Swarm Intelligence Response Methods Based on Urban Crime Event Prediction

Changhao Wang 1, Feng Tian 1 and Yan Pan 2,*

1 School of Electronic Information and Artificial Intelligence, Shaanxi University of Science and Technology, Xi’an 710021, China; wangchanghao@sust.edu.cn (C.W.); 211612093@sust.edu.cn (F.T.)
2 Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, Changsha 410073, China
* Correspondence: panyan@nudt.edu.cn

Abstract: Cities attract a large number of inhabitants due to their more advanced industrial and commercial sectors and more abundant and convenient living conditions. According to statistics, more than half of the world’s population resides in urban areas, contributing to the prosperity of cities. However, it also brings more crime risks to the city. Crime prediction based on spatiotemporal data, along with the implementation of multiple unmanned drone patrols and responses, can effectively reduce a city’s crime rate. This paper utilizes machine learning and data mining techniques, predicts crime incidents in small geographic areas with short timeframes, and proposes a random forest algorithm based on oversampling, which outperforms other prediction algorithms in terms of performance. The research results indicate that the random forest algorithm based on oversampling can effectively predict crimes with an accuracy rate of up to 95%, and an AUC value close to 0.99. Based on the crime prediction results, this paper proposes a multi-drone patrol response strategy to patrol and respond to predicted high-crime areas, which is based on target clustering and combined genetic algorithms. This strategy may help with the pre-warning patrol planning within an hourly range. This paper aims to combine crime event predictions with crowd-sourced cruise responses to proactively identify potential crimes, providing an effective solution to reduce urban crime rates.

Keywords: random forest; patrol response; genetic algorithm; crime event prediction

1. Introduction

In recent years, with the accelerated urbanization process and improved living standards, cities have become hubs for every large population. However, this has also brought about a series of security problems, leading to various types of crimes. According to the statistics from the United Nations Office on Drugs and Crime (UNODC), various types of crimes, including robbery, theft, and violent crimes, are on the rise worldwide [1]. Data released by the Chinese Ministry of Public Security shows that overall crime cases in China have been on the rise in recent years, posing serious challenges to social order and public safety [2,3]. According to daily statistical reports from the China Emergency Service Network, in July 2018 alone, there were 21,426 reports of warning information and 5129 reports of emergency incidents nationwide [4]. Criminal events seriously affect economic development and social stability, posing a significant threat to the safety of people’s lives.

With the advancement of artificial intelligence and big data analytics technology, many scholars both domestically and internationally are using spatiotemporal data to predict occurrences of criminal events, providing effective means to reduce crime incidents. Using machine learning, researchers like Jenga [5] have assessed the latest technologies in crime prediction and proposed future research directions in the field of crime prediction. Zhang et al. [6], for instance, extracted patterns and features from historical crime data and improved the accuracy of LSTM models by incorporating built environment covariates.
These achievements in crime event prediction have been substantial, but many have not addressed intelligent responses based on the prediction results.

Using crime prediction results as a base, designing real-time patrol routes for multiple drones to comprehensively cover high-crime areas can effectively prevent crime incidents and enable rapid responses when crimes occur, thereby reducing crime rates and losses. This paper utilizes machine learning and big data analytic techniques to propose a crowd-sourced response method based on crime event prediction for both proactive alerts and post-event responses. The main contributions of this work are as follows:

- Crime events in short time frames and small areas are predicted, using data mining techniques and various machine learning methods. The specific prediction process is described in Section 3, and the prediction results is described in Section 3.4.4. The random forest algorithm is based on the oversampling proposed in this paper and outperforms other prediction algorithms. The results show that the oversampled random forest prediction has an accuracy of up to 95%, an AUC value close to 0.99, an F1-score of 0.94, and a recall of 0.95;
- A drone patrol response strategy built upon the foundation of the previous section is designed based on target clustering, as is described in Section 4. Combined with a genetic algorithm, this strategy can be used for patrols and responds to high-crime areas predicted in advance. The experimental results can help patrol planning with area-wide pre-warning within one hour, providing an effective solution to reduce urban crime rates.

In this study, Figure 1 presents the proposed system structure, which consists of two parts: crime event prediction and the collective intelligence response. It includes a central database, a server, and drones equipped with various sensors (UAVS). The central database stores various feature data and prediction results. The server is the central processing entity in the system, responsible for data preprocessing, model computation, crime prediction, and the allocation of drones to patrolled areas. First, data are inputted into the model on the left, and the predicted output P is set to be the high-crime area in region A_n at time period T_m, corresponding to Targets in the circle on the right. Next, UAVS are clustered into UAVS_i, where i = j, with each UAV patrolling its corresponding targets. Finally, drones can collect data through sensors during patrolling and feed them back to the dataset before repeating the above process.

Figure 1. System structure proposed in this study.

This paper aims to integrate crime event prediction with the collective intelligence response. It provides a comprehensive approach that can proactively identify potential crime events, optimize resource allocation, and plan response routes, thereby, effectively
addressing urban crime issues and enhancing public safety. This approach also offers valuable insights for predicting and responding to other emergent incidents, contributing to the field of smart disaster prevention in cities.

2. Relevant Work

2.1. Crime Event Prediction

Crime event prediction involves the utilization of techniques such as data analysis, statistical methods, and machine learning to forecast potential future criminal incidents by analyzing historical crime data along with spatiotemporal factors associated with criminal incidents.

Primarily, crime events are closely correlated with external factors and predicting them involves analyzing and modeling this relationship. In other words, it entails training a model that establishes a connection between crime events and external factors to enable the model to predict criminal incidents. For example, Catlett et al. [7] introduced a predictive approach based on a spatial analysis and autoregressive model. This method employs a density-sensitive clustering algorithm Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [8] and seasonal Autoregressive Integrated Moving Average (ARIMA) [9] models to automatically detect high-risk crime areas within a city and reliably forecast crime trends.

Furthermore, enhancing prediction accuracy can be achieved by analyzing the spatiotemporal characteristics of data. Yi et al. [10] proposed a Clustered Continuous Conditional Random Field (Clustered-CCRF) model that combines autoregressive temporal correlation and feature-based spatial correlation between regions. By utilizing a tree-based clustering algorithm, highly similar regions are identified, thereby, enhancing the performance of crime prediction models. Zhang et al. [11] developed an interpretable machine learning crime prediction model that combines Extreme Gradient Boosting (XGBoost) [12,13] and Shapley additive explanation (SHAP) [14] methods. This model elucidates the precise spatial variation of each variable, thereby, improving the accuracy and transparency of the crime prediction model. Hajela et al. [15] introduced a spatiotemporal crime prediction technique based on machine learning and a two-dimensional hotspot analysis. They employed a finer-grained partitioning method to capture the spatial distribution characteristics of crime and used multiple features and complex classification models to enhance prediction accuracy and robustness. These approaches have achieved certain breakthroughs in the realm of spatiotemporal features.

Additionally, in certain instances, different types of security events may exhibit evident or covert interactions. For instance, Ahsan et al. [16] proposed a machine learning-based approach to analyze and predict road traffic accident risks. They selected multiple attributes as input variables and utilized decision trees and random forest algorithms to predict accidents. Thakkar et al. [17] employed a random forest algorithm to predict fires, calculating the Pearson correlation coefficients between variables and using a correlation matrix to unveil relationships between different factors and the probability of fire occurrence. Their research provided a novel approach and reference for preventing and controlling fires and traffic accidents, concurrently offering valuable insights for the crime event prediction aspect of this paper.

The advantages and disadvantages of the aforementioned technologies are shown in Table 1.

Currently, both domestic and international research on crime event predictions primarily focuses on long-term and large-scale forecasting; yet, such coarse predictions fail to meet modern urban requirements for crime event responses. Therefore, this paper proposes a granular prediction algorithm based on spatiotemporal factors. By utilizing an oversampled random forest algorithm, it forecasts crime incidents in small geographical areas within short time periods for the subsequent collective intelligence responses.
Table 1. Summary of Works Related to Crime Event Prediction.

<table>
<thead>
<tr>
<th>Method/Model</th>
<th>Technical Advantages</th>
<th>Technical Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spatial analysis and ARIMA [7]</td>
<td>Automatic detection of high-risk crime areas using DBSCAN and ARIMA models</td>
<td>Requires appropriate parameter configuration</td>
</tr>
<tr>
<td>Clustered-CCRF model [10]</td>
<td>Combines temporal and spatial correlation, improving prediction performance</td>
<td>Requires an effective data clustering method</td>
</tr>
<tr>
<td>XGBoost and SHAP methods [11]</td>
<td>Enhanced model accuracy and transparency, explaining spatial variations</td>
<td>May require substantial computational resources</td>
</tr>
<tr>
<td>2D hotspot analysis and machine learning [15]</td>
<td>Improved prediction accuracy and robustness, fine-grained spatial partitioning</td>
<td>Complex classification models may require large datasets</td>
</tr>
<tr>
<td>Decision trees and random forest [16]</td>
<td>Effective prediction of road traffic accidents, multiple input attributes</td>
<td>More feature engineering and data preprocessing may be needed</td>
</tr>
<tr>
<td>Random forest algorithm [17]</td>
<td>Revealing correlations between different factors, providing fire prevention methods</td>
<td>Data quality and correlations may affect prediction accuracy</td>
</tr>
</tbody>
</table>

2.2. Multi-Agent-Based Collective Intelligence Response

In recent years, intelligent agents represented by unmanned aerial vehicles (UAVs), as compared to ground vehicles, have gained widespread application in areas such as missing person searches, medical transports, and emergency communication due to their agility and efficiency [18–24]. Goodrich et al. [25] and Nakadai et al. [26] implemented camera-equipped and microphone-equipped intelligent collectives, respectively, and tested their performance in searching and rescuing tasks. Advanced planning and navigation algorithms are the key for intelligent collectives to accomplish emergency response tasks. Heintzman et al. [27] primarily focused on target-predictive motion models and investigated path planning for human-intelligent collective search and rescue movements. Wu et al. [28] utilized Markov models to address uncertainties in the deployment of intelligent collective-relief personnel. Liu et al. [29] and Huang et al. [30] employed road models to guide intelligent collectives in road anomaly detection and response. Ding et al. [31] improved the Particle Swarm Optimization (PSO) [32] algorithm and applied the enhanced Artificial Bee Colony-Particle Swarm Optimization (ABC-PSO) algorithm to solve task allocation problems, effectively addressing the reassignment of tasks in multi-agent emergency relief scenarios. Han et al. [33] introduced an optimized A* path planning algorithm, allocating intelligent collectives and rapidly devising optimal flight paths. Zhao et al. [34] proposed a unified framework for UAV-assisted disaster emergency networks and investigated optimization problems in three scenarios, including UAV trajectory and scheduling, transmitter–receiver design, multi-hop Device-to-Device (D2D) [35] communication, and multi-hop UAV relay. Jaradat et al. [36] employed a finite-state Q-Learning algorithm to enhance the efficiency of path planning in unknown environments. Liu et al. [37] proposed an improved reinforcement learning method, Neural Networks Heuristic Q-learning (NNH-QL), using Back Propagation (BP) [38] neural networks, enhancing the learning efficiency of the Q-learning algorithm with the neural network’s fitting and enabling the algorithm to operate effectively in larger environments. Zhao et al. [39] presented a UAV path planning algorithm based on deep multi-agent reinforcement learning, considering complex urban environments, flight time restrictions, wireless channel characteristics, and various scenario parameters, achieving generalization across diverse scenarios without the need for retraining or adaptation.

The advantages and disadvantages of the aforementioned technologies are shown in Table 2.

These studies collectively demonstrate significant advancements in intelligent collectives for emergency response tasks. Advanced planning and navigation algorithms empower them to efficiently tackle emergency communication and other tasks, offering robust support for societal safety and rescue efforts. However, most existing research predominantly focuses on post-incident responses, while this paper centers on proactive pre-warning patrols and post-emergency responses. By predicting potential events in
advance, intelligent collectives can take preemptive measures before incidents occur, effectively managing crises and emergency situations. Urban-scale crime events possess highly dynamic spatiotemporal characteristics, providing intelligent collectives for the opportunity to forecast and optimize patrol strategies, facilitating both swift post-event responses and pre-event measures. This spatiotemporal dynamism presents intelligent collectives with more effective and flexible options for emergency responses.

### Table 2. Summary of Works Related to Collective Intelligence Response.

<table>
<thead>
<tr>
<th>Approach/Algorithm</th>
<th>Technical Advantages</th>
<th>Technical Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microphone-equipped intelligent collectives [26]</td>
<td>Enhanced audio-based search capabilities</td>
<td>Limited use in non-audio scenarios</td>
</tr>
<tr>
<td>Target-predictive motion models [27]</td>
<td>Improved path planning for collective</td>
<td>Complexity in modeling target motion</td>
</tr>
<tr>
<td>Markov models [28]</td>
<td>Addressing uncertainties effectively</td>
<td>Limited applicability in certain tasks</td>
</tr>
<tr>
<td>Road models [29]</td>
<td>Effective guidance for road-based tasks</td>
<td>Limited to tasks related to roads</td>
</tr>
<tr>
<td>Enhanced PSO (ABC-PSO) algorithm [31]</td>
<td>Improved task reassignment in emergencies</td>
<td>Potential complexity in parameter tuning</td>
</tr>
<tr>
<td>Optimized A* path planning algorithm [33]</td>
<td>Rapid task allocation and path planning</td>
<td>May not handle complex urban scenarios</td>
</tr>
<tr>
<td>Trajectory and scheduling optimization [34]</td>
<td>Comprehensive framework for UAV scenarios</td>
<td>May require complex optimization</td>
</tr>
<tr>
<td>Finite-state Q-Learning algorithm [36]</td>
<td>Enhanced path planning in unknown areas</td>
<td>Limited to simple Q-learning scenarios</td>
</tr>
<tr>
<td>Neural Networks Heuristic Q-learning [37]</td>
<td>Improved learning efficiency</td>
<td>Potential complexity in neural networks</td>
</tr>
</tbody>
</table>

### 3. Crime Event Prediction

#### 3.1. Overview of Denver Crime Datasets

In this paper, publicly available datasets from the city of Denver in the United States were utilized. These datasets encompass spatiotemporal information from 2016 to 2021. The raw data comprise 546,882 records and 19 columns, with each record representing an individual event record and each column representing distinct attributes of the event, as illustrated in Table 3. This dataset encompasses a total of 19 attributes.

### Table 3. Attributes of the Crime Dataset.

<table>
<thead>
<tr>
<th>Id</th>
<th>Columns</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>OFFENSE_ID</td>
</tr>
<tr>
<td>2</td>
<td>INCIDENT_ID</td>
</tr>
<tr>
<td>3</td>
<td>OFFENSE_CODE</td>
</tr>
<tr>
<td>4</td>
<td>OFFENSE_CODE_EXTENSION</td>
</tr>
<tr>
<td>5</td>
<td>OFFENSE_TYPE_ID</td>
</tr>
<tr>
<td>6</td>
<td>OFFENSE_CATEGORY_ID</td>
</tr>
<tr>
<td>7</td>
<td>FIRST_OCCURRENCE_DATE</td>
</tr>
<tr>
<td>8</td>
<td>LAST_OCCURRENCE_DATE</td>
</tr>
<tr>
<td>9</td>
<td>REPORTED_DATE</td>
</tr>
<tr>
<td>10</td>
<td>INCIDENT_ADDRESS</td>
</tr>
<tr>
<td>11</td>
<td>GEO_LON</td>
</tr>
<tr>
<td>12</td>
<td>GEO_LAT</td>
</tr>
<tr>
<td>13</td>
<td>GEO_X</td>
</tr>
<tr>
<td>14</td>
<td>GEO_Y</td>
</tr>
<tr>
<td>15</td>
<td>DISTRICT_ID</td>
</tr>
<tr>
<td>16</td>
<td>PRECINCT_ID</td>
</tr>
<tr>
<td>17</td>
<td>NEIGHBORHOOD_ID</td>
</tr>
<tr>
<td>18</td>
<td>IS_CRIME</td>
</tr>
<tr>
<td>19</td>
<td>IS_TRAFFIC</td>
</tr>
</tbody>
</table>

#### 3.2. Data Preprocessing

##### 3.2.1. Attribute Reduction

The original dataset contains 19 attributes, some of which are irrelevant to this study. These redundant attributes are removed from the analysis.

##### 3.2.2. Handling Missing Values

The original datasets comprise 546,882 records, after analyzing with the latitude and longitude that exhibit the most severe instances of missing values which account for
approximately 0.8% of the total. As the proportion of missing values is relatively small, after these instances were removed, a final dataset of 505,285 records were left.

3.2.3. Feature Extraction

Temporal features, such as “Year”, “Month”, “Weekday”, “Hour”, and “Minute” representing the incident occurrence time, are extracted from the “LAST_OCCURRENCE_DAT” field. To effectively facilitate the analysis of feature importance and model training, unique numerical identifiers are assigned for different blocks, months, and weekdays. For instance, [1: ‘Mon’, 2: ‘Thu’, . . ., 7: ‘Sun’].

3.2.4. Spatial Division

This paper employs the United States National Grid (USNG) system to convert GPS coordinates into unique block identifiers serving as new input features. This feature is labeled as “Grid_3Km_no”, offering accuracy down to 10 m. The study divides the area into grid cells of 3 km × 3 km, resulting in a total of 84 grids. This division approach enhances the foundation for model training.

3.3. Exploratory Analysis and Feature Selection

3.3.1. Temporal Correlation Analysis

In this section, an analysis of the number of crime events, which occurred during different time periods in Denver, is conducted, with different time intervals exhibiting distinct patterns in the distribution of crime events. As is shown in Figure 2a, there is a fluctuation in the number of crime events between 2020 and 2021, displaying an upward trend. This suggests that public security in Denver still requires improvement. Figure 2b illustrates the monthly distribution of crime events. The results reveal that August has the highest occurrence rate of crime events, followed by July and January, while February witnesses the lowest number of incidents. Figure 2c delves into the impact of dates within each month on the number of crime events. Notably, there is a dip in occurrences on the 31st day, as not all months have 31 days. On the contrary, a peak is observed on the 1st day, possibly attributed to New Year’s Day and its higher population density, which may lead to an abnormal increase in crime incidents.

![Figure 2. Temporal correlation analysis. (a) The trend of crime numbers over the years; (b) The trend of crime numbers over the months; (c) The trend of crime numbers over the days; (d) The trend of crime numbers over the hours.](image-url)
As is depicted in Figures 2d and 3, there are significant variations in the number of crime events during different time intervals throughout the day. On workdays, the occurrences of crime events peak between 12:00 and 18:00, with a smaller peak observed between 12:00 and 13:00. On weekends, the distribution trend of crime events is similar to that of the workdays. Before 5:00 AM, the total number of crime events on weekends tends to surpass that of the workdays.

![Figure 3](image.png)

**Figure 3.** Incidence of crime by time of week.

The aforementioned analysis shows a close correlation between the occurrence of crime events and temporal factors. Consequently, features such as year (Year), month (Month), weekday (Week), and hour (Hour) are chosen as temporal features for the analysis and modeling, with an aim to achieve accurate crime event predictions and swift responses for an optimized resource allocation. This approach seeks to enhance public safety and effectively deal with security risks that arise during various time periods.

### 3.3.2. Spatial Correlation Analysis

In this section, we delve into the correlation between crime events and spatial factors. By means of a statistical analysis, the spatial distribution patterns of crime events can be derived, as is depicted in Figure 4. The entire city is partitioned into seven police districts, with each color representing the distribution of crime events in a specific district. A distinct difference in district sizes is evident, as seen from the Figure. Specifically speaking, District 3 has the largest area, followed by Districts 4 and 5, while District 6 encompasses the smallest area. This implies that District 3 covers a relatively larger area, and likely encompasses the most densely populated regions and various social activity hubs, which may result in a higher concentration of crime events. In contrast, District 7’s smaller area might indicate a more secluded region or a relatively sparser community population, potentially leading to fewer crime events and a comparatively safer environment.

![Figure 4](image.png)

**Figure 4.** Illustration of the distribution of crime incidents across police precincts.

However, reality does not always adhere to this notion. The size of a district does not necessarily correlate with the number of crime events, as is shown in Figure 4, which presents a heatmap of crime events in Denver. This heatmap illustrates the density of crime...
events within each police district, with brighter colors indicating a higher crime event density. Despite District 6 having the smallest area, it displays the highest crime density, indicating that it is a hotspot for criminal activity. This might be linked to factors such as population density, social activity hubs, or other variables that concentrate criminal incidents in this region. On the other hand, despite District 3’s area being the largest, its crime event density is relatively low. This suggests that while the district covers a larger region, it might include relatively safer communities or areas.

Taking into consideration the analysis from both Figures 4 and 5, it can be concluded that the size of a police district does not directly determine the number of crime events. In fact, the spatial distribution of crime events is more influenced by internal social and environmental factors within each district. Therefore, when formulating crime prevention measures, it is crucial to consider both the district’s size and the hotspots of criminal activity within it. In this way, targeted measures can be adopted to enhance public safety effectively.

**Figure 5.** Displays of a heatmap of crime incidents.

### 3.3.3. Feature Selection

Based on the aforementioned analysis, it is evident that the factors influencing crime primarily involve temporal and spatial aspects. To facilitate model computation and maximize the utilization of these attributes for predicting the probability of crime occurrence within a region, this paper has selected 14 attributes, as are illustrated in Table 4.

<table>
<thead>
<tr>
<th>Factors</th>
<th>Feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Temporal</td>
<td>Year, Min, Day, Week, Month, AM/PM</td>
</tr>
<tr>
<td>Spatial</td>
<td>GEO_LON, GEO_LAT, PRECINCT_ID, DISTRICT_ID, INCIDENT_ADDRESS, NEIGHBORHOOD_ID, Gird_3Km_no, IS_TRAFFIC</td>
</tr>
</tbody>
</table>

Among these attributes, “A_P_M”, “INCIDENT_ADDRESS”, “PRECINCT_ID”, and “NEIGHBORHOOD_ID” are nominal variables that need a Dummy Variable to be processed. Thus, one-hot encoding is employed to convert these nominal variables into binary vectors. This approach enhances the model’s ability to comprehend and process the data more effectively.
3.4. Crime Event Prediction

This section focuses on predicting crime events. Considering both temporal and spatial factors, precise patterns and distribution characteristics of events are captured through the analysis of historical data and real-time information. This is achieved using an oversampling-based random forest algorithm to accurately predict crime events occurring within a short time frame in a small area.

3.4.1. Problem Description

Initially, the time series $T$ is defined as a set comprising time points, denoted as $T = \{t_0, t_1, \cdots, t_m\}$, where $m \in [0, 23]$. Here, $m$ represents the division of a day into 24 equidistant time intervals, each spanning 1 h. This equidistant division of time intervals enhances the time series’ temporal relevance.

Subsequently, the spatial sequence $A$ is defined as a collection composed of spatial regions, represented as $A = \{a_0, a_1, \cdots, a_n\}$, where $n \in [0, k]$. In this context, $n$ signifies the division of geographical latitude and longitude into distinct regions, with $a_n$ signifying the $n$th region. This grid-based partitioning facilitates the segmentation of geographical space into numerous smaller areas, facilitating the analysis of region-specific attributes and the spatial localization of crime incidents.

By merging the time series and spatial sequence, it becomes possible to conduct the analysis and prediction of crime incidents within specific time intervals and designated spatial regions.

Furthermore, the forecasting objective is as follows: the primary goal of this study is to predict the locations of crime incidents occurring within each hour, alongside estimating the probability of crime occurrences at each location. This can be represented in the following manner:

$$P = \text{IsCrime} = 1 | T = t_m, A = a_n.$$ 

Equation (1) signifies the probability of a crime incident occurring in the time interval $[m, m + 1]$ at location $a_0$. Here, $T$ represents the time series, $t_m$ denotes the $m$th hour, $A$ signifies the spatial sequence, $a_n$ represents the $n$th region, and $P$ represents probability. The term “IsCrime = 1” within the formula signifies the occurrence of a crime incident at the specific location $a_n$ during the provided time interval $t_m$.

3.4.2. Crime Prediction Model

The traditional Random Forest algorithm typically employs balanced sample sets during the training process of each decision tree. However, when dealing with imbalanced datasets, certain classes may have fewer samples, which can lead to a poorer performance of the model in predicting minority classes. Oversampling-based Random Forest algorithms can enhance the classification performance on imbalanced datasets, as the use of oversampling techniques helps balance the data distribution among different classes by replicating existing minority class samples or generating new synthetic samples. This, in turn, improves the model’s ability to learn from minority classes.

For each sample $x$ in the minority class, calculate its Euclidean distance to all samples in the minority class sample set to obtain its $k$ nearest neighbors. Based on the imbalance ratio of the dataset, set a sampling ratio to determine the sampling multiplier $N$. For each minority class sample $x$, randomly select several samples from its $k$ nearest neighbors, assuming the selected neighbors are denoted as $x_{\text{new}}$. For each randomly selected neighbor $x_{\text{new}}$, construct a new sample according to Formula (2).

$$x_{\text{new}} = x + \text{rand}(0, 1) \times (x - x)$$

The positive-to-negative sample ratio in this dataset is approximately 3:1, as illustrated in Figure 6. Negative samples are augmented to achieve a 1:1 positive-to-negative sample ratio.
To prevent overfitting, assess the model’s performance objectively, and ascertain its generalization capability; the 5-Fold Cross Validation method is employed. The original training dataset was evenly divided into five subsets. Each subset’s data were used as a validation set in turn, while the remaining four subset data served as the training set, and this process was iterated. The model’s performance, as indicated in Table 5, demonstrates a notably high average accuracy, approximately 0.95, especially in the oversampled random forest model. Consequently, after data preprocessing, the original dataset was randomized and split in an 8:2 ratio to create a training set and a test set.

Table 5. 5-Fold Cross-Validation Average Accuracy Evaluation.

<table>
<thead>
<tr>
<th>Model</th>
<th>Average Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logistic Regression</td>
<td>0.6106596299259336</td>
</tr>
<tr>
<td>Bayesian Classifier</td>
<td>0.6690273878087176</td>
</tr>
<tr>
<td>KNN</td>
<td>0.6968863712611938</td>
</tr>
<tr>
<td>Decision Tree</td>
<td>0.9292069472760587</td>
</tr>
<tr>
<td>Random Forest</td>
<td>0.8563860906722042</td>
</tr>
<tr>
<td>Random Forest with Oversampling</td>
<td>0.9508477638132596</td>
</tr>
</tbody>
</table>

A random forest model with oversampling is used for crime prediction. Random forests utilize a decision tree approach, involving an optimal feature selection, with criteria such as the Gini Index and Information Gain [40]. Here, the Gini Index is used as the splitting criterion. This metric is highly sensitive in classification problems and effectively measures the impurity between different categories. As is shown in Table 6 through a simple validation, it can be observed that this metric results in a shorter training and prediction time. This makes it particularly advantageous to deal with large datasets, as it can significantly improve computational speed and reduce processing times.

Table 6. Time and Accuracy of Different Metrics.

<table>
<thead>
<tr>
<th>Criterion</th>
<th>Training Time/s</th>
<th>Prediction Time/s</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entropy</td>
<td>146.37401604652405</td>
<td>5.760926723480225</td>
<td>0.9504632377609745</td>
</tr>
<tr>
<td>Gini</td>
<td>140.28293132781982</td>
<td>5.735930585861206</td>
<td>0.9508309891868177</td>
</tr>
</tbody>
</table>

3.4.3. Crime Prediction Process

As is shown in Figure 7, the crime event prediction process in this paper follows the following steps:

1. Data Collection and Preprocessing. Collect historical crime data and other relevant information, and preprocess the data as necessary, including handling missing values, outlier treatment, and more.

2. Feature Engineering. Construct appropriate feature variables based on the data analysis’ results, which may include factors related to crime events such as time, location, etc.
3. Negative Sample Set Partitioning. Utilize feature recombination to reliably partition the negative sample set (samples where no crime occurred) to ensure the quality and effectiveness of the training set.
4. Training and Testing Set Splitting. Building upon the partitioned negative sample set, create training and testing sets for model training and evaluation.
5. Random Forest Model Construction. Select the random forest as the base classifier for learning due to its strong classification performance, ability to provide high accuracy and stable classification results, and effectiveness in handling high-dimensional data. The core pseudocode for this section is as shown in Algorithm 1, with specific parameter descriptions in Section 3.4.1.
6. Model Training. Train the random forest model using the training set, enabling the model to learn the characteristics and patterns of crime events.
7. Parameter Selection. Utilize 5-fold cross-validation and grid search techniques to select the optimal model parameters, further enhancing the model’s performance and generalization ability.
8. Model Evaluation and Prediction. Evaluate the well-trained random forest model using the testing set, calculate the metrics such as prediction accuracy, recall rate, etc., and employ the model to predict future crime events.

Figure 7. Outlines of the process flow for crime event prediction.

Algorithm 1: Crime Predict (Crime Datasets, Time, Area, Target = IsCrime)

INPUT: Crime Datasets (T, A, IsCrime)
OUTPUT: P = (IsCrime = 1 | T = t_m, A = a_n)

1. T = {t_0, t_1, \ldots, t_m} ← define time series() /* define time series
2. A = {a_0, a_1, \ldots, a_n} ← define spatial sequence() /* define spatial sequence
3. Merged data ← merge timeand spatial (T, A) /* merge time and spatial
4. for t_m in T:
5. for a_n in A:
6. P = predict crime probability (merged data) /* predict crime probability
7. if P(IsCrime) > 0.5 OR IsCrime = 1 /* crime has occurred
8. P = (IsCrime = 1 | T = t_m, A = a_n) /* probability of the crime
9. Else if P(IsCrime) < 0.5 OR IsCrime = 0 /* crime has not occurred
10. P = -1 /* output -1
11. END FOR /* end the loop
12. RETURN P /* output result
3.4.4. Crime Prediction Results

According to the prediction process shown in Figure 7, crime events were forecasted. The prediction models are evaluated using metrics such as the accuracy, precision, recall, and F1-score, and were compared with algorithms including logistic regression, decision trees, Bayesian methods, random forests, and KNN [5–7]. Precision is the ratio of true positives (TP) to the sum of true positives (TP) and false positives (FP), recall is the ratio of true positives (TP) to the sum of true positives (TP) and false negatives (FN), and the F1-score is the harmonic mean of precision and recall. Accuracy refers to the percentage of correct predictions [5,13]. As is shown in Table 7, the oversampled random forest model utilized in this study exhibits a superior performance.

Table 7. Model Evaluation Metrics.

<table>
<thead>
<tr>
<th>Model</th>
<th>Precision</th>
<th>Recall</th>
<th>F1-Score</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logistic Regression</td>
<td>0.62</td>
<td>0.61</td>
<td>0.61</td>
<td>0.61</td>
</tr>
<tr>
<td>Bayesian Classifier</td>
<td>0.67</td>
<td>0.67</td>
<td>0.67</td>
<td>0.67</td>
</tr>
<tr>
<td>KNN</td>
<td>0.70</td>
<td>0.70</td>
<td>0.69</td>
<td>0.70</td>
</tr>
<tr>
<td>Decision Tree</td>
<td>0.93</td>
<td>0.92</td>
<td>0.92</td>
<td>0.92</td>
</tr>
<tr>
<td>Random Forest</td>
<td>0.85</td>
<td>0.85</td>
<td>0.84</td>
<td>0.85</td>
</tr>
<tr>
<td>Random Forest with Oversampling</td>
<td>0.95</td>
<td>0.95</td>
<td>0.94</td>
<td>0.95</td>
</tr>
</tbody>
</table>

As is shown in Figure 8, the confusion matrices are presented for each crime prediction model. Clearly, the oversampled random forest model exhibits the highest proportion of TP (True Positives) and TN (True Negatives), indicating the best predictive performance. Therefore, for the remaining sections of this paper, we will use this model as the basis for predictions.

This section validates the algorithm’s performance by evaluating the model’s complexity. The experiments were conducted on a computer equipped with an Nvidia RTX 3080 graphics card and a dedicated 16-thread CPU. Table 8 presents the average accuracy, training time, testing set prediction time, and real-time simulation prediction time for each model over the 5-fold cross-validation. It can be observed that although the random forest model achieves the highest accuracy, it also exhibits relatively longer execution times, indirectly reflecting its higher computational complexity. However, in terms of real-time predictions, most models are completed within 10 s, achieving a nearly-real-time forecast-
ing. In the future, we aim to further enhance the models, reduce the model’s complexity, and improve their prediction accuracy.

Table 8. The Time Complexity of the Model.

<table>
<thead>
<tr>
<th>Model</th>
<th>Average Accuracy</th>
<th>Training Time/s</th>
<th>Testing Time/s</th>
<th>Actual Simulation Time/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logistic Regression</td>
<td>0.61</td>
<td>60.1</td>
<td>0.08567643165588379</td>
<td>0.000561516165111566</td>
</tr>
<tr>
<td>Bayesian Classifier</td>
<td>0.67</td>
<td>13.29</td>
<td>0.3382411003112793</td>
<td>0.003650665283203125</td>
</tr>
<tr>
<td>KNN</td>
<td>0.70</td>
<td>8511.59</td>
<td>1841.9281747341156</td>
<td>30.335577487945557</td>
</tr>
<tr>
<td>Decision Tree</td>
<td>0.92</td>
<td>92.37</td>
<td>0.1147933062866211</td>
<td>0.014016151428222656</td>
</tr>
<tr>
<td>Random Forest</td>
<td>0.85</td>
<td>740.73</td>
<td>6.163089036941528</td>
<td>0.02435851097106936</td>
</tr>
<tr>
<td>Random Forest with Oversampling</td>
<td>0.95</td>
<td>1203.10</td>
<td>7.163089036941528</td>
<td>0.10506129264831543</td>
</tr>
</tbody>
</table>

As depicted in Figure 9, the AUC value of the random forest curve approaches 0.99, which is the highest among the presented models. This signifies that, in terms of the AUC evaluation metric, the random forest model excels in the classification prediction performance.

Figure 9. Illustrates the AUC curves for various models.

Based on the analysis results presented above, data for the model input from 5 PM to 6 PM on a certain day are randomly selected. As is shown in Figure 10, each red dot represents a crime event and includes attributes such as the time, location, and likelihood of the event occurring. The data input to the model every hour is updated based on information collected by various sensors, enabling real-time crime predictions. Target points are extracted from the prediction results. These target points form the basis for multi-agent patrolling, which will be elaborated on in the subsequent chapters.

Figure 10. Visualization of the predictive results.
4. Multi-Drone Response Based on Crime Prediction

In this section, we will discuss the cruise response to the extracted target points. As depicted in Figure 8, each red point represents a target point for drone patrol. Consequently, the challenge of a group intelligence response can be viewed as a multi-target response problem. As the number of targets increases and the coverage area expands, issues such as prolonged response times and complex algorithmic calculations arise. This hampers the identification of an optimal response allocation scheme. To address these challenges, this paper proposes a drone response strategy based on target clustering, coupled with an enhanced genetic algorithm for simulating patrol responses.

4.1. Problem Description

The multi-drone cruise response strategy can be described as follows:

Given N predicted crime points as targets, employ the k-means algorithm to cluster the targets in the response area into M clusters. Consequently, the task of accessing N target points from M centers can be treated as a multi-center drone task allocation problem. Each center can dispatch a maximum of \( K_m \) drones \( (m = 1, 2, \ldots, M) \), and any drone associated with the centroid of a target can access that target point. The objective is to design a rational drone dispatch plan to minimize the drone flight paths while satisfying the following constraints:

1. Each target point must be accessed by only one drone, i.e., multiple drones cannot pass through a single target point simultaneously;
2. Drones must return to their original centers after visiting the target points, with the determined center as the starting point.

For the purpose of simplifying the model complexity, let the target points be encoded as \{1, 2, 3, 4, 5, \ldots, N\}, and the center codes as \{N + 1, N + 2, N + 3, \ldots, N + M\}. Define the variable \( x_{ij}^{mk} \) to indicate whether drone \( k \) from the center \( m \) travels from target point \( i \) to target point \( j \). If no travel occurs, \( x_{ij}^{mk} \) is assigned a value of 0; otherwise, \( x_{ij}^{mk} \) is assigned a value of 1, as shown in Equation (3):

\[
x_{ij}^{mk} = \begin{cases} 1 & \text{if travel occurs} \\ 0 & \text{otherwise} \end{cases}
\]

The driving distance is denoted as totalDist, as shown in Equation (4):

\[
totalDist = \sum_{i=1}^{N+M} \sum_{j=1}^{N+M} \sum_{m=1}^{M} \sum_{k=1}^{K_m} d_{ij} x_{ij}^{mk}.
\]

The objective function aims to minimize the cost, where the cost is defined as the total distance multiplied by the distance weight, as expressed in Equation (5). The weights represent the probability of a crime event occurring at each target point.

\[
mintotalCoat = totalDist \times Weight
\]

Among the constraints (6)–(9), Equation (6) stipulates that the starting and ending points of a drone’s cruise must be its associated center. Equations (7) and (8) denote that each target point should be visited by only one drone. Equation (9) ensures that drones cannot travel from one center to another center.

\[
\sum_{j=1}^{N+M} x_{ij}^{mk} = \sum_{i=1}^{N} x_{ij}^{mk} \leq 1, i = m \in \{N + 1, N + 2, \ldots, N + M\}, k \in \{1, 2, \ldots, K_m\}
\]

\[
\sum_{i=1}^{N+M} \sum_{m=1}^{M} \sum_{k=1}^{K_m} x_{ij}^{mk} = 1, i \in \{1, 2, \ldots, N\}
\]

\[
\sum_{i=1}^{N+M} \sum_{m=1}^{M} \sum_{k=1}^{K_m} x_{ij}^{mk} = 1, j \in \{1, 2, \ldots, N\}
\]

\[
\sum_{i=N+1}^{N+M} x_{ij}^{mk} = \sum_{j=N+1}^{N+M} x_{ij}^{mk} = 0, i = m \in \{N + 1, N + 2, \ldots, N + M\}, k \in \{1, 2, \ldots, K_m\}
\]
4.2. Cruise Response Model
4.2.1. Target Selection Strategy

Initially, the k-means algorithm is employed to cluster targets within the response area based on the principle of minimizing distances. This is illustrated in Figure 11, where the right-side red section depicts the clustering of the target set into M clusters, each denoted as Cluster, (where i = 1, 2, 3, . . ., M). The left-side blue section represents the grouping of Unmanned Aerial Vehicles (UAVs), where drones are divided into M groups according to the same principle of an equal number of clusters and groups, each denoted as Groupi (where i = 1, 2, 3, . . ., M). Subsequently, cluster allocation is performed.

Subsequently, following the strategy depicted in Figure 11, the predicted target points are clustered, resulting in the configuration in Figure 12. In this representation, red cross markers denote the centroids of each cluster, and the surrounding points of the same color signify multiple targets within each respective cluster.

Figure 11. Depiction of the process of target clustering.

Figure 12. Display of the results of target clustering. Different colored dots represent different target clusters, and red cross symbols represent cluster centroids.
Finally, an exemplary cruise response cluster is chosen from the clusters, as is illustrated in Figure 12. The cluster selected by the red circle is extracted, encompassing a total of 46 target points. However, it is important to note that some selected targets might appear duplicated due to multiple occurrences at certain locations. Yet, for the purpose of drone cruising, a single localization suffices, allowing for the removal of duplicates while still meeting the cruising criteria. This refinement leads to a reduced set of 36 unique target points, as demonstrated in Figure 13. The same procedure can be applied to other clusters, enabling concurrent responses.

Figure 13. The selected target outcomes.

4.2.2. Cruise Algorithm

This paper employs an enhanced genetic algorithm for the drone cruise response, as is depicted in Figure 14. The figure outlines the fundamental steps of the genetic algorithm process.

After finalizing the algorithm, improvements were made to the crossover operation. Unlike the traditional crossover approach, this paper introduces an enhanced crossover operation which is depicted in Figure 15. The specific steps are outlined as follows:

1. Set a mutation probability $p$.
2. For each parent chromosome in the population:
   a. Randomly select a crossover point, $G_1$, within the chromosome. Let us say $G_1 = 34$.
   b. Generate a random decimal number, $R$, between 0 and 1.
   c. If $R < p$, go to Step 3. Otherwise, proceed to Step 4.
3. If $R < p$ (mutation occurs):
   a. Randomly select another point, $G_2$, from the same individual’s chromosome.
   b. Invert the segment between $G_1$ and $G_2$.
4. If $R \geq p$ (no mutation):
   a. Select another individual, Parent B, randomly from the population.
   b. Locate $G_1 = 34$ within Parent B’s chromosome and identify the point before it as $G_3$.
   c. Invert the segment between $G_1$ and $G_3$ in the original parent chromosome.
5. Repeat the above steps for all parent chromosomes to generate the offspring population.
3. If $R < p$ (mutation occurs):
   a. Randomly select another point, G2, from the same individual’s chromosome.
   b. Invert the segment between G1 and G2.
4. If $R \geq p$ (no mutation):
   a. Select another individual, Parent B, randomly from the population.
   b. Locate G1 = 34 within Parent B’s chromosome and identify the point before it as G3.
   c. Invert the segment between G1 and G3 in the original parent chromosome.
5. Repeat the above steps for all parent chromosomes to generate the offspring population.

![Diagram](image_url)

Figure 15. Illustrations of the Enhanced Genetic Operations.

The cross-operator enhancement Algorithm 2 pseudocode is as follows:

Algorithm 2: CrossoverIndividual (individual0, individual1, crossoverRate)

```plaintext
INPUT: individual0, individual1, CrossoverRate
OUTPUT: newIndividual
1  D ← Decision variable dimension;
2  if rand > crossoverRate
3     \[r0, r1 \leftarrow \text{Generate two random number between} \ 1 \sim D\]
4   else
5     \[r0 \leftarrow \text{Generate a random number between} \ 1 \sim D\]
6     gene ← individual0[r0] /* Get r0th gene from individual0
7     index ← find(individual1, gene) /* Get the location of gene from individual1
8     if index < D
9       index++ /* right location
10  else
11     index ← — /* left location
12     gene ← individual1[index] /* A gene adjacent to a previous gene
13     r1 ← find(individual0, gene) /* Get the location of gene from individual0
14     if r0 > r1
15       swap(r0, r1)
16     newIndividual ← individual0 /* Get the location of gene from individual0
17     newIndividual[r0:r1] = individual0[r1:r0] /* Reverse the elements from r0 to r1
18     return newIndividual /* return new Individual
```

4.2.3. Simulation Results

For the purpose of conducting a collaborative intelligence response, the clusters extracted from Figure 13 are selected in this study. A total of 36 target regions are present within these clusters, and each target is sequentially numbered with 0 to 35. The experiment involves four drones, and the outcomes are presented in Table 9. Each row represents the response route of an individual drone, starting from point 0, returning to point 0, and ensuring that each target point is visited only once.
Table 9. Drone Cruise Waypoints.

<table>
<thead>
<tr>
<th>Drone</th>
<th>Waypoints</th>
</tr>
</thead>
<tbody>
<tr>
<td>Drone1</td>
<td>0-10-30-2-34-12-8-20-6-5-24-22-31-25-0</td>
</tr>
<tr>
<td>Drone2</td>
<td>0-21-3-23-19-29-17-0</td>
</tr>
<tr>
<td>Drone3</td>
<td>0-15-11-4-13-14-18-0</td>
</tr>
<tr>
<td>Drone4</td>
<td>0-1-26-27-33-9-7-28-16-35-0</td>
</tr>
</tbody>
</table>

As is shown in Figure 16a, the cruise trajectories from the experimental simulation are illustrated. The four distinct colors correspond to the four individual drones, and their trajectories neither overlap nor revisit the target points. This approach ensures an efficient enhancement of response efficiency. Furthermore, Figure 16b depicts the population evolution curve. The objective function gradually converges to its optimal value as the iteration count increases. By the time it reaches 400 iterations, it demonstrates a near-complete convergence.

**Figure 16.** Cruise results. (a) Cruise trajectories of UAVs, where different colors represent different UAVs; (b) Population evolution curve, which shows the convergence of population evolution.

In addition, as is shown in Table 10, we also use the Ant Colony Optimization (ACO) algorithm for comparison. It is evident that the algorithm employed in this study yielded notably favorable results. An equal number of drones being considered, the algorithm demonstrates fewer iterations and shorter flight distances compared to the ACO algorithm. These findings underscore the effectiveness of the approach employed in the current study, showcasing its potential for optimizing drone routing and responses in scenarios with similar parameters.

Table 10. Comparative Simulation Results of Different Algorithms.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Flight Distance</th>
<th>UAV</th>
<th>Evolution Generations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Improved Genetic Algorithm (GA)</td>
<td>83.760 km</td>
<td>4</td>
<td>326</td>
</tr>
<tr>
<td>ACO</td>
<td>86.748 km</td>
<td>4</td>
<td>714</td>
</tr>
</tbody>
</table>

This article proposes a crowdsourced response method based on crime event predictions. Firstly, the Section 3 of the article predicts crime hotspots using a random forest
algorithm based on oversampling. Secondly, a drone response strategy based on target clustering is designed in the Section 4 of the article. This strategy clusters, segments, and extracts the predicted crime hotspots, addressing the issue of high computational complexity in the multi-target response. Finally, an improved genetic algorithm is combined for a patrol response to obtain the optimal response allocation scheme. The specific research contributions are as follows:

1. **High Accuracy:** This method achieves a high accuracy of up to 95% in both the prediction and response, indicating its effectiveness in crime event predictions.
2. **Prediction Granularity:** The method can provide predictions on an hourly basis, enabling real-time prediction and warning responses, which are highly valuable for urban public safety management.
3. **Intelligent Response Strategy:** By using target clustering in the drone response strategy, responses to multiple targets can be effectively handled to reduce computational complexity and enhance response efficiency.
4. **Genetic Algorithm Optimization:** Combining genetic algorithms for optimizing patrol responses allows for finding the best response allocation scheme, reducing the number of iterations, and further improving the response effectiveness.
5. **Providing Public Safety Insights:** While perfect accuracy cannot be achieved, this method can offer valuable insights that contribute to urban public safety governance and help reduce crime rates.

This article has yielded positive results to realize real-time crime prediction and alert responses. Additionally, leveraging deep learning and artificial intelligence techniques can improve the prediction algorithm, making it more adaptive and refined. This can help in better recognizing new crime patterns and trends, as well as enabling timely decision-making. Moreover, the predictions and patrol alerts presented in the article offer valuable insights for urban public safety governance. However, the article only considers a limited set of factors, and thus, it cannot achieve complete accuracy. To obtain more accurate prediction results, it is essential to identify additional crime attributes beyond the ones currently considered. So far, the method has been trained using specific attributes, but it should explore more factors to enhance its accuracy.

**Author Contributions:** Conceptualization, Y.P. and F.T.; methodology, Y.P.; software, F.T.; validation, F.T. and C.W.; formal analysis, C.W.; resources, Y.P.; data curation, F.T.; writing—original draft preparation, F.T.; writing—review and editing, C.W.; visualization, F.T.; supervision, Y.P.; project administration, C.W.; funding acquisition, Y.P. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work is supported in part by the National Natural Science Foundation of China under the grant of NO. 62102431, the National University of Defense Technology under the grant of NO. ZK21-32, and the Science and Technology on Information Systems Engineering Laboratory under the grant of NO. 6142101220209.

**Data Availability Statement:** The dataset pertaining to the City and County of Denver crime, available at https://www.denvergov.org/opendata/dataset/city-and-county-of-denver-crime, is publicly accessible.

**Acknowledgments:** The authors thank the editor, the anonymous reviewers, and all Intelligent Instrument Laboratory members for their insightful comments and feedback. The authors thank Yan Pan at the National University of Defense Technology for giving them suggestions.

**Conflicts of Interest:** The authors declare no conflict of interest.

**References**


38. Cilimkovic, M. Neural Networks and Back Propagation Algorithm; Institute of Technology Blanchardstown: Dublin, Ireland, 2015; Volume 15.

39. Wei, Z.; Zhao, X. Multi-UAVs Cooperative Reconnaissance Task Allocation under Heterogeneous Target Values. IEEE Access 2022, 10, 70955–70963. [CrossRef]


Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.