with a cell size of 250m x 250m. The pixel values are multiplied by the cell area to obtain the crime count in each cell. The estimated crime type crime count for a particular cell is the label for the SVI extracted from the corresponding cell. The following figures are the crime rate maps of the four crimes and the distribution of their frequency.

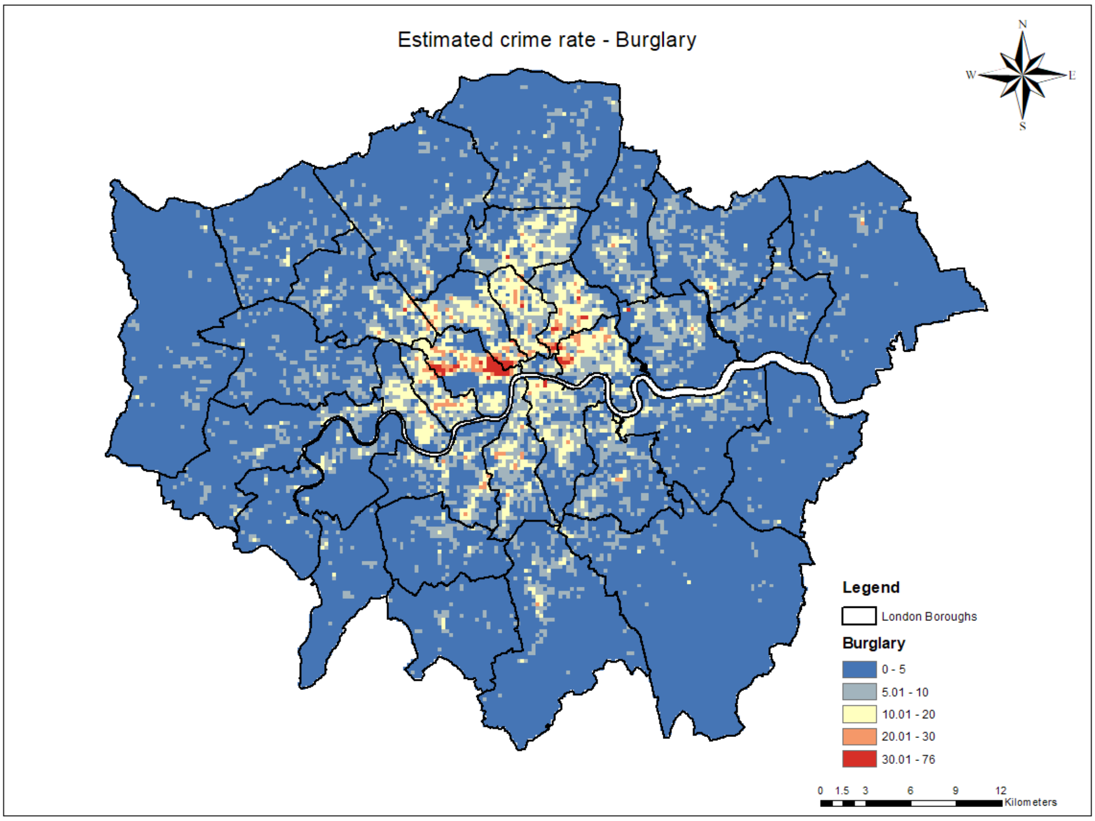


Figure 5.1: Burglary crime rate map

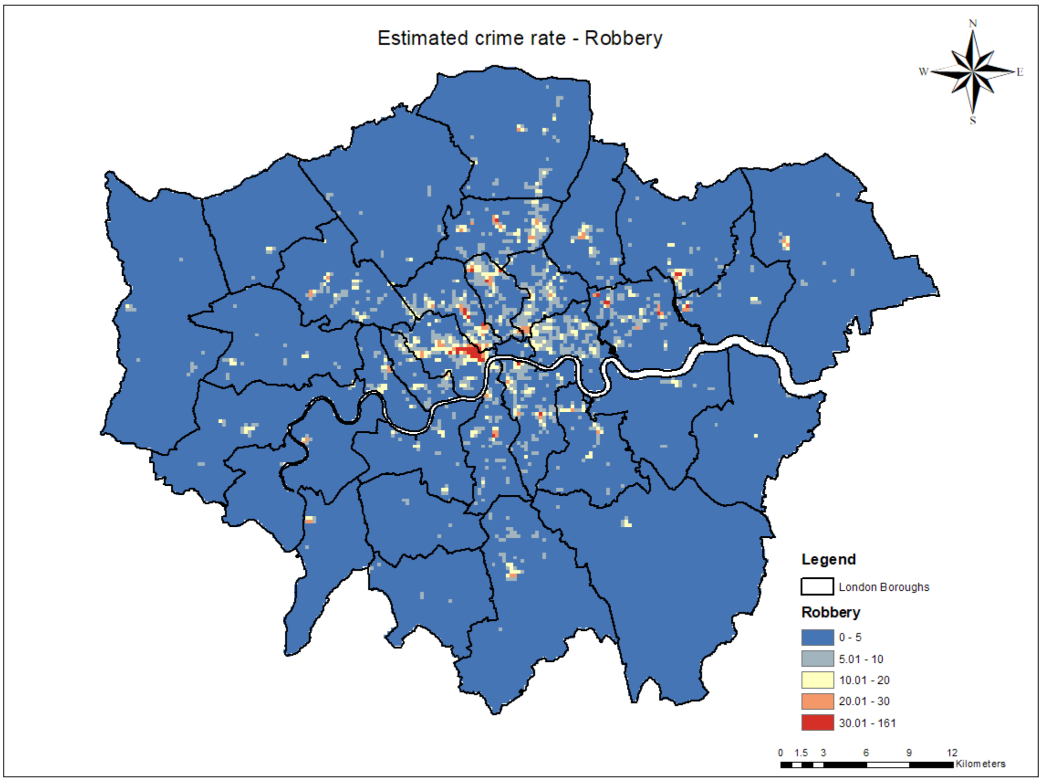


Figure 5.2: Robbery crime rate map

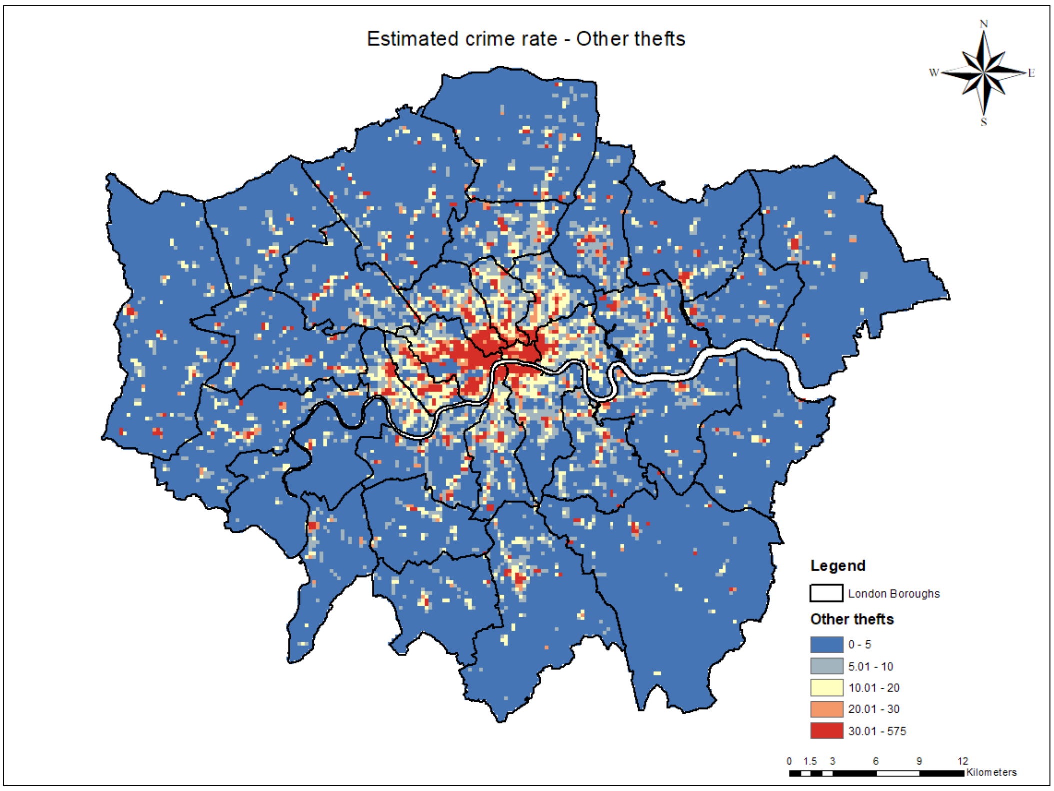


Figure 5.3: Other thefts crime rate map

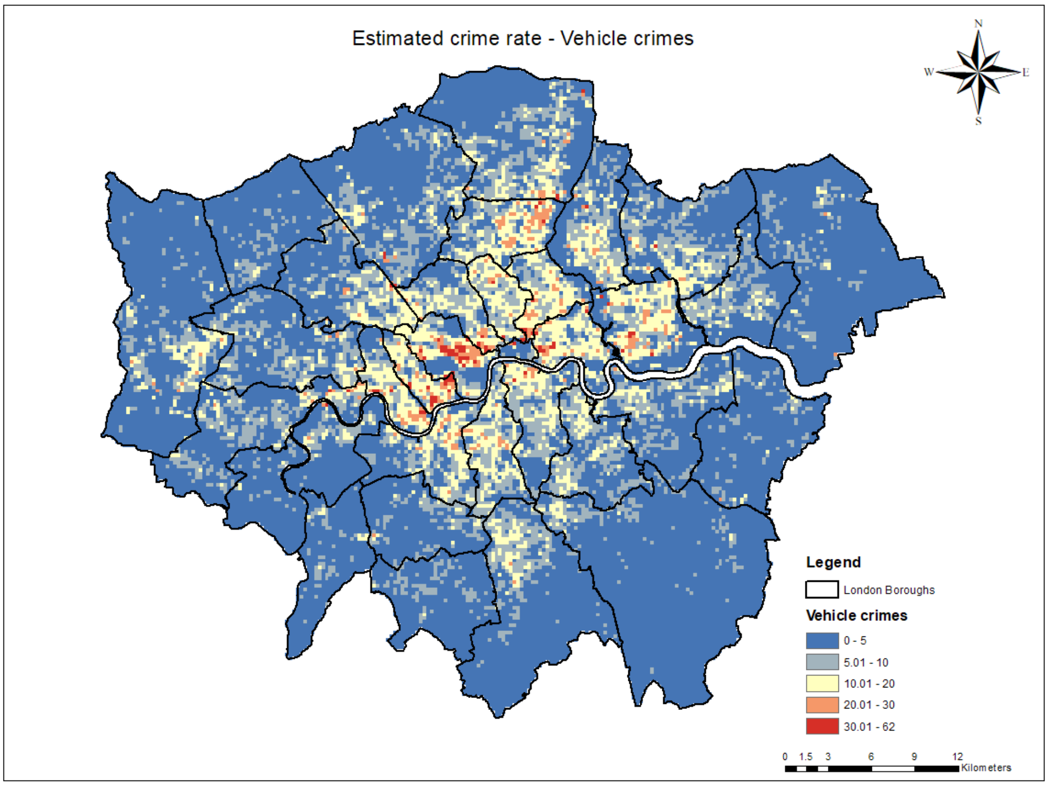
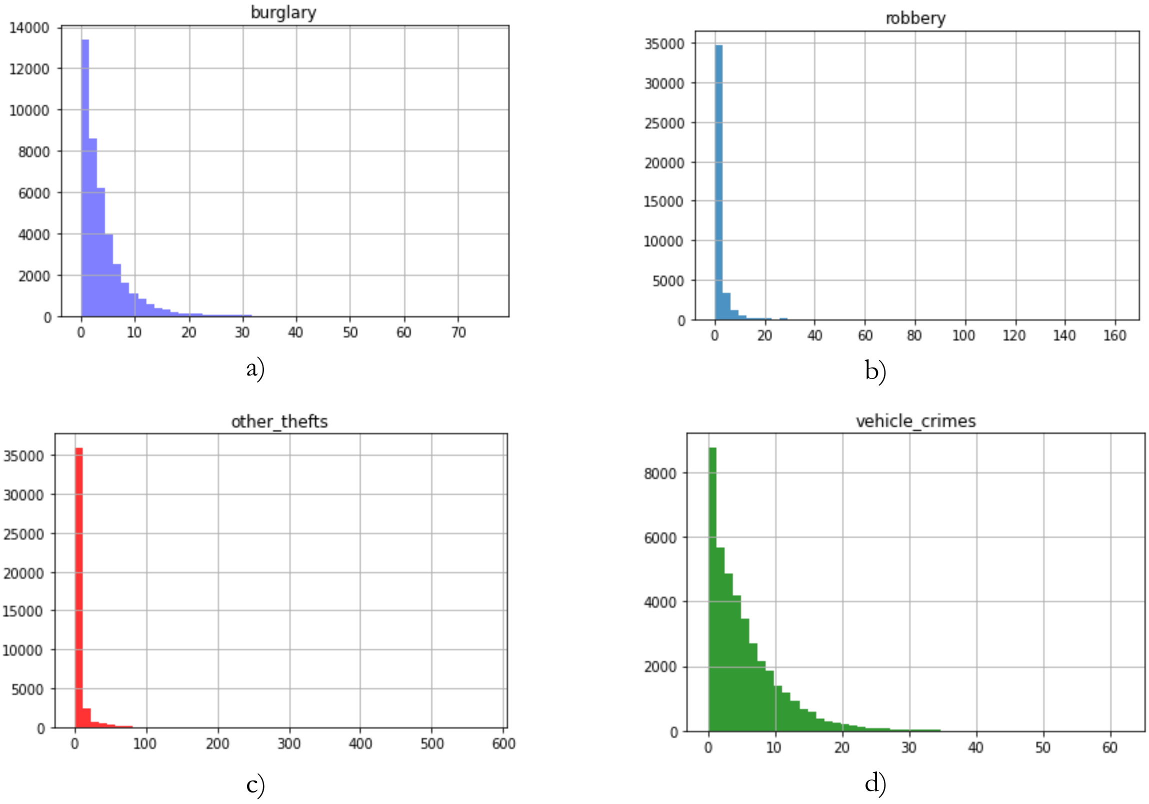


Figure 5.5: Vehicle crimes crime rate map

The frequency distribution of crime rates of different crime types is as shown in Figure 5.4 below



a)

b)

c)

d)

Figure 5.4: Frequency distribution of crime rates of a) burglary b) robbery c) other thefts and d) vehicle crimes

For visualization purposes, crime rates of each crime type are divided into five class intervals. The range of crime rates for each crime type is not in the same value range. It can be noticed that the distribution of every crime type is highly skewed and most of the values are zero. As mentioned in chapter 4, the cell size of KDE (250m x 250m) is considered as the unit of analysis and further analysis is carried out. The values of each cell are assigned as labels to SVI extracted from the individual cell, and the labelled data is prepared.

5.1.2. Training, validation, and testing data split

After extraction of SVI, the already prepared labelled data table is cleaned by removing the records for which the SVI images are not available. The labelled data table has the unique cell id, unique SVI collection point id, population density and the values of four crime types. The SVI collection point id is assigned as an image identifier. Once the data is cleaned, the data is split into training, validation, and testing based on the unique cell id by random selection. The following image shows the data split setup mapped on the study area.

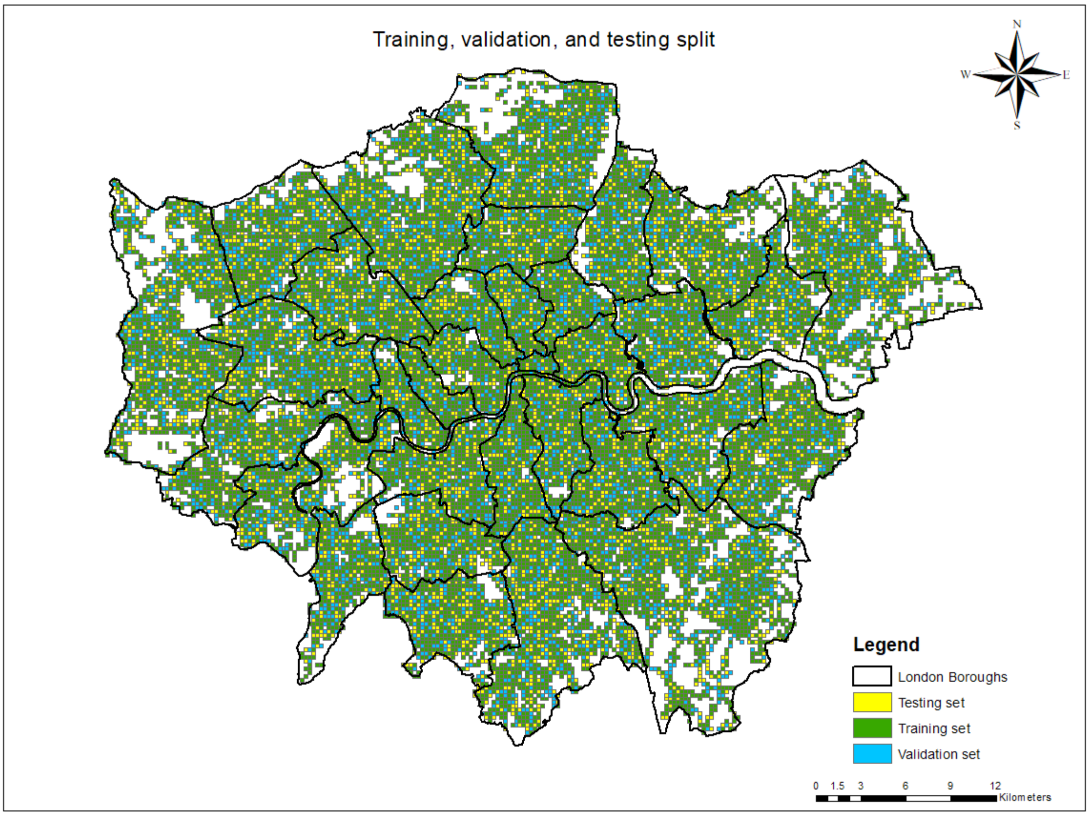


Figure 5.6: Map showing training, validation and testing cells split

As shown in Figure 5.6, SVI is not available in few parts of the study area. After removing them, the remaining cells are considered to split the data into training (12239 cells), validation (4080 cells), and testing (4079) by random selection. This step completes the data preparation part, and the labelled data is ready to input into the model.

5.2. Results of CNN model

4-CNN model is designed which takes multiple inputs and give multiple outputs. The model is initialized with Places365 weights, which is trained on a scene-centric image database. Half of the weights are frozen, and the bottom half of the weights are learned by the model. Three different models with different input and output combinations are trained, and the results are compiled:

i) SVI as input and crime rates of selected crime types as output (multi-output)
ii) SVI and population density as inputs and crime rates as output (multi-output)
iii) SVI and population density as input and total crimes as output (single-output)

5.2.1. Model 1: SVI as input and crime rates as output

The custom data generator which feeds data to the model for training is configured to take four inputs and four outputs. As mentioned above, the upper half of the weights are frozen in the model. MSE is used as a loss function to be optimized, and adam optimizer is used to adjust the weights. The model is trained for 30 epochs, and the weights are saved after each epoch. The training set is used for training, and the validation set is used to tune the weights based on the loss value. The output is obtained per 4 images, i.e., one SVI collection point. As the unit of analysis is a cell with two SVI collection points, the results are averaged later to estimate the output. The evaluation metrics used are MSE, RMSE, MAE and R^2. The metrics are tabulated per SVI collection point and per cell for both training and testing sets. All the plots and error metrics use the logarithmic transformed data.

	Burglary	**Robbery**	**Other thefts**	**Vehicle Crimes**
MAE	0.45	0.41	0.54	0.49
MSE	0.33	0.31	0.52	0.39
RMSE	0.57	0.55	0.72	0.62
R^2	0.44	0.39	0.46	0.41

Table 5.1: Evaluation metrics of the testing set for Model 1. Metrics calculated are per SVI collection point (4 images)

	Burglary	**Robbery**	**Other thefts**	**Vehicle Crimes**
MAE	0.37	0.33	0.42	0.40
MSE	0.21	0.18	0.31	0.26
RMSE	0.46	0.43	0.55	0.51
R^2	0.64	0.63	0.69	0.61

Table 5.2: Evaluation metrics of the training set for Model 1. Metrics calculated are per SVI collection point (4 images)

The scatterplots of the actual and predicted values for training and testing sets are as shown in Figure 5.7.

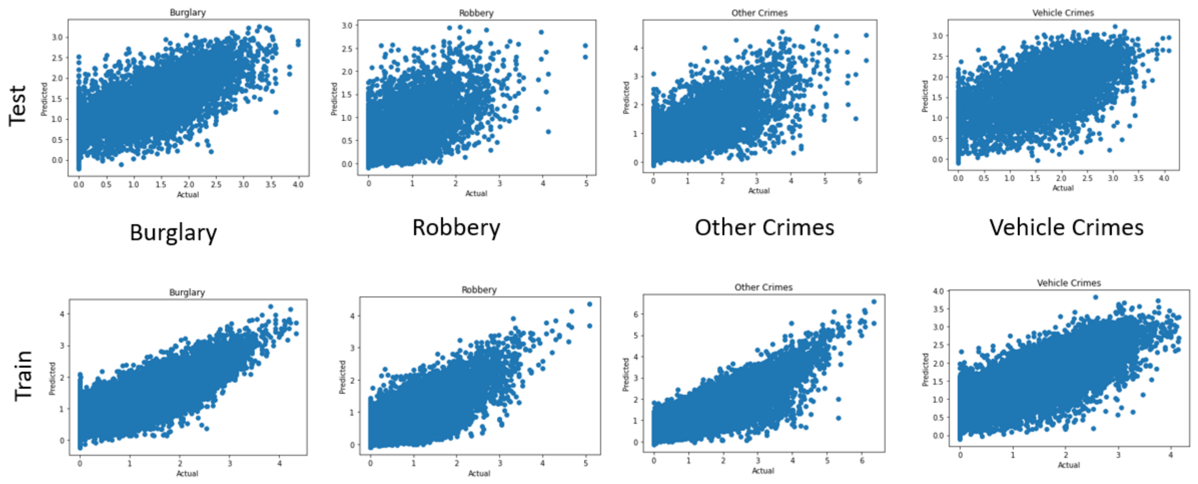


Figure 5.7: Scatterplots of actual vs. predicted values (4 images) for Model 1.

As the unit of analysis is a unit cell, the outputs are grouped and averaged per cell to check the performance of the model. The evaluation metrics for testing and training sets are calculated and compiled in Table 5.3 and Table 5.4.

	Burglary	Robbery	Other thefts	Vehicle Crimes
MAE	0.44	0.39	0.50	0.47
MSE	0.30	0.27	0.45	0.36
RMSE	0.55	0.52	0.67	0.60
R^2	0.51	0.45	0.53	0.49

Table 5.3: Evaluation metrics of the testing set for Model 1. Metrics calculated are for single-cell (8 images)

	Burglary	Robbery	Other thefts	Vehicle Crimes
MAE	0.35	0.31	0.39	0.39
MSE	0.20	0.16	0.26	0.24
RMSE	0.44	0.40	0.51	0.49
R^2	0.68	0.67	0.73	0.66

Table 5.4: Evaluation metrics of the training set for Model 1. Metrics calculated are for single-cell (8 images)

The scatterplots of the actual and predicted values of the modified outputs for testing and training sets are as shown in Figure 5.8.

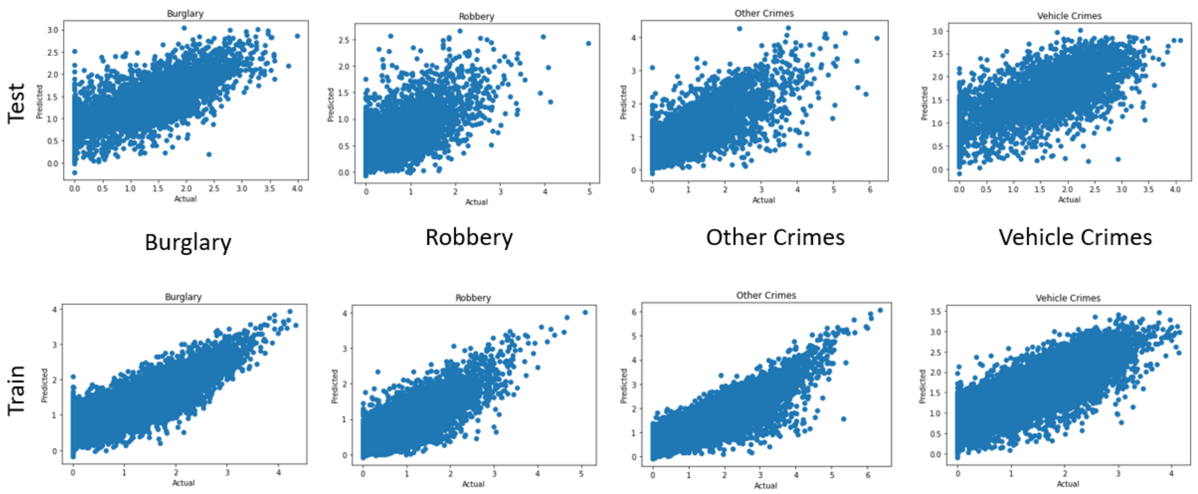


Figure 5.8: Scatterplots of actual vs. predicted values (8 images) for Model 1.

5.2.2. Model 2: SVI & population density as input and crime rates as output

In addition to SVI, population density is also added as an input to observe the effect of population density on model prediction. So, the data generator for the model is configured to take five inputs and four outputs. The rest of the configuration of the model is the same as Model 1. The evaluation metrics are calculated the same as mentioned in section 5.2.1. The results are segregated for training, testing sets and per point, per cell outputs. The results did not improve significantly, even with the inclusion of population density as an additional variable. However, the results were slightly better when the predictions were averaged per cell. The metrics calculated per SVI collection point are shown in Table 5.5 and Table 5.6.

	Burglary	Robbery	Other thefts	Vehicle Crimes
MAE	0.45	0.41	0.55	0.49
MSE	0.33	0.31	0.54	0.39
RMSE	0.57	0.55	0.74	0.62
R^2	0.44	0.38	0.44	0.41

Table 5.5: Evaluation metrics of the testing set for Model 2. Metrics calculated are per SVI collection point (4 images)

	Burglary	Robbery	Other thefts	Vehicle Crimes
MAE	0.35	0.31	0.39	0.38
MSE	0.19	0.17	0.28	0.23
RMSE	0.44	0.41	0.52	0.48
R^2	0.67	0.66	0.72	0.65

Table 5.6 Evaluation metrics of the training set for Model 2. Metrics calculated are per SVI collection point (4 images)

The scatterplots of actual versus predicted crime rates of training and testing sets are as shown in Figure 5.9.

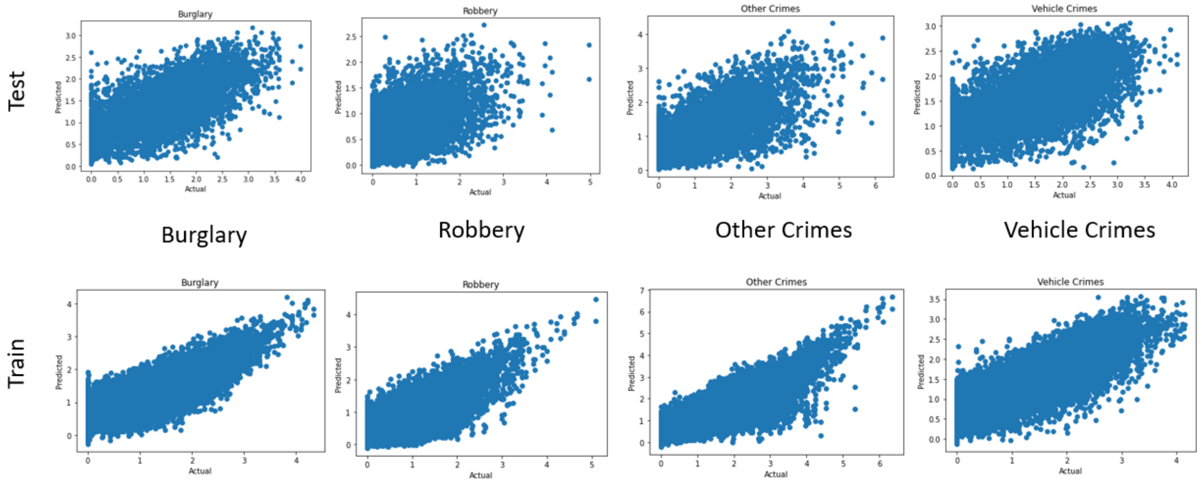


Figure 5.9: Scatterplots of actual vs. predicted values (4 images) for Model 2.

The evaluation metrics of the outputs modified per cell for testing and training datasets are as shown in Table 5.7. and Table 5.8.

	Burglary	Robbery	Other thefts	Vehicle Crimes
MAE	0.44	0.39	0.51	0.47
MSE	0.30	0.27	0.48	0.36
RMSE	0.55	0.52	0.69	0.60
R²	0.51	0.44	0.50	0.49

Table 5.7: Evaluation metrics of the testing set for Model 2. Metrics calculated are per single cell (8 images)

	Burglary	Robbery	Other thefts	Vehicle Crimes
MAE	0.34	0.29	0.37	0.37
MSE	0.18	0.15	0.24	0.22
RMSE	0.42	0.39	0.49	0.47
R²	0.71	0.69	0.76	0.69

Table 5.8: Evaluation metrics of the training set for Model 2. Metrics calculated are per single cell (8 images)

The scatterplots of actual vs predicted crime rates for the modified outputs is shown in Figure 5.10.

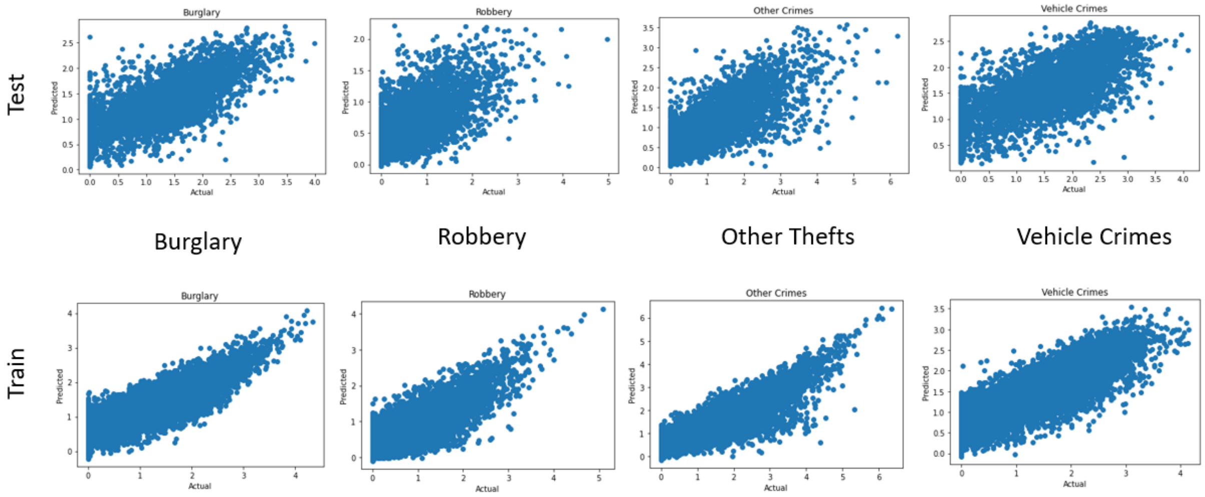


Figure 5.10: Scatterplots of actual vs. predicted values (8 images) for Model 2.

5.2.3. Model 3: SVI & population density as input and total crime rate as output

In this model, instead of individual crime rate per crime type, crime rates are summed up, and the total crime rate is considered output. The model is trained to solve the single-output regression problem. The custom data generator is modified to handle the inputs and output. The rest of the configuration is the same as Model 2. In this model, evaluation metrics are calculated for a single output. Like the previous model, one set of evaluation metrics is for SVI collection point output, and one is for the cell output. The R-squared values are higher than the average R-squared of all crime types obtained in model 1 and model

2. However, the other metrics are relatively lower than that of previous models. The R-squared value significantly improved when the predictions are averaged per cell. The evaluation metrics for point output are shown in Table 5.9 and Table 5.10.

	Total Crimes
MAE	0.56
MSE	0.55
RMSE	0.74
R^2	0.50

Table 5.9: Evaluation metrics of the testing set for Model 3. Metrics calculated are per SVI collection point (4 images)

	Total Crimes
MAE	0.52
MSE	0.49
RMSE	0.70
R^2	0.59

Table 5.10: Evaluation metrics of the training set for Model 3. Metrics calculated are per SVI collection point (4 images)

The evaluation metrics per single cell for testing and training sets are tabulated and shown in Table 5.11 and Table 5.12.

	Total Crimes
MAE	0.35
MSE	0.21
RMSE	0.46
R^2	0.81

Table 5.11: Evaluation metrics of the testing set for Model 3. Metrics calculated are per single cell (8 images)

	Total Crimes
MAE	0.32
MSE	0.18
RMSE	0.42
R^2	0.85

Table 5.12: Evaluation metrics of the testing set for Model 3. Metrics calculated are per single cell (8 images)

The scatterplots of actual versus predicted total crime rates per SVI collection point and per single cell for training and testing sets are shown in Figure 5.11.

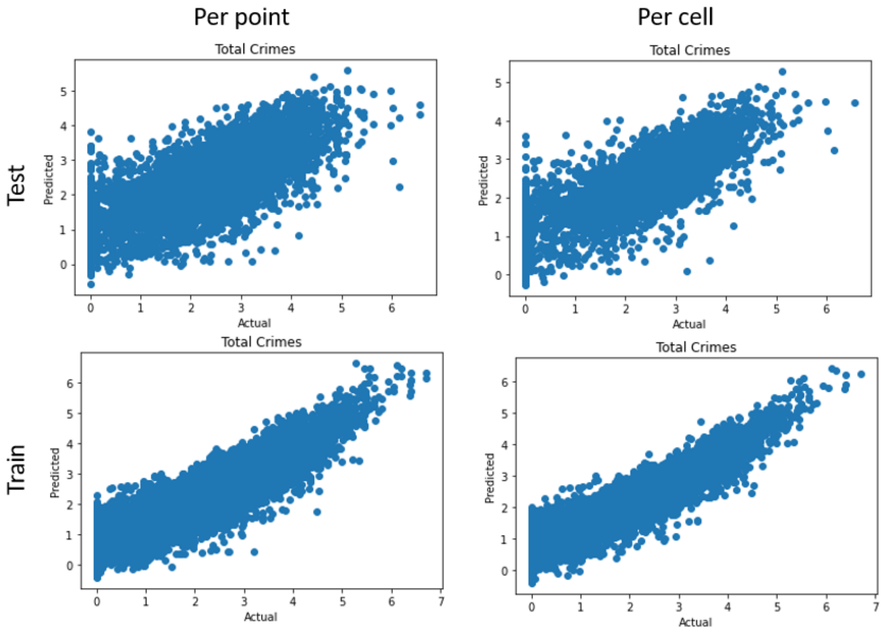


Figure 5.11: Scatterplots of actual vs. predicted total crime rates.

5.2.4. Prediction results

The training, testing and validation set predictions are taken from model 2 and are mapped to compare against the original labels. Inverse logarithmic transformation is applied to predicted values, and the raster is generated based on the unique cell id. To compare the output with the initial KDE map, the same intervals are chosen to show the intensity of the crime. The maximum value and the number of crimes predicted changed, but the hotspots did not change compared to the KDE output. The results are shown in the figures below.

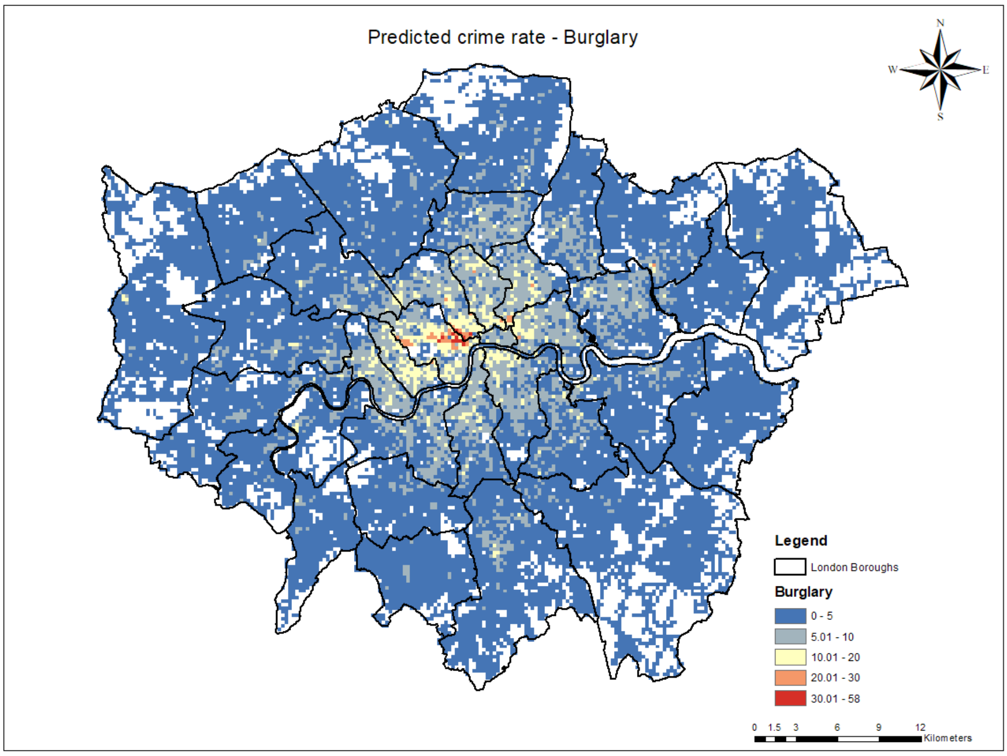


Figure 5.13: Predicted crime rate - Burglary

.

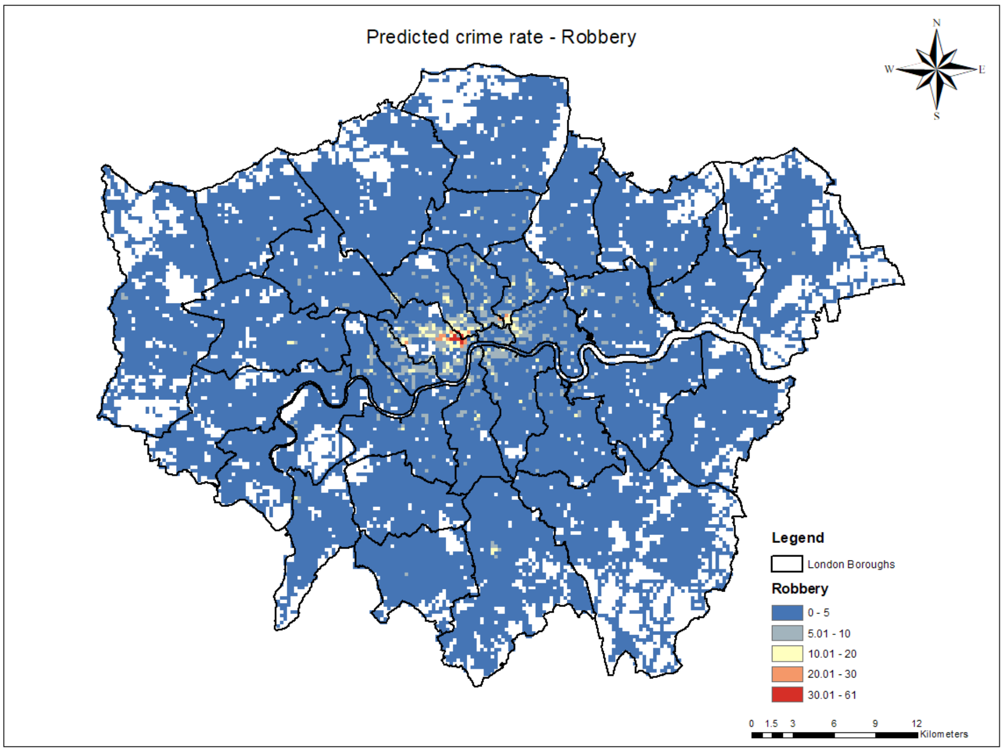


Figure 5.12: Predicted crime rates - Robbery

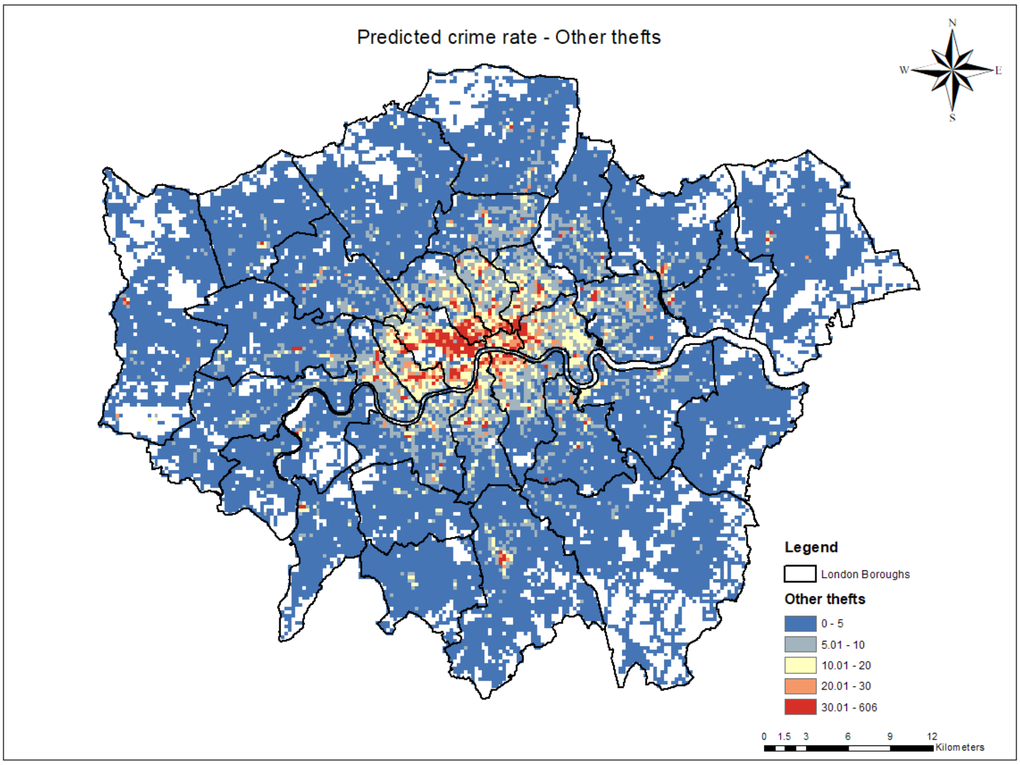


Figure 5.14: Predicted crime rate – Other thefts

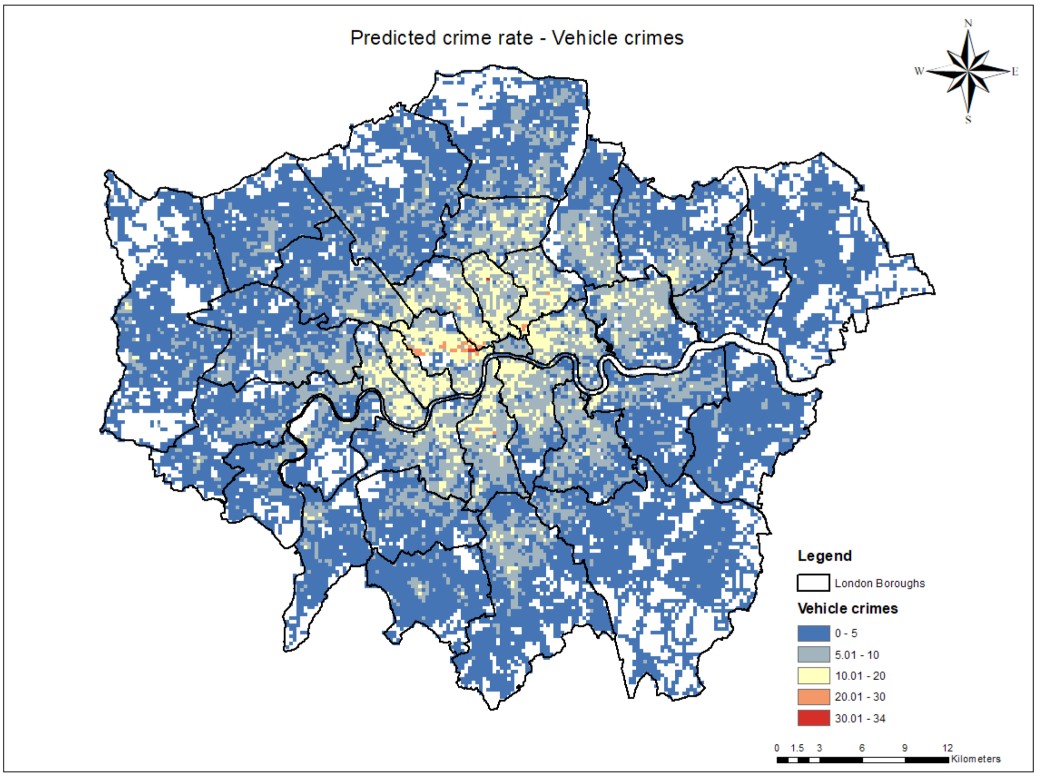


Figure 5.15: Predicted crime rate – Vehicle crimes

5.3. Discussion

This section briefly reflects on the results and evaluates the methodology followed to achieve the main objective of building a model to predict the crime rate from the visual variables by solving a regression problem.

The maps generated by KDE show the distribution of crime in different parts of the study area. In almost all maps, the crime is majorly concentrated in the centre, and as we move towards the boundary, the crime occurrence is very low. The output of KDE is the key in preparing the labelled data for the model. In almost; all crime types, the higher frequency of cells with zero value skewed the distribution. In general, if the distribution is normal for regression tasks, the model learns better and performs well.

Nevertheless, in this case, the values cannot be removed as they have information about the areas with fewer crimes. The neighbourhoods in such a large study area differ considerably, and the elimination of few areas may affect the model training process. The chosen hyperparameters are a grid cell size of 250m x 250m and a bandwidth of 275m to consider the neighbouring cells. A smaller cell size is chosen because aggregation can be done at a later stage, but the segregation will be difficult. However, the bandwidth choice may be the reason for such data distribution. If the bandwidth is large, then the search area is larger, and apparently, more crimes would have been considered to estimate the density of crime. A choice of larger bandwidth may have smoothened the crime density map affecting the data distribution. This may influence the model training and performance considerably.

The 4-CNN model designed is inspired by previous works. This has a major influence on the model training process. If a model with a single CNN is used for training, the 4 street view images corresponding to a single point are considered as individual inputs. Using the 4-CNN model, the model gets the holistic view of the environment, which affects the model's performance. If individual images are given as input, then the outputs need to be averaged twice to predict for the unit of analysis. Now the model takes five inputs at once and learns the crime rate from all the variables provided. There are advantages and disadvantages to both methods. If a single image is given as an input, we can infer the features in the image that influence the crime rate, but by averaging multiple times, the information is lost. It is difficult for four images to interpret which image or visual variables affect the crime, but the advantage is that the model sees the whole scene and learn from the data. However, there are methods to see which image has the dominant features affecting the predictions.

The results of the model evaluation are encouraging. Model 1 with four images as input has performed competitively with model 2. For the metrics calculated per point, there is hardly any difference between the metric values. Even the R-squared value, which explains the variance in the target variables, is similar. However, the evaluation metrics are slightly better when calculated per cell. This may be due to the inclusion of the population density variable in model 2. The extra information of population density might have caused a slight variation in the result.

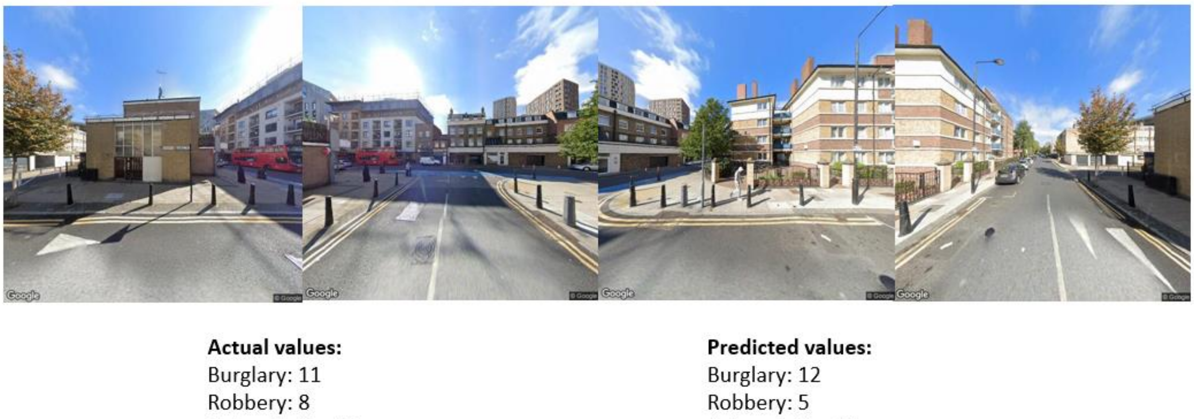
The models tend to give better predictions with training sets as the models already sees them. However, in model 1 and model 2, we can see a significant change in the error metrics. From this, it can be inferred that either the information provided for the model is insufficient to make the predictions or the model is overfitting on training datasets. Even though the results seem to be considerate, model 1 and model 2 are not generalizing for the unseen data. The possible reasons are insufficient training data, insufficient information, or better data preparation. It can be either of them or all of them. Insufficient data can be handled by either extracting more images from the same cell, i.e., more SVI collection points or by data augmentation. For insufficient information, more relevant variables can be added as explanatory variables. As mentioned above, better bandwidth can be chosen to model the crime data for better data preparation.

The model 3 evaluation metrics are interesting. In this model, all the crimes are summed up and considered a single output to be predicted. The R-squared value of model 3 calculated on the testing set per point and cell is 0.50 and 0.81, respectively. There is a remarkable improvement in the explained variance of crime rate by SVI, unlike model 1 and model 2. However, there is not much difference in the training and testing set evaluation metrics as observed in model 1 and model 2. It is safe to say that model 3 generalizes well compared to model 1 and model 2. However, this can be the case because all the crime rates are summed up and considered a single output.

The scatterplots of actual versus predicted tell us about how better the model performed. It tells about the deviation of predicted values from the actual values along with the direction of deviation. In an ideal scenario, all the data points of the plot should fall on a straight line which makes 45° with the X-axis. However, usually, that is not the case as the model has few limitations. It can be observed from the scatterplots that the lower values are overestimated, and the higher values are underestimated. A bulge can be observed near the origin, and as the value gets higher, the clustering narrows down even though underestimated. Insufficient information can be a potential reason for this behaviour.

The crime rate predictions of burglary are consistently better than the other crime types. The scatterplots of burglary are consistent and follow the straight line in both model 1 and model 2 even though there is an estimation problem as stated. The scatterplots of robbery and other thefts are clustered, and the dispersion is high compared to burglary and vehicle crimes. In the vehicle crimes scatterplot, the dispersion of values from the straight line is almost the same for lower and higher values. However, as expected, the scatter plots of training sets for model 1 and model 2 are better than the testing sets. Model 3 is much better in terms of estimation problem compared to model 1 and model 2. A definite pattern following a straight line can be observed, and the deviation from the actual value is less in comparison with model 1 and model 2. The maps generated from the predictions of model 2 show the underestimation and overestimation clearly. Even if the hotspot pattern is the same, the crime density is different from the original KDE outputs.

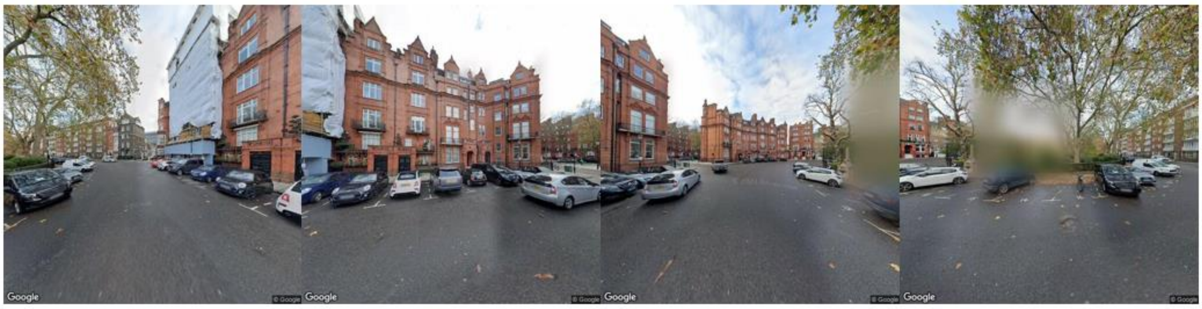
Two sets of images, their actual values, and predictions, are shown in Figure 5.16 and Figure 5.17.



Actual values:
Burglary: 11
Robbery: 8
Other thefts: 12
Vehicle Crimes: 15

Predicted values:
Burglary: 12
Robbery: 5
Other thefts: 16
Vehicle Crimes: 15

Figure 5.16: Actual and predicted values for a set of images. The predicted values are closer to the actual values.



Actual values:
Burglary: 19
Robbery: 12
Other thefts: 18
Vehicle Crimes: 58

Predicted values:
Burglary: 7
Robbery: 3
Other thefts: 10
Vehicle Crimes: 9

Figure 5.17: Actual and predicted values for street view images. There is a huge difference between actual and predicted values.

Comparing the above figures, the built environment in Figure 5.16 is comparatively new. It seems well organized with pavements and fewer vehicles parked on the street. In Figure 5.17, it can be observed that most of the vehicles are parked outside, and the construction of a building is in progress. This can be a possible factor for a high number of vehicle crimes. The difference is evident by the visual interpretation of two sets of images. However, only the visual interpretation cannot justify the crime rates predicted. The visual features or elements are not explicitly extracted from the images, so it is difficult to interpret the results. As the scatterplots suggest, the underestimation of higher values is evident in the predictions of Figure 5.17. Class activation maps can be generated to interpret the attention of CNN to image in predicting a class. The heatmaps can be generated to understand the features responsible in an image for certain predictions. As the model is heavily altered from the original model, generating these class activation maps is cumbersome. Finally, it can be understood from the results obtained that visual features extracted from SVI can be used to predict the crime rate. However, there are few limitations which are explained in the next section.

5.4. Limitation

First, the population density data available is dated back to 2011. However, the crime data and the SVI are obtained for the time period 2018-2019. So, this may be the most likely reason for population density not affecting the result as expected. Second, the availability of socio-economic data at the desired scale. Socio-economic variables may have a greater impact on predicting crime. The socio-economic data can help in understanding the neighbourhood better in addition to visual variables. This data is not available at the desired scale, and disassociation of such properties can be difficult and heavily altered. The availability of such data helps in the model training process. Third, the capability and generalizing the model with other study areas. The comparison studies help to transfer the model to predict for the areas with limited data availability. Fourth, better system requirements. Initially, in place of ResNet18, a more common model, ResNet50 is used for the experimentation. However, the hardware is not compatible and handling the 4-CNN was not possible. ResNet50 is deeper than ResNet18 and has better accuracy for classification. Fifth, if a single CNN module is used, class activation maps can be generated, and the features in an image responsible for a certain prediction can be interpreted and understood. With the current configuration of the model, it is not easy to generate the activation maps.

6. CONCLUSION AND RECOMMENDATIONS

The previous chapter evaluated the results and the methodology. This chapter outlines the research work done and answers the research questions framed in chapter 1. Finally concludes with few recommendations for future work.

6.1. Conclusion

This research studies the effect of visual variables of environment inferred from SVI in quantifying the crime rate. A CNN model is used to learn the features from SVI and associate them with the crime rates of four different crime types. The multi-value prediction is achieved by solving the multi-output regression problem. A workflow is designed to model the crime rates and prepare the data for input to the model. Initially, the crime data is modelled using KDE, which is used commonly for crime hotspot analysis. The output of KDE, along with the road network, is used to extract SVI and eventually prepare input data for the model. Finally, a 4-CNN model is built with ResNet18 blocks, and the multi-output regression is implemented to handle the multiple inputs and outputs. The model takes five inputs, population density, and SVI and predicts crime rates for four crime types: burglary, robbery, other thefts and vehicle crimes. Different models are configured to handle different combinations of inputs and outputs. The results show a considerate relationship between visual variables and crime. Three models with different configurations are compared to evaluate the extent of the relationship between SVI and crime rate. The results show a significant relationship between visual variables and the crime rate, but additional variables like socio-economic variables can significantly impact the model's performance.

The answer to research questions are as follows:

1) What crime types are to be considered for the study?

The crime types of burglary, robbery, other thefts, and vehicle crimes are considered for the study. The UK police classify the crime types and their subclasses on their website. These crime types are also considered street crimes. Street crimes usually happen in public places and often involve violence. Bicycle theft, violent and sexual offences, arson, and white-collar crimes are not considered due to inadequate data and ambiguity in the data.

2) How to model the distribution of crime data?

The crime data is modelled using Kernel Density Estimation (KDE). KDE is a density estimation technique that smoothens the crime point pattern and assigns a density value to the decided cell size (the cell size is 250m x250m). It also takes the crimes in neighbourhood cells for density estimation. When multiplied by the cell area size, the density gives the respective crime count or crime rate. The obtained density map is used for further analysis in the research.

3) What is the best strategy to label the SVI?

A uniform grid of cell size 250m x 250m is overlayed on the KDE output, and this cell is used as a unit of analysis in the research. In each cell, two points are generated which represent the cell. The value of the cell is assigned to the point, and the SVI extracted from the point. Similarly, the population density values are also taken from the OA in which the point is present. The identifiers for the SVI are generated by appending a number (0,1,2,3) to the point identifier from which the SVI is extracted. All the identifiers, crime rate values, and population density are compiled in a table which is then processed to input the data to the model.

4) Which CNN architecture best quantifies the relationship between SVI and crime occurrence?

ResNet18 is selected to build the DL model. It belongs to the ResNet family and is a lightweight model with fewer parameters than legacy architectures like AlexNet and VGGNet. Moreover, it uses skip connections to handle the vanishing gradient problem and trains faster than other heavy models. As the model built uses 4 CNN backbones, ResNet18 is the best suited lightweight architecture for the purpose.

5) How to achieve the multiple-output regression?

Multi-output regression is an extension of single-output regression. In the output layer of the built 4-CNN model, a linear activation is used and Mean Square Error (MSE) is used as a loss function. The MSE is calculated for all the crime types and averaged to get a single value to adjust the weights.

6) How to implement the model to handle multiple inputs and multiple outputs?

The model is built to simultaneously take five inputs, out of which four inputs are images. The images are processed through CNN, and features are extracted. The extracted features are then fused together and progressed forward to the output layer with four nodes and have a linear activation function. A custom data generator is built to process a table and generate data to input to the model. The data generator takes the records from the labelled data table and prepares the inputs and outputs to input to the model.

7) How to select training, validation, and test sets to train and configure the deep learning model?

The uniform grid cells overlayed on KDE output are used to split the dataset. Once the images are extracted, and the data is cleaned for blank and interior images, the remaining cells are considered for data selection. The total cells are divided in ratios 60%, 20%, 20% for training (12239 cells), validation (4080 cells), and testing (4079 cells) sets by random selection.

8) To what extent can the visual variables from SVI explain the crime occurrence?

A model which takes only SVI as input and predicts the crime rates is configured. The R-squared value explains the variance of crime occurrence by SVI. The R-squared value for burglary is 51%, robbery is 44%, other thefts is 50%, and vehicle crimes is 49%. The values are calculated on the predictions obtained from an unseen dataset by the model.

6.2. Recommendations

The recommendations for future works and stakeholders are suggested in this section. The model built in this work uses a fully connected layer with the help of a loss function to perform the multi-output regression. However, first, the features can be extracted from the model, and a different regressor like Random Forest regressor or Support Vector Regressor can be used to perform the multi-output regression. Socio-economic variables also play a major role along with environmental variables. There is scope to add more variables to the developed model. However, the scale at which the data is available, its pre-processing, and the cell size for the unit of analysis needs attention. The dataset size also needs to be chosen accordingly. The temporal dimension can be considered in the crime rate prediction. The factors affecting the distribution of crime in time and the time itself can be added as variables. The predictions of different crime types can be combined to create a prospective risk map that helps the police department and the public. Certain crime types can be given weightage if needed. It helps the police department to manage the resources at hand effectively. Urban planners can use the outputs to understand the built environments much better and plan the city accordingly. It reduces the effort of field surveys. Nevertheless, there is a major scope of improvement in the proposed workflow and developed model.

LIST OF REFERENCES

Andersson, V. O., Birck, M. A. F., & Araujo, R. M. (2017). Investigating Crime Rate Prediction Using Street-Level Images and Siamese Convolutional Neural Networks. *Communications in Computer and Information Science*, *720*(December), 81–93. https://doi.org/10.1007/978-3-319-71011-2_7

Arietta, S. M., Efros, A. A., Ramamoorthi, R., & Agrawala, M. (2014). City forensics: Using visual elements to predict non-visual city attributes. *IEEE Transactions on Visualization and Computer Graphics*, *20*(12), 2624–2633. https://doi.org/10.1109/TVCG.2014.2346446

Bowers, K. J., Johnson, S. D., & Pease, K. (2004). Prospective hot-spotting: The future of crime mapping? *British Journal of Criminology*, *44*(5), 641–658. https://doi.org/10.1093/bjc/azh036

Brantingham, P., & Brantingham, P. (1995). Criminality of place - Crime generators and crime attractors. *European Journal on Criminal Policy and Research*, *3*(3), 5–26. https://doi.org/10.1007/BF02242925

Chainey, S., Tompson, L., & Uhlig, S. (2008). The Utility of Hotspot Mapping for Predicting Spatial Patterns of Crime. *Security Journal*, *21*(1–2), 4–28. https://doi.org/10.1057/palgrave.sj.8350066

Chopra, S., Hadsell, R., & LeCun, Y. (2005). Learning a similarity metric discriminatively, with application to face verification. *Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005*, *I*, 539–546. https://doi.org/10.1109/CVPR.2005.202

Clark, D. (2021, July 8). *Crime rate in London 2010-2020*. Statista. https://www.statista.com/statistics/380963/london-crime-rate/

Cohen, D. A., Mason, K., Bedimo, A., Scribner, R., Basolo, V., & Farley, T. A. (2003). Neighborhood physical conditions and health. *American Journal of Public Health*, *93*(3), 467–471. https://doi.org/10.2105/AJPH.93.3.467

Doersch, C., Singh, S., Gupta, A., Sivic, J., Efros, A., & Efros, A. A. (2012). What Makes Paris Look like Paris? In *ACM Transactions on Graphics* (Vol. 31, Issue 4). Association for Computing Machinery. https://hal.inria.fr/hal-01053876

Dubey, A., Naik, N., Parikh, D., Raskar, R., & Hidalgo, C. A. (2016). Deep learning the city: Quantifying urban perception at a global scale. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *9905 LNCS*, 196–212. https://doi.org/10.1007/978-3-319-46448-0_12

Fu, K., Chen, Z., & Lu, C. T. (2018). StreetNet: Preference learning with convolutional neural network on urban crime perception. *GIS: Proceedings of the ACM International Symposium on Advances in Geographic Information Systems*, *3274975*(c), 269–278. https://doi.org/10.1145/3274895.3274975

Hart, T., & Zandbergen, P. (2014). Kernel density estimation and hotspot mapping: Examining the influence of interpolation method, grid cell size, and bandwidth on crime forecasting. *Policing*, *37*(2), 305–323. https://doi.org/10.1108/PIJPSM-04-2013-0039

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep Residual Learning for Image Recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *2016-Decem*, 770–778. https://arxiv.org/abs/1512.03385v1

HORNE, J. S., & GARTON, E. O. (2006). Likelihood Cross-Validation Versus Least Squares Cross-Validation for Choosing the Smoothing Parameter in Kernel Home-Range Analysis. *Journal of Wildlife Management*, *70*(3), 641–648. https://doi.org/10.2193/0022-541x(2006)70[641:lcvlsc]2.0.co;2

Hu, Y., Wang, F., Guin, C., & Zhu, H. (2018). A spatio-temporal kernel density estimation framework for predictive crime hotspot mapping and evaluation. *Applied Geography*, *99*(May 2020), 89–97. https://doi.org/10.1016/j.apgeog.2018.08.001

Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, *2017-January*, 2261–2269. https://doi.org/10.1109/CVPR.2017.243

Ibrahim, M. R., Haworth, J., & Cheng, T. (2020). Understanding cities with machine eyes: A review of deep computer vision in urban analytics. *Cities*, *96*, 102481. https://doi.org/10.1016/j.cities.2019.102481

Kang, H.-W., & Kang, H.-B. (2017). Prediction of crime occurrence from multi-modal data using deep learning. *Plos One*, *12*(4), e0176244. https://doi.org/10.1371/journal.pone.0176244.t005

Kang, H. W., & Kang, H. B. (2017). Prediction of crime occurrence from multimodal data using deep learning. *PLoS ONE*, *12*(4). https://doi.org/10.1371/journal.pone.0176244

Kelly, C. M., Wilson, J. S., Baker, E. A., Miller, D. K., & Schootman, M. (2013). Using Google Street View to audit the built environment: Inter-rater reliability results. *Annals of Behavioral Medicine*, *45*(SUPPL.1). https://doi.org/10.1007/s12160-012-9419-9

Khosla, A., An, B., Lim, J. J., & Torralba, A. (2014). Looking beyond the visible scene. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 3710–3717. https://doi.org/10.1109/CVPR.2014.474

Kinney, J. B., Brantingham, P. L., Wuschke, K., Kirk, M. G., & Brantingham, P. J. (2008). Crime attractors, generators and detractors: Land use and urban crime opportunities. *Built Environment*, *34*(1), 62–74. https://doi.org/10.2148/benv.34.1.62

Kounadi, O., Ristea, A., Araujo, A., & Leitner, M. (2020). A systematic review on spatial crime forecasting. *Crime Science*, *9*(1), 1–22. https://doi.org/10.1186/s40163-020-00116-7

Krizhevsky, B. A., Sutskever, I., & Hinton, G. E. (2012). *Communications of the ACM-2017-Krizhevsky-Hinton-ImageNet classification with deep convolutional neural networks.pdf.*

Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. In *Nature* (Vol. 521, Issue 7553, pp. 436–444). Nature Publishing Group. https://doi.org/10.1038/nature14539

Li, Y., Chen, Y., Rajabifard, A., Khoshelham, K., & Aleksandrov, M. (2018). Estimating building age from google street view images using deep learning. *Leibniz International Proceedings in Informatics, LIPIcs*, *114*(May 2020). https://doi.org/10.4230/LIPIcs.GIScience.2018.40

Milam, A. J., Furr-Holden, C. D. M., & Leaf, P. J. (2010). Perceived School and Neighborhood Safety, Neighborhood Violence and Academic Achievement in Urban School Children. *Urban Review*, *42*(5), 458–467. https://doi.org/10.1007/s11256-010-0165-7

Naik, N., Kominers, S. D., Raskar, R., Glaeser, E. L., & Hidalgo, C. A. (2017). Computer vision uncovers predictors of physical urban change. *Proceedings of the National Academy of Sciences of the United States of America*, *114*(29), 7571–7576. https://doi.org/10.1073/pnas.1619003114

Naik, N., Philipoom, J., Raskar, R., & Hidalgo, C. (2014). Streetscore-predicting the perceived safety of one million streetscapes. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 793–799. https://doi.org/10.1109/CVPRW.2014.121

Piro, F. N., Nœss, Ø., & Claussen, B. (2006). Physical activity among elderly people in a city population: The influence of neighbourhood level violence and self perceived safety. *Journal of Epidemiology and Community Health*, *60*(7), 626–632. https://doi.org/10.1136/jech.2005.042697

Salesses, P., Schechtner, K., & Hidalgo, C. A. (2013). The Collaborative Image of The City: Mapping the Inequality of Urban Perception. *PLoS ONE*, *8*(7), e68400. https://doi.org/10.1371/journal.pone.0068400

Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 1–14.

Wilson, J. Q., & Kelling, G. L. (1982). Broken Windows. *The Atlantic Monthly*, *March*, 1–8. http://www.theatlantic.com/doc/print/198203/broken-windows

Zhang, F., Wu, L., Zhu, D., & Liu, Y. (2019). Social sensing from street-level imagery: A case study in learning spatio-temporal urban mobility patterns. *ISPRS Journal of Photogrammetry and Remote Sensing*, *153*, 48–58. https://doi.org/10.1016/j.isprsjprs.2019.04.017

Zhang, Z., Chen, D., Liu, W., Racine, J. S., Ong, S. H., Chen, Y., Zhao, G., & Jiang, Q. (2011). Nonparametric evaluation of dynamic disease risk: A spatio-temporal kernel approach. *PLoS ONE*, *6*(3). https://doi.org/10.1371/journal.pone.0017381

Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., & Torralba, A. (2018). Places: A 10 Million Image Database for Scene Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *40*(6), 1452–1464. https://doi.org/10.1109/TPAMI.2017.2723009

Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., & Oliva, A. (2014). Learning deep features for scene recognition using places database. *Advances in Neural Information Processing Systems*, *1*(January), 487–495.