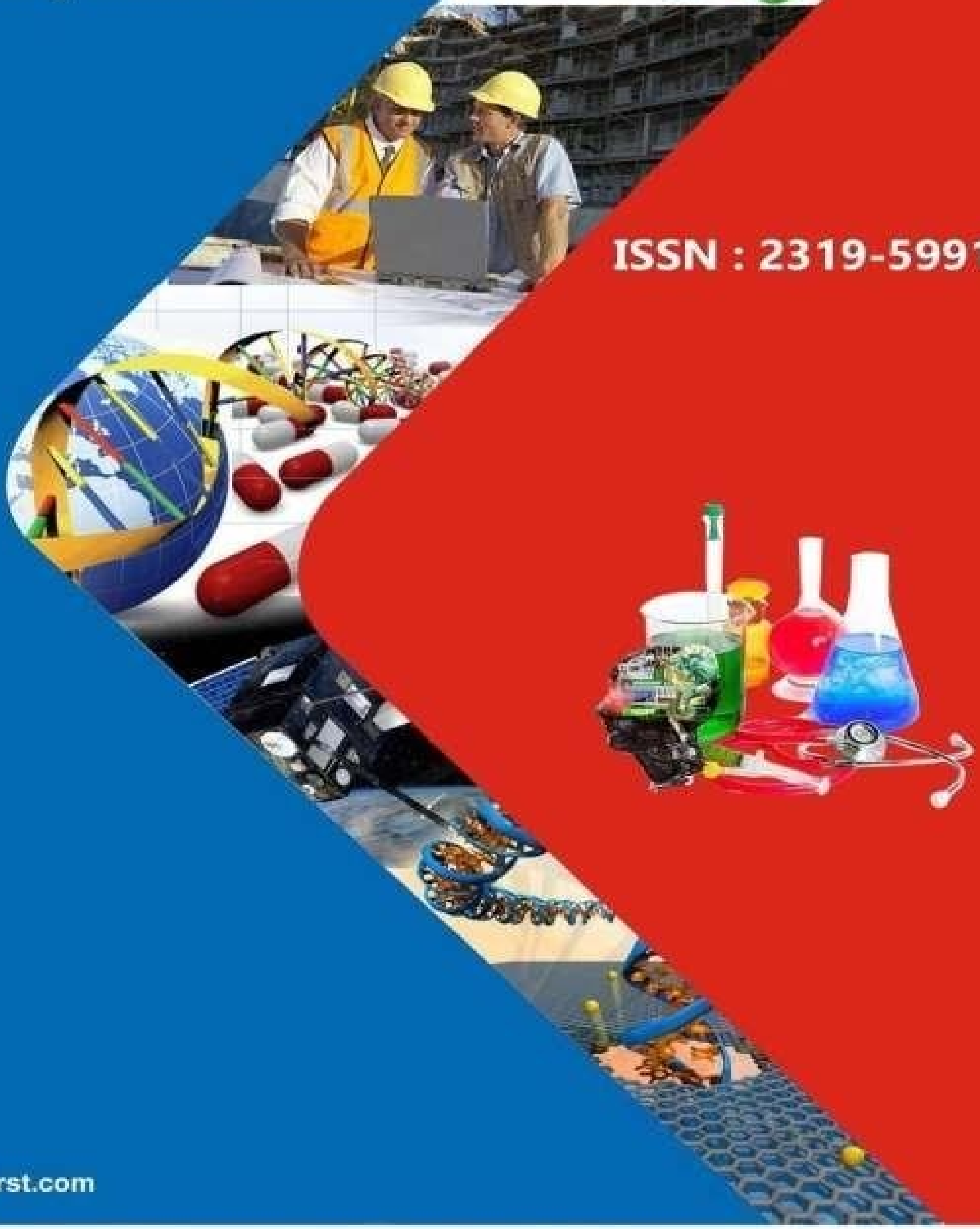
IJERST

International Journal of
Engineering Research and Science & Technology

ISSN : 2319-5991



www.ijerst.com

Email: editor@ijerst.com or editor.ijerst@gmail.com

https://doi.org/10.62643/ijerst.2024.v20.i3.pp250-262

ISSN 2319-5991 www.ijerst.com

Vol. 20, Issue 3, 2024

PROPOUNDING FIRST ARTIFICIAL INTELLIGENCE APPROACH FOR PREDICTING ROBBERY BEHAVIOUR POTENTIAL IN AN INDOOR SECURITY CAMERA

[1] Mr. V. Ranadheer Reddy,[2] Puppala Chaturya,[3] Patil Shruthi,[4] Ponnam Sudha
[1]Assistant Professor,[234]Students
Department Of CSE
Malla Reddy Engineering College for Women

ABSTRACT

For video surveillance systems to stop incidents and safeguard assets, crime prediction is necessary. In this regard, our paper presents the first artificial intelligence method for indoor camera Robbery Behavior Potential (RBP) identification and prediction. Three detection modules—the head cover, crowd, and loitering detection modules—are the foundation of our approach, which enables us to take prompt action and stop robberies. Using our collected, manually annotated dataset, the YOLOV5 model is retrained for the first two modules. Additionally, we provide a brand-new definition for the DeepSORT-based loitering detection module. A fuzzy inference computer converts expert information into rules before making a final determination about the likelihood of a heist. This is difficult because the thief uses a different technique, the security camera is angled differently, and the video pictures have poor quality.Using actual video surveillance footage, we successfully completed our experiment and obtained an F1-score of 0.537. Thus, we develop threshold value for RBP to assess video pictures as a robbery detection issue and compare experimentally with other similar research. Assuming this, the experimental findings clearly reflect an F1-score of 0.607, indicating that the suggested technique outperforms other robbery detection methods in terms of identifying robberies. We firmly think that by anticipating and averting robbery incidents, the suggested method's implementation might reduce the harm caused by robberies at a security camera control center. However, a human operator's situational awareness improves and additional cameras may be controlled.

I. INTRODUCTION

These days, surveillance cameras are often installed in a variety of locations, including houses, banks, shops, and airports, in order to improve public safety and deter crime. Alternatively, by examining these films and trying to identify the offender, it is possible to determine the date, time, and location of the incident as well as the identity of the perpetrator. A person is required to monitor the recordings from behind the scenes and identify any anomalies that may occur. However, the individual becomes exhausted since an anomaly occurs extremely seldom, and if it does, sometimes he is unaware that it has occurred. Stated otherwise, he loses the oddity [1]. Moreover, the process of detecting anomalies is grounded on human intuition that is acquired over time. On the other side, further issues with non-automated crime prediction and detection systems that rely on monitoring surveillance recordings include the skill level of the individual for recognizing indicators of crime occurrence and the expense of hiring him.

Certain visual cues need to be retrieved using machine learning and deep learning methods in order to automate anomaly detection [2], [3]. Certain properties for various anomaly classes [4], such as vandalism [5], violence detection [6], and robbery [7], may be helpful for these algorithms to operate more efficiently.Reducing the amount of devastation

https://doi.org/10.62643/ijerst.2024.v20.i3.pp250-262

ISSN 2319-5991 www.ijerst.com

Vol. 20, Issue 3, 2024

is predicting the place and timing of the act. However, security personnel are also there on schedule. In one such trial, cops in Santa Cruz, California, get daily crime projections every morning. They use this predicting to guide them as they monitor certain areas. According to a Santa Cruz representative, during the first six months of the trial, thirteen wrongdoers were apprehended in the designated regions [8].Predictive policing is important for federal funding and security systems since it reduces crime and costs money, as shown by paper [8], [9], and [10]. The likelihood of victimization makes violent crimes more risky, and according to a Seattle Police Department (SPD) study from 2021 in Washington, USA, the rate of victimization rose by 20% [11]. Robbery is one of the top five most frequent crimes in the US, using data obtained from the FBI's Uniform Crime Reporting System (FBI-UCR) [12]. One of the numerous reasons security cameras are installed is to detect robberies. According to the Oxford Dictionary, robbery is the act of stealing or trying to take any property by force, threat, or weaponry [13]. It is distinguished from other types of theft like shoplifting, pickpocketing, or burglary by its inherently violent nature [14], [15]. Robbery is usually classified as a crime in jurisdictions, even when many lower forms of theft are only penalized as misdemeanors. The time and place of the robbery, whether it is armed or not, the kind of weapon used, and the degree of force used are all factors that criminologists consider when classifying various sorts of robberies. Thus, street and commercial robberies are two examples of classic scissor [16].Robberies on the street often occur in impoverished, congested areas without closed-circuit television systems (CCTV). There are two ways that commercial robberies happen: either the perpetrator dresses up as a client and enters the store, hiding his face with a mask or helmet, or he just walks up and starts pulling a gun and threatening the staff. The other is when violent criminals arrive, usually in groups, and most often cover their faces or heads [17]. Both kinds of business robberies took place in indoor locations that are often equipped with CCTVs to enable the discovery of criminals or even the anticipation of such crimes. Furthermore, criminals who carry a knife or weapon often threaten people physically. However, a large force is more likely to be used against criminals who are unarmed or who are carrying any kind of stick [16], [17]. Hence, forceful commercial robberies with or without weapons result in harm, suffering, and even fatalities.

Predicting the behavior of commercial robberies, whether by humans, machines, or a mix of these, is crucial in averting their occurrence and associated risks [18].

Generally speaking, there are techniques for automating the detection or prediction of crimes that are based on taking various criminal situations and applying them to various sectors. However, no technique has been able to foresee the possibility of robbery conduct. Consequently, an algorithm for RBP prediction in video pictures has to be developed. It is evident that in order to make predictions, characteristics and evidence from surveillance footage must be extracted. Investigating the possibility of robbery behavior in video footage is necessary to do this. The circumstances of each location chosen for robbing and the diverse cultures of the nations involved cause scenarios of robbery occurrence to change from one context to another [19]. As a result, assurance does not accompany strong feature extraction.

Even though there are many different robbery occurrence scenarios, scenario-based techniques [20], [21] allow for the consideration of a common scenario with key aspects for commercial robbery videos. In particular, someone or anything selecting a poorly frequented location, often hiding their face or head with a helmet, mask, glasses, or

https://doi.org/10.62643/ijerst.2024.v20.i3.pp250-262

ISSN 2319-5991 www.ijerst.com

Vol. 20, Issue 3, 2024

any other article of clothing to avoid recognition, and waiting for a chance to brandish a weapon, threaten, or use force. This situation perfectly aligns with the concept of the first kind of commercial robbery conduct as well as the expertise of an individual [17], [22]. We take into consideration similar traits present in the majority of robbery instances under three modules, namely head cover, crowd, and loitering detection, in order to construct a system based on this common scenario abstracted from other situations inferred from robbery footage. An inference machine is required to conclude on the RBP once these characteristics have been extracted for module implementation. For possible derivation, the conclusion method ought to be as competent as human decision making. Fuzzy measurement suggests that fuzzy set theory's capacity to imitate human inference [23] allows experience to be expressed as fuzzy rules, which in turn helps with complicated decision diagnostics and reasoning [24], [25]. However, deep learning techniques lack this flexibility and may not be up to par with the subtleties and fluctuations of ambiguous data [25]. For these reasons, this study proposes a fuzzy inference engine.

In summary, our paper's primary contributions are listed below:

1. The suggested algorithm is predicated on a cutting-edge technique that forecasts RBP and guards against damages brought on by its occurrence inside. As far as we are aware, this is the first study on the prediction of robbery behavior, based on three primary modules: loitering, crowd, and head cover recognition.

2. A dataset with and without head covers has been manually tagged and produced for our system. We total the outcomes of the two states that the head cover detecting module reports in order to count the crowd. The technique outperforms security video limitations including poor quality and single camera footage.

3. Our concept of the lingering point offers a fresh approach to the loitering computation. The length of time each individual spends loitering has been calculated using the Deep Simple Online Real-time Tracking (Deep SORT) algorithm in relation to the tracking techniques. Each one has been given a point by evaluating the quantity of loitering that was discovered using the Euclidean distance computation.

4. Our algorithm's main innovation is the use of a fuzzy inference engine with phases for fuzzification and defuzzification as well as optimal rules. The data from these three modules were examined using an inference machine and an expert's understanding of robbery conduct.

The remaining text is formatted as follows: Section II examines some of the literature that is relevant to our study, such as studies that discuss our modules and the prediction or detection of suspicious behavior. In Section III, the suggested method is explained together with the principles of RBP prediction, along with the suggested modules and how they improve YOLOV5 to produce low-resolution video pictures. Section IV presents and discusses the experimental outcomes. The last part wraps up the findings and outlines next projects.

1.1 PURPOSE:

The purpose of proposing an artificial intelligence (AI) approach for predicting robbery behavior potential in an indoor security camera system can be multi-faceted, including:

1. Enhancing Security and Safety:

- **Proactive Measures:** By predicting robbery behavior, AI can enable security personnel to take proactive measures to prevent crimes before they happen, enhancing overall safety.
- **Real-Time Alerts:** AI systems can provide real-time alerts to security staff, enabling quick responses to potential threats.

2. Improving Surveillance Efficiency:

https://doi.org/10.62643/ijerst.2024.v20.i3.pp250-262
ISSN 2319-5991 www.ijerst.com
Vol. 20, Issue 3, 2024

- **Automated Monitoring:** AI can continuously monitor security footage without fatigue, ensuring constant vigilance.

- **Resource Allocation:** Security teams can focus their efforts on verified threats, optimizing resource allocation and reducing the need for continuous manual monitoring.

3. Advanced Threat Detection:

- **Behavioral Analysis:** AI can analyze patterns of behavior that might indicate potential criminal activity, which may be missed by human observers.

-**Pattern Recognition:** By recognizing suspicious activities based on historical data and patterns, AI can identify potential threats more accurately.

4. Data-Driven Decision Making:

-**Predictive Analytics:** AI can leverage historical data to predict future incidents, helping organizations to make informed decisions on security protocols and preventive measures.

-**Trend Analysis:** Understanding trends in robbery behaviors can help in formulating strategies to mitigate risks.

5. Cost Reduction:

- **Lowering Operational Costs:** By automating surveillance and reducing the need for a large number of security personnel, organizations can lower their operational costs.

- **Preventing Losses:** Preventing robberies can significantly reduce financial losses due to theft and damage.

6. Technological Advancement:

- **Innovation in Security:** Implementing AI for robbery prediction can position an organization as a leader in adopting innovative security technologies.

-**Integration with Existing Systems:** AI can enhance existing security systems by integrating with current surveillance infrastructure, improving overall system capabilities.

7. Legal and Compliance:

-**Adherence to Standards**: Using AI can help organizations comply with security standards and regulations that require advanced monitoring and threat detection capabilities.

-**Evidence Collection:** AI can aid in collecting and analyzing evidence in the event of an incident, supporting legal and investigative processes.

Overall, the primary purpose is to leverage AI's capabilities to create a more secure, efficient, and cost-effective surveillance system that can predict and prevent robbery behaviors, thus safeguarding people and property.

1.2 EXISTING SYSTEM:

Things that don't fit the ordinary, happen seldom, or seem strange are called anomalies. Anomalies, defined as any behavior that differs from the norm, constitute criminal conduct [2]. One may make the case that the broad installation of CCTV is a response to the increase in crimes committed in public places. Crimes could be foreseen if suspicious activity is detected. In order to make predictions, one needs information that is vague, inaccurate, and unclear [26]. Indoor RBP prediction is the main emphasis of our proposed technique. The proposed algorithm has to be able to distinguish between loitering, crowds, and head cover in order to identify robbery. Important to our approach is the provision of a generic RBP prediction framework, a topic that has not been covered by any previous research. Several relevant publications pertaining to the detection or prediction of suspicious behavior, criminality, loitering, and head coverings will be included in this section.

Elhamod and Levine [27] proposed a semantics-based suspicious behavior detection system dependent on object tracking using blob matching with color histograms and spatial information, with the goal of updating objects intended in each frame. The similarity between objects and blobs is defined by computing the value of the histogram's

https://doi.org/10.62643/ijerst.2024.v20.i3.pp250-262

ISSN 2319-5991 www.ijerst.com

Vol. 20, Issue 3, 2024

intersection and comparing it to a predefined threshold. After that, it assigns the correct classes, such as objects for inanimate things and people for animated ones. Semantically identifying behaviors is achieved by calculating their 3D motion properties and documenting it in the past. The following suspicious behaviors have been discovered: fighting through the computation of merge, split, and simultaneous movement of a blob's centroid; abandoned luggage through the use of background subtraction methods; fainting through the comparison of a person's assumed 2D and actual 3D foot and head coordinates; and finally, loitering through the aggregation of a person's presence time in an area.

Ishikawa and Zin introduced a typical automated approach to troublesome pedestrian identification via lingering detection. [18]. A suspicious person allegedly spends a lot of time strolling, pausing, and circling the location with an increasing number of route alterations, according to [18]. His acceleration fluctuates drastically, and his distance value is greater than the normal person. Applying these features requires counting the occurrence of block numbers that indicate the person's foot position in each of the 25 blocks that comprise the video frame [18]. The pedestrian was viewed with suspicion if the frequency surpassed the threshold. In order to determine changes in direction, it calculates the angles of motion. By calculating the changing distance and acceleration, all the required properties are retrieved. Finally, suspicious-looking pedestrians are identified using an aggregated set of data from each step via a decision fusion process.

Rajapakshe et al. [28] suggested a two-pronged E-police system that includes crime prediction and a video surveillance monitoring system.[28] classify suspicious behaviors (such as acts of aggression or vandalism) as either normal or abnormal using human activity recognition tools. Thanks to feature extraction from Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), they are able to detect suspicious individuals hiding their faces. To foretell when and where criminal behavior will occur, the crime prediction technique of [28] use public information resources and classification algorithms like as SVM and Decision tree.

Arroyo et al. [22] proposed a method for expert real-time identification of suspicious activity in shopping malls. Foreground detection is accomplished by the use of an image segmentation and background removal approach. Then, a blob fusion approach is used to collect the blobs from each split part so that people may be recognized. A tracking algorithm is used using a unique two-step procedure: (a) managing occlusion using support vector machines (SVMs), and (b) identifying and tracking humans with a Kalman filter. The obtained trajectories are then used to study human behavior. When a person tries to escape or if there are too many people trying to enter, the entrance or existence alert goes off, and trajectory analysis picks it up. Furthermore, areas inside containing expensive things chosen by human security guards are marked as risk zones. People's routes and the amount of time spent in various regions are used to measure their loitering behavior. According to security experts, their technology is designed to trigger a warning if the timeframe goes above 30 seconds. The installation of a camera on the cash counter was done to ensure its safety. If there is someone loitering at the cash desk and no one from the shop is there, an alarm will go off. Additionally, a realistic dataset is assessed using data collected from many cameras placed at the entrance, within, and at the register of a business.

Disadvantages:

Existing machine learning models for Android malware detection need to be able to correctly understand big and complicated datasets, which is a major consideration due to the

https://doi.org/10.62643/ijerst.2024.v20.i3.pp250-262

ISSN 2319-5991 www.ijerst.com

Vol. 20, Issue 3, 2024

complexity of the data.

• Access to data: In order for machine learning models to provide reliable predictions, they often need massive datasets. The reliability of the model could be compromised if there is a lack of data in enough amounts.

• Mislabeled data: Current ML models can only learn as much as the data used to train them. The accuracy of the model's predictions is dependent on the accuracy of the data labels.

1.3 Proposed System

The suggested algorithm can anticipate RBP and protect interior spaces from harm since it is based on a new technique.This is the first study that we are aware of that uses three primary modules—head cover, crowd, and loitering detection—to anticipate robbery behavior.

2. Our system is ready to go using a dataset that has been manually annotated with two states: with and without a head cover. We add the two states supplied by the head cover detection module to get the crowd count. In low-resolution or single-camera surveillance footage, the approach is dominant.

3. Our recently established loitering point represents a fresh approach to the computation of loitering. The quantity of loitering for each individual has been calculated using a Deep Simple Online Real-time Tracking (DeepSORT) algorithm in relation to the tracking techniques. A point has been allocated to each one by assessing the amount of loitering that was acquired using the Euclidean distance computation.

4. Our algorithm's main innovation is a fuzzy inference engine that utilizes optimal rules, fuzzification, and defuzzification processes. Analyzed outcomes from these three modules using inference machine and expert human knowledge of robbery behavior.

Advantages:

We provide an intelligent AAMD-OELAC method for detecting Android malware that includes data pretreatment, ensemble learning, and hyper parameter optimization based on HPO. As far as we are aware, the AAMD-OELAC method was never published.

• Detect Android malware using an ensemble learning-based classification technique that includes the LS-SVM, KELM, and RRVFLN models.

• The HPO algorithm and ensemble learning method work together to make Android malware detection more accurate. Using a combination of classifiers and optimization techniques, the model is able to detect dangerous patterns and behaviors in Android apps.

II. LITERATURE SURVEY

"Real-world anomaly detection in surveillance videos." Sultani, Waqas, Chen Chen, and Mubarak Shah.

Surveillance videos are able to capture a variety of realistic anomalies. In this paper, we propose to learn anomalies by exploiting both normal and anomalous videos. To avoid annotating the anomalous segments or clips in training videos, which is very time consuming, we propose to learn anomaly through the deep multiple instance ranking framework by leveraging weakly labeled training videos, i.e. the training labels (anomalous or normal) are at video-level instead of clip-level. In our approach, we consider normal and anomalous videos as bags and video segments as instances in multiple instance learning (MIL), and automatically learn a deep anomaly ranking model that predicts high anomaly scores for anomalous video segments. Furthermore, we introduce sparsity and temporal smoothness constraints in the ranking loss function to better localize anomaly during training. We also introduce a new large-scale first of its kind dataset of 128 hours of videos. It consists of 1900 long and untrimmed real-world surveillance videos, with 13 realistic anomalies such as fighting, road accident, burglary, robbery, etc. as well as normal activities. This dataset can be used for two tasks. First, general anomaly detection considering all anomalies in one group and all

https://doi.org/10.62643/ijerst.2024.v20.i3.pp250-262

ISSN 2319-5991 www.ijerst.com

Vol. 20, Issue 3, 2024

normal activities in another group. Second, for recognizing each of 13 anomalous activities. Our experimental results show that our MIL method for anomaly detection achieves significant improvement on anomaly detection performance as compared to the state-of-the-art approaches. We provide the results of several recent deep learning baselines on anomalous activity recognition. The low recognition performance of these baselines reveals that our dataset is very challenging and opens more opportunities for future work. The dataset is available at: http://crcv.ucf.edu/projects/real-world.

"A survey on deep learning techniques for video anomaly detection." Suarez, Jessie James P., and Prospero C. Naval Jr.

Researchers have been looking at the issue of anomaly identification in videos for over ten years. Researchers are intrigued by this field because of its broad range of potential applications. This has led to a large variety of methods, ranging from those based on statistics to those based on machine learning, being suggested throughout the years. Although there have been several surveys covering this ground, the purpose of this work is to summarize the most current developments in anomaly detection as they pertain to Deep Learning. Many areas of AI have found success with Deep Learning's use, including computer vision, NLP, and many more. This overview, on the other hand, is concerned with the ways in which Deep Learning has advanced and shed light on video anomaly detection. In this study, we classify the various Deep Learning methods according to their intended use. Furthermore, it delves into the frequently used datasets and assessment measures. The next step is a conversation that combines all of the current methods in order to point the way and suggest potential research directions.

"Deep learning for intelligent video analysis." Mei, Tao, and Cha Zhang.

In both academia and business, big data applications are taking up the majority of desk space. Video feeds from CCTV cameras play an equally significant role as other types of big data, such as data from social media, sensors, agriculture, medicine, and space research. Unstructured large data is greatly aided by surveillance films. Everywhere safety is a top priority, CCTV cameras are set up. Seems like a lot of work and time to do manual surveillance. The phrase "security" may have a variety of meanings depending on the setting, such as "theft detection," "violence detection," "explosion chances," etc. The word "security" is used to describe a wide range of unusual occurrences in busy public spaces. Because it requires participation from several people, violence detection is one of these challenging tasks. There are a number of practical limitations that make it very difficult to analyze crowd film scenes for anomalous or aberrant behavior. Beginning with object identification and moving on to action recognition, crowd analysis, and ultimately, violence detection in a crowd context, the study contains a thorough examination. The survey primarily focuses on articles that use deep learning techniques. The models and algorithms used by different deep learning approaches are compared. Applying deep learning algorithms to accurately recognize the number of participants, the events that took place inside a huge crowd, regardless of the weather, is the primary goal of this survey. In this paper, we'll look at the technology that allows different crowd video analysis approaches to use deep learning. Additionally, we take real-time processing into account, a crucial but as-yet-unexplored aspect of this area. There aren't many ways to deal with all these problems at once. We list and outline the problems with the current approaches. Additional guidance is provided for overcoming the highlighted challenges in the future. Science Direct, IEEE Xplore, and the ACM digital library are cited in the poll.

https://doi.org/10.62643/ijerst.2024.v20.i3.pp250-262
ISSN 2319-5991 www.ijerst.com
Vol. 20, Issue 3, 2024

"Real-time auto-matic detection of vandalism behavior in video sequences." Ghazal, Mohammed, Carlos Vázquez, and Aishy Amer.

A technique for the real-time identification of vandalism in video sequences is presented in this research. Without employing object identification or relying on a single camera, the suggested technique may identify vandalism by robustly extracting a chain of high-level events that led to it. A pay phone or sign is an example of an item that can be considered vandalized if it were to enter the scene and alter it without permission inside a designated region. Results from both online and offline tests demonstrate that the suggested strategy is effective in identifying instances of vandalism or graffiti in CCTV footage.

"Automatic video-based human motion analyzer for consumer surveillance system."
Lao, Weilun, Jungong Han, and Peter Hn De With.

As video-analysis methods continue to advance, consumer-grade automated, low-cost video surveillance is slowly making its way into the market. In addition to facilitating control over home-entrance and equipment-usage functions, video surveillance may help keep people secure in the house. In this research, we investigate a versatile paradigm for the semantic interpretation of human actions in consumer-grade monocular surveillance footage. When human-body modeling and trajectory estimation work together, it becomes much easier to do semantic analysis on human actions and events in video sequences. The inclusion of a 3-D reconstruction technique for scene comprehension is another feature; this allows for the analysis of human behaviors from diverse perspectives. In all, there are four stages of processing in the framework: (1) preprocessing, which encompasses tasks like multiple-person detection and background modeling; (2) object-based, which estimates trajectories and classifies postures; (3) event-based, which conducts semantic analysis; and (4) visualization, which includes tasks like camera calibration and 3-D scene reconstruction. The evaluation of our suggested framework revealed its high quality and efficacy, with an accuracy of 86% for posture classification and 90% for events. It achieves near real-time performance, averaging 6-8 frames/second.

"Automated visualsurveillance in realistic scenarios." Shah, Mubarak, Omar Javed, and Khurram Shafique.

Here we introduce Knight, an automated monitoring system that has found use in many real-world contexts, from police enforcement to railway security. Additionally, we review Knight's performance in unconstrained contexts, talk about the difficulties of creating surveillance systems, and show various techniques that Knight uses to overcome these difficulties.

III. SYSTEM DESIGN

Flow Chart: Remote User

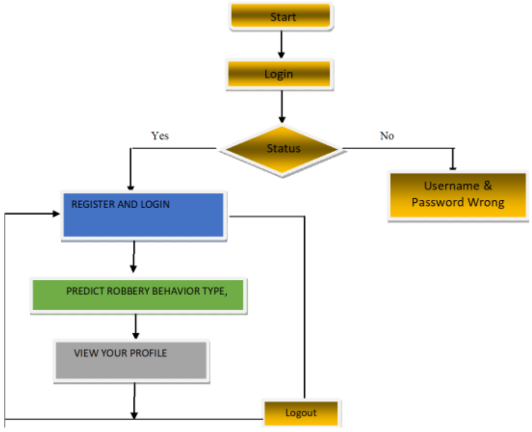


fig:2.1

- **Flow Chart:** Service Provider

https://doi.org/10.62643/ijerst.2024.v20.i3.pp250-262

ISSN 2319-5991 www.ijerst.com

Vol. 20, Issue 3, 2024

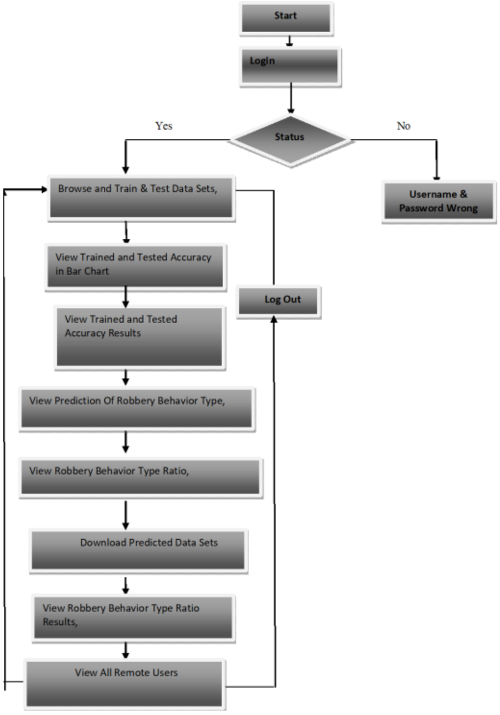
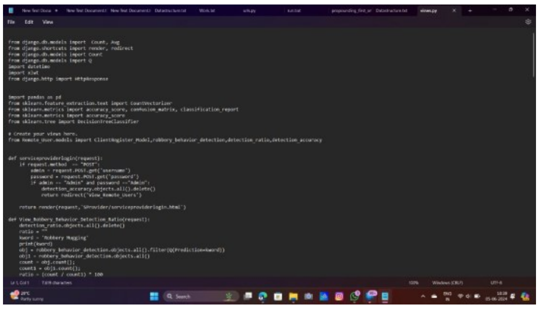
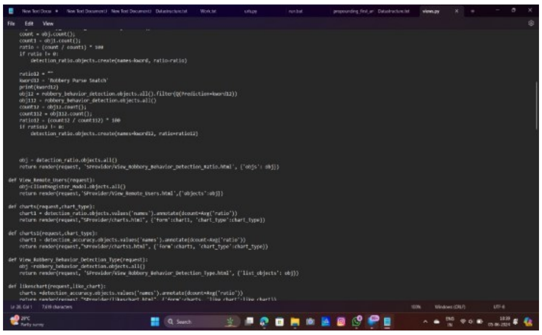

Start
Login
Yes
Status
No
Browse and Train & Test Data Sets,
Username & Password Wrong
View Trained and Tested Accuracy in Bar Chart
Log Out
View Trained and Tested Accuracy Results
View Prediction Of Robbery Behavior Type,
View Robbery Behavior Type Ratio,
Download Predicted Data Sets
View Robbery Behavior Type Ratio Results,
View All Remote Users

fig:2.2

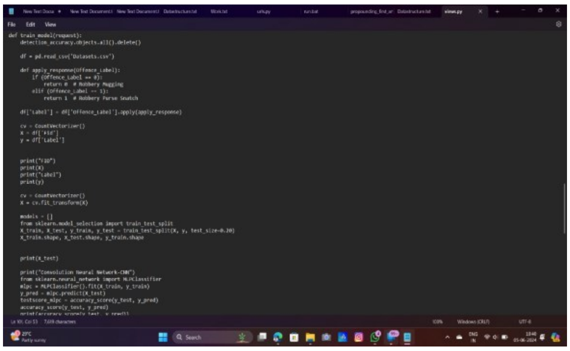
IV. SCREENSHOT CODE



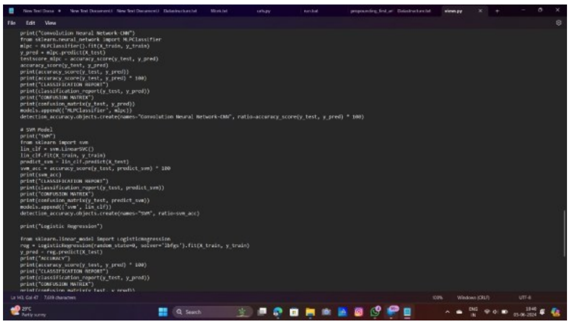
```
from django.db.models import  Count, Avg
from django.shortcuts import render, redirect
from django.db.models import Count
from django.db.models import Q
import datetime
import xlwt
from django.http import HttpResponse


import pandas as pd
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
from sklearn.metrics import accuracy_score
from sklearn.tree import DecisionTreeClassifier

# Create your views here.
from Remote_User.models import ClientRegister_Model,robbery_behavior_detection,detection_ratio,detection_accuracy


def serviceproviderlogin(request):
    if request.method  == "POST":
        admin = request.POST.get('username')
        password = request.POST.get('password')
        if admin == "Admin" and password =="Admin":
            detection_accuracy.objects.all().delete()
            return redirect('View_Remote_Users')

    return render(request,'SProvider/serviceproviderlogin.html')

def View_Robbery_Behavior_Detection_Ratio(request):
    detection_ratio.objects.all().delete()
    ratio = ""
    kword = 'Robbery Mugging'
    print(kword)
    obj = robbery_behavior_detection.objects.all().filter(Q(Prediction=kword))
    obj1 = robbery_behavior_detection.objects.all()
    count = obj.count();
    count1 = obj1.count();
    ratio = (count / count1) * 100
```



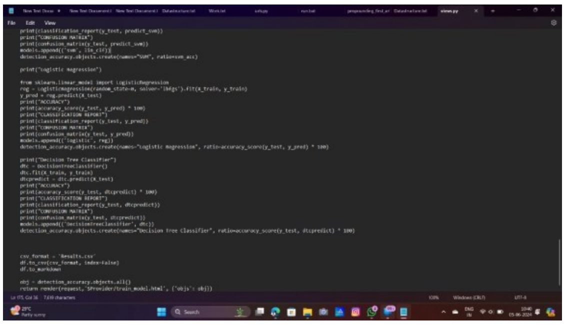
```
    count = obj.count();
    count1 = obj1.count();
    ratio = (count / count1) * 100
    if ratio != 0:
        detection_ratio.objects.create(names=kword, ratio=ratio)

    ratio12 = ""
    kword12 = 'Robbery Purse Snatch'
    print(kword12)
    obj12 = robbery_behavior_detection.objects.all().filter(Q(Prediction=kword12))
    obj112 = robbery_behavior_detection.objects.all()
    count12 = obj12.count();
    count112 = obj112.count();
    ratio12 = (count12 / count112) * 100
    if ratio12 != 0:
        detection_ratio.objects.create(names=kword12, ratio=ratio12)



    obj = detection_ratio.objects.all()
    return render(request, 'SProvider/View_Robbery_Behavior_Detection_Ratio.html', {'objs': obj})

def View_Remote_Users(request):
    obj=ClientRegister_Model.objects.all()
    return render(request,'SProvider/View_Remote_Users.html',{'objects':obj})

def charts(request,chart_type):
    chart1 = detection_ratio.objects.values('names').annotate(dcount=Avg('ratio'))
    return render(request,"SProvider/charts.html", {'form':chart1, 'chart_type':chart_type})

def charts1(request,chart_type):
    chart1 = detection_accuracy.objects.values('names').annotate(dcount=Avg('ratio'))
    return render(request,"SProvider/charts1.html", {'form':chart1, 'chart_type':chart_type})

def View_Robbery_Behavior_Detection_Type(request):
    obj =robbery_behavior_detection.objects.all()
    return render(request, 'SProvider/View_Robbery_Behavior_Detection_Type.html', {'list_objects': obj})

def likeschart(request,like_chart):
    charts =detection_accuracy.objects.values('names').annotate(dcount=Avg('ratio'))
```



```
def train_model(request):
    detection_accuracy.objects.all().delete()

    df = pd.read_csv('Datasets.csv')

    def apply_response(Offence_Label):
        if (Offence_Label == 0):
            return 0  # Robbery Mugging
        elif (Offence_Label == 1):
            return 1  # Robbery Purse Snatch

    df['Label'] = df['Offence_Label'].apply(apply_response)

    cv = CountVectorizer()
    X = df['Fid']
    y = df['Label']


    print("FID")
    print(X)
    print("Label")
    print(y)

    cv = CountVectorizer()
    X = cv.fit_transform(X)

    models = []
    from sklearn.model_selection import train_test_split
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20)
    X_train.shape, X_test.shape, y_train.shape


    print(X_test)

    print("Convolution Neural Network-CNN")
    from sklearn.neural_network import MLPClassifier
    mlpc = MLPClassifier().fit(X_train, y_train)
    y_pred = mlpc.predict(X_test)
    testscore_mlpc = accuracy_score(y_test, y_pred)
    accuracy_score(y_test, y_pred)
```

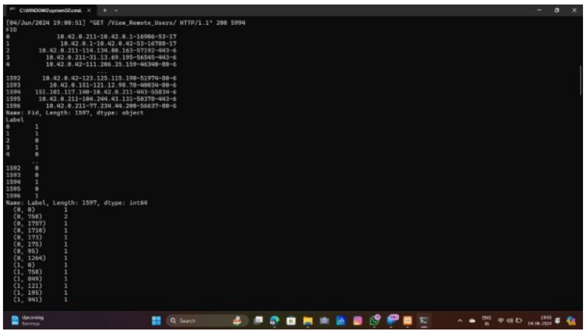


```
    print("Convolution Neural Network-CNN")
    from sklearn.neural_network import MLPClassifier
    mlpc = MLPClassifier().fit(X_train, y_train)
    y_pred = mlpc.predict(X_test)
    testscore_mlpc = accuracy_score(y_test, y_pred)
    accuracy_score(y_test, y_pred)
    print(accuracy_score(y_test, y_pred))
    print(accuracy_score(y_test, y_pred) * 100)
    print("CLASSIFICATION REPORT")
    print(classification_report(y_test, y_pred))
    print("CONFUSION MATRIX")
    print(confusion_matrix(y_test, y_pred))
    models.append(('MLPClassifier', mlpc))
    detection_accuracy.objects.create(names="Convolution Neural Network-CNN", ratio=accuracy_score(y_test, y_pred) * 100)

    # SVM Model
    print("SVM")
    from sklearn import svm
    lin_clf = svm.LinearSVC()
    lin_clf.fit(X_train, y_train)
    predict_svm = lin_clf.predict(X_test)
    svm_acc = accuracy_score(y_test, predict_svm) * 100
    print(svm_acc)
    print("CLASSIFICATION REPORT")
    print(classification_report(y_test, predict_svm))
    print("CONFUSION MATRIX")
    print(confusion_matrix(y_test, predict_svm))
    models.append(('svm', lin_clf))
    detection_accuracy.objects.create(names="SVM", ratio=svm_acc)

    print("Logistic Regression")

    from sklearn.linear_model import LogisticRegression
    reg = LogisticRegression(random_state=0, solver='lbfgs').fit(X_train, y_train)
    y_pred = reg.predict(X_test)
    print("ACCURACY")
    print(accuracy_score(y_test, y_pred) * 100)
    print("CLASSIFICATION REPORT")
    print(classification_report(y_test, y_pred))
    print("CONFUSION MATRIX")
```

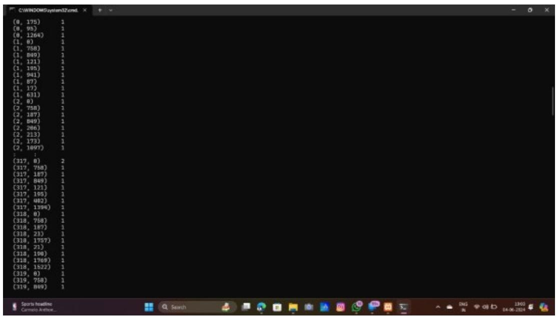


```
    print(classification_report(y_test, predict_svm))
    print("CONFUSION MATRIX")
    print(confusion_matrix(y_test, predict_svm))
    models.append(('svm', lin_clf))
    detection_accuracy.objects.create(names="SVM", ratio=svm_acc)

    print("Logistic Regression")

    from sklearn.linear_model import LogisticRegression
    reg = LogisticRegression(random_state=0, solver='lbfgs').fit(X_train, y_train)
    y_pred = reg.predict(X_test)
    print("ACCURACY")
    print(accuracy_score(y_test, y_pred) * 100)
    print("CLASSIFICATION REPORT")
    print(classification_report(y_test, y_pred))
    print("CONFUSION MATRIX")
    print(confusion_matrix(y_test, y_pred))
    models.append(('logistic', reg))
    detection_accuracy.objects.create(names="Logistic Regression", ratio=accuracy_score(y_test, y_pred) * 100)

    print("Decision Tree Classifier")
    dtc = DecisionTreeClassifier()
    dtc.fit(X_train, y_train)
    dtcpredict = dtc.predict(X_test)
    print("ACCURACY")
    print(accuracy_score(y_test, dtcpredict) * 100)
    print("CLASSIFICATION REPORT")
    print(classification_report(y_test, dtcpredict))
    print("CONFUSION MATRIX")
    print(confusion_matrix(y_test, dtcpredict))
    models.append(('DecisionTreeClassifier', dtc))
    detection_accuracy.objects.create(names="Decision Tree Classifier", ratio=accuracy_score(y_test, dtcpredict) * 100)



    csv_format = 'Results.csv'
    df.to_csv(csv_format, index=False)
    df.to_markdown

    obj = detection_accuracy.objects.all()
    return render(request,'SProvider/train_model.html', {'objs': obj})
```

5.2.2.3 Result:



```
[04/Jun/2024 19:00:51] "GET /View_Remote_Users/ HTTP/1.1" 200 5994
FID
0               10.42.0.211-10.42.0.1-16906-53-17
1                10.42.0.1-10.42.0.42-53-16788-17
2          10.42.0.211-114.134.80.163-57192-443-6
3            10.42.0.211-31.13.69.195-56545-443-6
4            10.42.0.42-111.206.25.159-46340-80-6
                              ...
1592        10.42.0.42-123.125.115.190-51974-80-6
1593          10.42.0.151-121.12.98.78-40034-80-6
1594     151.101.117.140-10.42.0.211-443-55834-6
1595       10.42.0.211-104.244.43.131-50378-443-6
1596         10.42.0.211-77.234.44.200-56637-80-6
Name: Fid, Length: 1597, dtype: object
Label
0       1
1       1
2       0
3       1
4       0
       ..
1592    0
1593    0
1594    1
1595    0
1596    1
Name: Label, Length: 1597, dtype: int64
  (0, 0)        1
  (0, 758)      2
  (0, 1757)     1
  (0, 1710)     1
  (0, 173)      1
  (0, 175)      1
  (0, 95)       1
  (0, 1264)     1
  (1, 0)        1
  (1, 758)      1
  (1, 849)      1
  (1, 121)      1
  (1, 195)      1
  (1, 941)      1
```

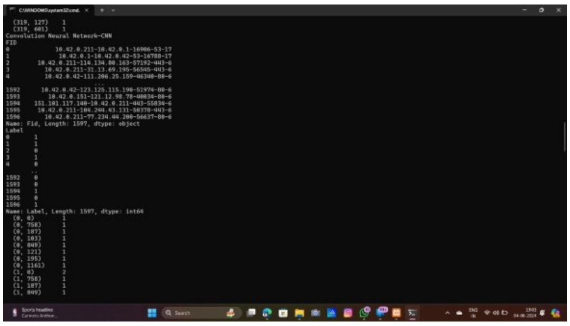


```
  (0, 175)      1
  (0, 95)       1
  (0, 1264)     1
  (1, 0)        1
  (1, 758)      1
  (1, 849)      1
  (1, 121)      1
  (1, 195)      1
  (1, 941)      1
  (1, 87)       1
  (1, 17)       1
  (1, 631)      1
  (2, 0)        1
  (2, 758)      1
  (2, 187)      1
  (2, 849)      1
  (2, 206)      1
  (2, 213)      1
  (2, 173)      1
  (2, 1097)     1
  :       :
  (317, 0)      2
  (317, 758)    1
  (317, 187)    1
  (317, 849)    1
  (317, 121)    1
  (317, 195)    1
  (317, 402)    1
  (317, 1394)   1
  (318, 0)      1
  (318, 758)    1
  (318, 187)    1
  (318, 23)     1
  (318, 1757)   1
  (318, 21)     1
  (318, 190)    1
  (318, 1769)   1
  (318, 1522)   1
  (319, 0)      1
  (319, 758)    1
  (319, 849)    1
```

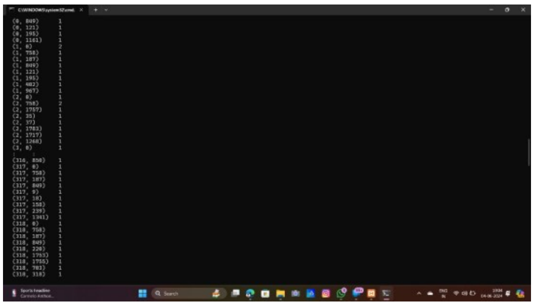
https://doi.org/10.62643/ijerst.2024.v20.i3.pp250-262

ISSN 2319-5991 www.ijerst.com

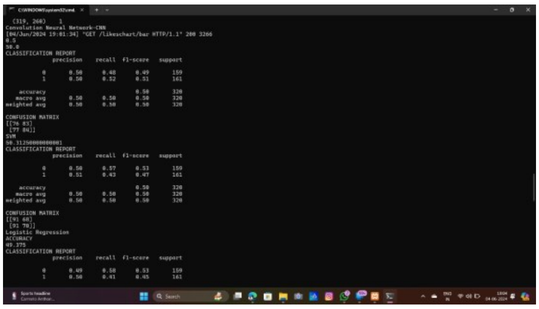
Vol. 20, Issue 3, 2024



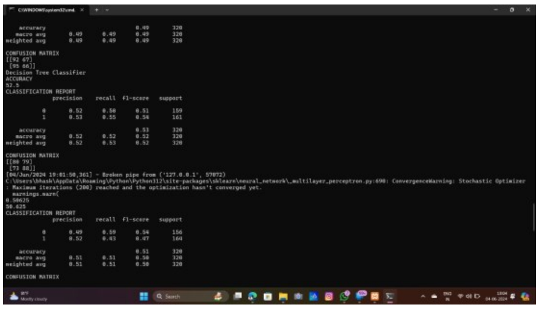
```
  (319, 127)    1
  (319, 601)    1
Convolution Neural Network-CNN
FID
0              10.42.0.211-10.42.0.1-16906-53-17
1               10.42.0.1-10.42.0.42-53-16788-17
2        10.42.0.211-114.134.80.163-57192-443-6
3           10.42.0.211-31.13.69.195-56545-443-6
4           10.42.0.42-111.206.25.159-46340-80-6
                                ...
1592       10.42.0.42-123.125.115.190-51974-80-6
1593          10.42.0.151-121.12.98.78-48034-80-6
1594     151.101.117.140-10.42.0.211-443-55834-6
1595      10.42.0.211-106.204.43.131-58378-443-6
1596         10.42.0.211-77.234.44.200-56637-80-6
Name: Fid, Length: 1597, dtype: object
Label
0       1
1       1
2       0
3       1
4       0
       ..
1592    0
1593    0
1594    1
1595    0
1596    1
Name: Label, Length: 1597, dtype: int64
  (0, 0)        1
  (0, 758)      1
  (0, 187)      1
  (0, 183)      1
  (0, 849)      1
  (0, 121)      1
  (0, 195)      1
  (0, 1161)     1
  (1, 0)        2
  (1, 758)      1
  (1, 187)      1
  (1, 849)      1
```



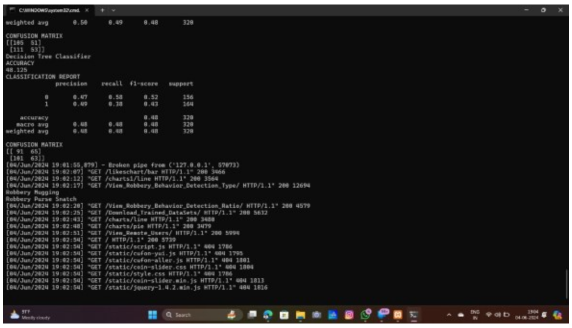
```
  (0, 849)      1
  (0, 121)      1
  (0, 195)      1
  (0, 1161)     1
  (1, 0)        2
  (1, 758)      1
  (1, 187)      1
  (1, 849)      1
  (1, 121)      1
  (1, 195)      1
  (1, 482)      1
  (1, 967)      1
  (2, 0)        1
  (2, 758)      2
  (2, 1757)     1
  (2, 35)       1
  (2, 37)       1
  (2, 1783)     1
  (2, 1717)     1
  (2, 1268)     1
  (3, 0)        1
  :     :
  (316, 850)    1
  (317, 0)      1
  (317, 758)    1
  (317, 187)    1
  (317, 849)    1
  (317, 9)      1
  (317, 18)     1
  (317, 158)    1
  (317, 239)    1
  (317, 1341)   1
  (318, 0)      1
  (318, 758)    1
  (318, 187)    1
  (318, 849)    1
  (318, 220)    1
  (318, 1753)   1
  (318, 1755)   1
  (318, 783)    1
  (318, 318)    1
```



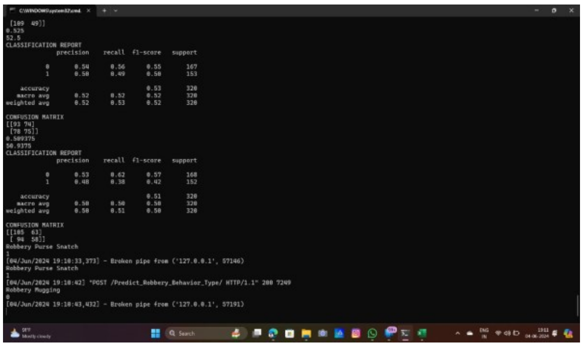
```
  (319, 260)    1
Convolution Neural Network-CNN
[04/Jun/2024 19:01:34] "GET /likeschart/bar HTTP/1.1" 200 3266
0.5
50.0
CLASSIFICATION REPORT
              precision    recall  f1-score   support

           0       0.50      0.48      0.49       159
           1       0.50      0.52      0.51       161

    accuracy                           0.50       320
   macro avg       0.50      0.50      0.50       320
weighted avg       0.50      0.50      0.50       320

CONFUSION MATRIX
[[76 83]
 [77 84]]
SVM
50.31250000000001
CLASSIFICATION REPORT
              precision    recall  f1-score   support

           0       0.50      0.57      0.53       159
           1       0.51      0.43      0.47       161

    accuracy                           0.50       320
   macro avg       0.50      0.50      0.50       320
weighted avg       0.50      0.50      0.50       320

CONFUSION MATRIX
[[91 68]
 [91 70]]
Logistic Regression
ACCURACY
49.375
CLASSIFICATION REPORT
              precision    recall  f1-score   support

           0       0.49      0.58      0.53       159
           1       0.50      0.41      0.45       161
```



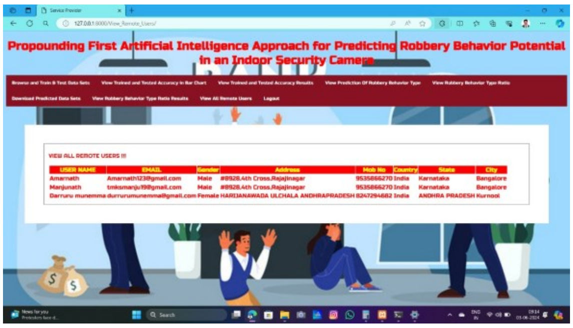
```
    accuracy                           0.49       320
   macro avg       0.49      0.49      0.49       320
weighted avg       0.49      0.49      0.49       320

CONFUSION MATRIX
[[92 67]
 [95 66]]
Decision Tree Classifier
ACCURACY
52.5
CLASSIFICATION REPORT
              precision    recall  f1-score   support

           0       0.52      0.50      0.51       159
           1       0.53      0.55      0.54       161

    accuracy                           0.53       320
   macro avg       0.52      0.52      0.52       320
weighted avg       0.52      0.53      0.52       320

CONFUSION MATRIX
[[80 79]
 [73 88]]
[04/Jun/2024 19:01:50,361] - Broken pipe from ('127.0.0.1', 57072)
C:\Users\bhask\AppData\Roaming\Python\Python312\site-packages\sklearn\neural_network\_multilayer_perceptron.py:690: ConvergenceWarning: Stochastic Optimizer
: Maximum iterations (200) reached and the optimization hasn't converged yet.
  warnings.warn(
0.50625
50.625
CLASSIFICATION REPORT
              precision    recall  f1-score   support

           0       0.49      0.59      0.54       156
           1       0.52      0.43      0.47       164

    accuracy                           0.51       320
   macro avg       0.51      0.51      0.50       320
weighted avg       0.51      0.51      0.50       320

CONFUSION MATRIX
```



```
weighted avg       0.50      0.49      0.48       320

CONFUSION MATRIX
[[105  51]
 [111  53]]
Decision Tree Classifier
ACCURACY
48.125
CLASSIFICATION REPORT
              precision    recall  f1-score   support

           0       0.47      0.58      0.52       156
           1       0.49      0.38      0.43       164

    accuracy                           0.48       320
   macro avg       0.48      0.48      0.48       320
weighted avg       0.48      0.48      0.48       320

CONFUSION MATRIX
[[ 91  65]
 [101  63]]
[04/Jun/2024 19:01:55,879] - Broken pipe from ('127.0.0.1', 57073)
[04/Jun/2024 19:02:07] "GET /likeschart/bar HTTP/1.1" 200 3466
[04/Jun/2024 19:02:12] "GET /charts1/line HTTP/1.1" 200 3564
[04/Jun/2024 19:02:17] "GET /View_Robbery_Behavior_Detection_Type/ HTTP/1.1" 200 12694
Robbery Mugging
Robbery Purse Snatch
[04/Jun/2024 19:02:20] "GET /View_Robbery_Behavior_Detection_Ratio/ HTTP/1.1" 200 4579
[04/Jun/2024 19:02:25] "GET /Download_Trained_DataSets/ HTTP/1.1" 200 5632
[04/Jun/2024 19:02:43] "GET /charts/line HTTP/1.1" 200 3488
[04/Jun/2024 19:02:48] "GET /charts/pie HTTP/1.1" 200 3479
[04/Jun/2024 19:02:51] "GET /View_Remote_Users/ HTTP/1.1" 200 5994
[04/Jun/2024 19:02:54] "GET / HTTP/1.1" 200 5739
[04/Jun/2024 19:02:54] "GET /static/script.js HTTP/1.1" 404 1786
[04/Jun/2024 19:02:54] "GET /static/cufon-yui.js HTTP/1.1" 404 1795
[04/Jun/2024 19:02:54] "GET /static/cufon-aller.js HTTP/1.1" 404 1801
[04/Jun/2024 19:02:54] "GET /static/coin-slider.css HTTP/1.1" 404 1804
[04/Jun/2024 19:02:54] "GET /static/style.css HTTP/1.1" 404 1786
[04/Jun/2024 19:02:54] "GET /static/coin-slider.min.js HTTP/1.1" 404 1813
[04/Jun/2024 19:02:54] "GET /static/jquery-1.4.2.min.js HTTP/1.1" 404 1816
```



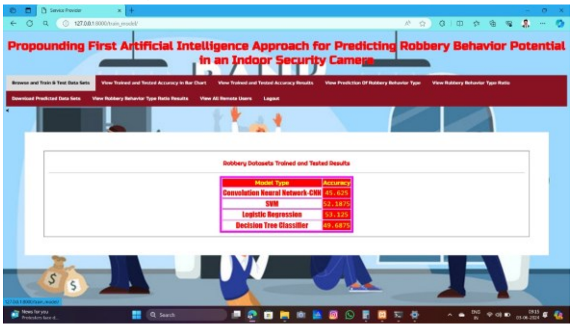
```
 [109  49]]
0.525
52.5
CLASSIFICATION REPORT
              precision    recall  f1-score   support

           0       0.54      0.56      0.55       167
           1       0.50      0.49      0.50       153

    accuracy                           0.53       320
   macro avg       0.52      0.52      0.52       320
weighted avg       0.52      0.53      0.52       320

CONFUSION MATRIX
[[93 74]
 [78 75]]
0.509375
50.9375
CLASSIFICATION REPORT
              precision    recall  f1-score   support

           0       0.53      0.62      0.57       168
           1       0.48      0.38      0.42       152

    accuracy                           0.51       320
   macro avg       0.50      0.50      0.50       320
weighted avg       0.50      0.51      0.50       320

CONFUSION MATRIX
[[105  63]
 [ 94  58]]
Robbery Purse Snatch
1
[04/Jun/2024 19:10:33,373] - Broken pipe from ('127.0.0.1', 57146)
Robbery Purse Snatch
1
[04/Jun/2024 19:10:42] "POST /Predict_Robbery_Behavior_Type/ HTTP/1.1" 200 7249
Robbery Mugging
0
[04/Jun/2024 19:10:43,432] - Broken pipe from ('127.0.0.1', 57191)
```


Service Provider
127.0.0.1:8000/View_Remote_Users/
Propounding First Artificial Intelligence Approach for Predicting Robbery Behavior Potential in an Indoor Security Camera
Browse and Train & Test Data Sets
View Trained and Tested Accuracy in Bar Chart
View Trained and Tested Accuracy Results
View Prediction Of Robbery Behavior Type
View Robbery Behavior Type Ratio
Download Predicted Data Sets
View Robbery Behavior Type Ratio Results
View All Remote Users
Logout

VIEW ALL REMOTE USERS !!!

USER NAME	EMAIL	Gender	Address	Mob No	Country	State	City
Amarnath	Amarnath123@gmail.com	Male	#8928,4th Cross,Rajajinagar	9535866270	India	Karnataka	Bangalore
Manjunath	tmksmanju19@gmail.com	Male	#8928,4th Cross,Rajajinagar	9535866270	India	Karnataka	Bangalore
Darruru munemma	durrurumunemma@gmail.com	Female	HARIJANAWADA ULCHALA ANDHRAPRADESH	8247294682	India	ANDHRA PRADESH	Kurnool

Fig: View all Remote Users


Service Provider
Propounding First Artificial Intelligence Approach for Predicting Robbery Behavior Potential in an Indoor Security Camera
Browse and Train & Test Data Sets
View Trained and Tested Accuracy in Bar Chart
View Trained and Tested Accuracy Results
View Prediction Of Robbery Behavior Type
View Robbery Behavior Type Ratio
Download Predicted Data Sets
View Robbery Behavior Type Ratio Results
View All Remote Users
Logout

Robbery Datasets Trained and Tested Results

Model Type	Accuracy
Convolution Neural Network-CNN	45.625
SVM	52.1875
Logistic Regression	53.125
Decision Tree Classifier	49.6875

Fig: Datasets Trained and Tested Results

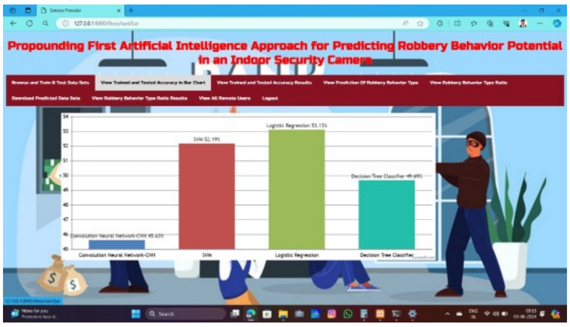

Service Provider
127.0.0.1:8000/likeschart/bar
Propounding First Artificial Intelligence Approach for Predicting Robbery Behavior Potential in an Indoor Security Camera
Browse and Train & Test Data Sets
View Trained and Tested Accuracy in Bar Chart
View Trained and Tested Accuracy Results
View Prediction Of Robbery Behavior Type
View Robbery Behavior Type Ratio
Download Predicted Data Sets
View Robbery Behavior Type Ratio Results
View All Remote Users
Logout
54
53
52
51
50
49
48
47
46
45
Convolution Neural Network-CNN 45.63%
SVM 52.19%
Logistic Regression 53.13%
Decision Tree Classifier 49.69%
Convolution Neural Network-CNN
SVM
Logistic Regression
Decision Tree Classifier

Fig: View Trained and Tested Accuracy in Bar Chart

https://doi.org/10.62643/ijerst.2024.v20.i3.pp250-262

ISSN 2319-5991 www.ijerst.com

Vol. 20, Issue 3, 2024

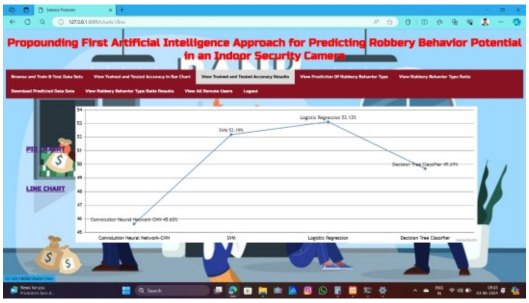
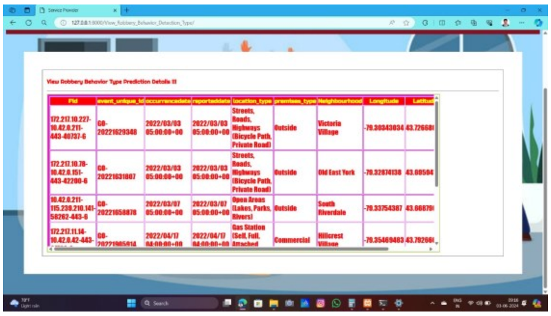
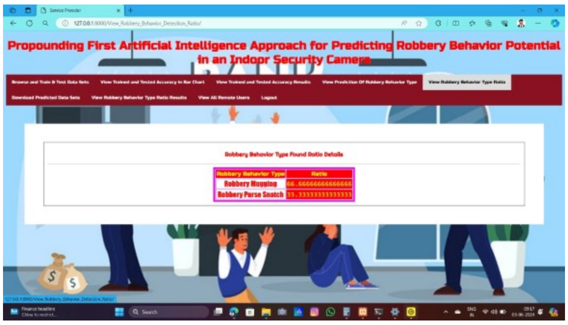

Propounding First Artificial Intelligence Approach for Predicting Robbery Behavior Potential in an Indoor Security Camera
Browse and Train & Test Data Sets
View Trained and Tested Accuracy in Bar Chart
View Trained and Tested Accuracy Results
View Prediction Of Robbery Behavior Type
View Robbery Behavior Type Ratio
Download Predicted Data Sets
View Robbery Behavior Type Ratio Results
View All Remote Users
Logout
PIE CHART
LINE CHART
54
53
52
51
50
49
48
47
46
45
Convolution Neural Network-CNN 45.63%
SVM 52.19%
Logistic Regression 53.13%
Decision Tree Classifier 49.65%
Convolution Neural Network-CNN
SVM
Logistic Regression
Decision Tree Classifier

Fig: View Trained and Tested Accuracy Results


View Robbery Behavior Type Prediction Details !!!

Fid	event_unique_id	occurrencedate	reporteddate	location_type	premises_type	Neighbourhood	Longitude	Latitud
172.217.10.227-10.42.0.211-443-40737-6	GO-20221629348	2022/03/03 05:00:00+00	2022/03/03 05:00:00+00	Streets, Roads, Highways (Bicycle Path, Private Road)	Outside	Victoria Village	-79.30343034	43.72668
172.217.10.78-10.42.0.151-443-42200-6	GO-20221631807	2022/03/03 05:00:00+00	2022/03/03 05:00:00+00	Streets, Roads, Highways (Bicycle Path, Private Road)	Outside	Old East York	-79.32874138	43.69504
10.42.0.211-115.239.210.141-56262-443-6	GO-20221658878	2022/03/07 05:00:00+00	2022/03/07 05:00:00+00	Open Areas (Lakes, Parks, Rivers)	Outside	South Riverdale	-79.33754387	43.66879
172.217.11.14-10.42.0.42-443-	GO-20221905414	2022/04/17 04:00:00+00	2022/04/17 04:00:00+00	Gas Station (Self, Full, Attached	Commercial	Hillcrest Village	-79.35469483	43.79266

Fig: View Robbery Behavior Type Prediction Details


Propounding First Artificial Intelligence Approach for Predicting Robbery Behavior Potential in an Indoor Security Camera
Browse and Train & Test Data Sets
View Trained and Tested Accuracy in Bar Chart
View Trained and Tested Accuracy Results
View Prediction Of Robbery Behavior Type
View Robbery Behavior Type Ratio
Download Predicted Data Sets
View Robbery Behavior Type Ratio Results
View All Remote Users
Logout

Robbery Behavior Type Found Ratio Details

Robbery Behavior Type	Ratio
Robbery Mugging	66.66666666666666
Robbery Purse Snatch	33.33333333333333

Fig: View Robbery Behavior Ratio Details

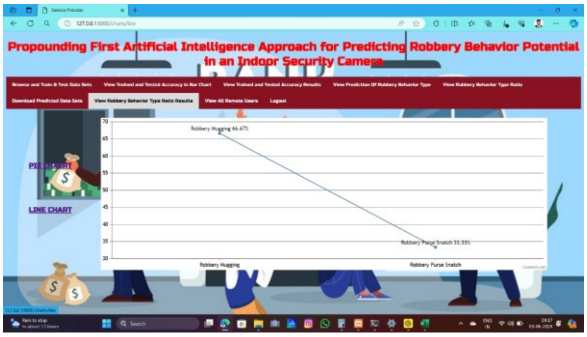
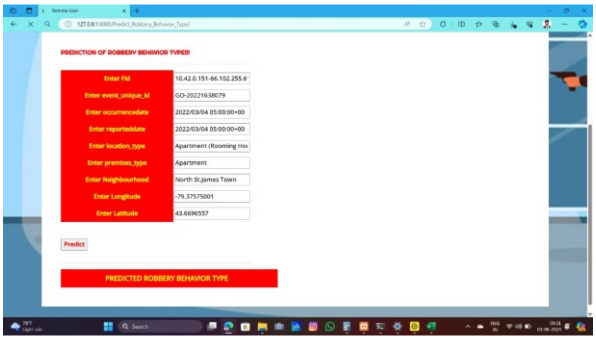

Propounding First Artificial Intelligence Approach for Predicting Robbery Behavior Potential in an Indoor Security Camera
Browse and Train & Test Data Sets
View Trained and Tested Accuracy in Bar Chart
View Trained and Tested Accuracy Results
View Prediction Of Robbery Behavior Type
View Robbery Behavior Type Ratio
Download Predicted Data Sets
View Robbery Behavior Type Ratio Results
View All Remote Users
Logout
PIE CHART
LINE CHART
70
65
60
55
50
45
40
35
30
Robbery Mugging 66.67%
Robbery Purse Snatch 33.33%
Robbery Mugging
Robbery Purse Snatch

Fig: View Robbery Behavior Ratio Results


PREDICTION OF ROBBERY BEHAVIOR TYPE!!!

Enter Fid	10.42.0.151-66.102.255.6
Enter event_unique_id	GO-20221638079
Enter occurrencedate	2022/03/04 05:00:00+00
Enter reporteddate	2022/03/04 05:00:00+00
Enter location_type	Apartment (Rooming Hou
Enter premises_type	Apartment
Enter Neighbourhood	North St.James Town
Enter Longitude	-79.37575001
Enter Latitude	43.6696557

PREDICTED ROBBERY BEHAVIOR TYPE :

Fig: Prediction of Robbery Behavior Type

V. CONCLUSION

In this study, we provide a method for RBP prediction using video surveillance footage. Poor video quality, different camera angles in different places, and different procedures for recording robbery situations are just a few of the issues that might arise with CCTV footage. By taking swift action and addressing these obstacles, we can prevent robberies that are seen on surveillance cameras. Our comprehensive literature review revealed that no RBP forecast has ever been made, which is why this research is being conducted. Preventing robberies is obviously important. After watching many CCTV videos of robberies and listening to expert discussion, we were able to identify commonalities in these crimes. Our goal in analyzing these cases is to find more shared features and to implement a practical approach to RBP prediction. Our research proposes a deep learning-based approach to estimating the probability of a heist using a fuzzy inference system. By gathering suitable datasets of individuals donning and donning non-hats, this strategy retrains the YOLOV5 algorithm. This method based on deep learning is successful in implementing the head cover detection and crowd detection modules. The loitering module, which calculates people's Euclidean traveled distance using the Deep SORT method, is also implemented in this work. The outputs of three modules form the basis of a fuzzy inference machine that averages the robbery potential of each clip and infers it every 10 frames. The proposed method is

https://doi.org/10.62643/ijerst.2024.v20.i3.pp250-262

ISSN 2319-5991 www.ijerst.com

Vol. 20, Issue 3, 2024

implemented in the Robbery category of the UCF-Crime dataset and has an F1-score of 0.537. This proves that our proposed method successfully predicts the probability of a theft in more than 50% of the films.

It follows that the problem of robbery detection becomes our focus instead of prediction. Consequently, we may compare it to other studies that employed the UCF-Crime dataset to investigate anomaly detection, namely robbery detection. Out of all the techniques, the detection method has the greatest F1-score at 0.607. The results show that the scenario-based approach we proposed functions well and is very good at detecting and predicting robbery movements. Our proposed method may be used by any establishment with surveillance cameras that is serious about reducing robbery incidents. They may easily assess the likelihood of a theft by carefully reviewing the real-time video captured by these cameras. However, this person should see the movies all the way to the conclusion to prevent making a mistake. In addition, anybody may privately change our methodology by changing the culturally-sensitive threshold values.

We can improve the F1-score by making loitering detection more accurate. Our long-term research objective is to develop a more effective tracking algorithm for low-resolution video images by improving the Deep SORT method. Accurate person identification and tracking is not possible with low-resolution videos. This is because the Deep SORT algorithm relies on FRR CNN as its detector. So, to retrain YOLOV5 using low-resolution human images, we will change the detection architecture of the Deep SORT algorithm. The planned YOLOV5 will only support low-resolution images as objects.

BIBLIOGRAPHY

[1] W. Sultani, C. Chen, and M. Shah, ‘‘Real-world anomaly detection in surveillance videos,’’ in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6479–6488.

[2] J. James P. Suarez and P. C. Naval Jr., ‘‘A survey on deep learning techniques for video anomaly detection,’’ 2020, *arXiv:2009.14146*.

[3] T. Mei and C. Zhang, ‘‘Deep learning for intelligent video analysis,’’ in *Proc. 25th ACM Int. Conf. Multimedia*, Oct. 2017, pp. 1955–1956.

[4] H. Yan, X. Liu, and R. Hong, ‘‘Image classification via fusing the latent deep CNN feature,’’ in *Proc. Int. Conf. Internet Multimedia Comput. Service*, Aug. 2016, pp. 110–113.

[5] M. Ghazal, C. Vazquez, and A. Amer, ‘‘Real-time automatic detection of vandalism behavior in video sequences,’’ in *Proc. IEEE Int. Conf. Syst.,Man Cybern.*, Oct. 2007, pp. 1056–1060.

[6] I. P. Febin, K. Jayasree, and P. T. Joy, ‘‘Violence detection in videos for an intelligent surveillance system using MoBSIFT and movement filtering algorithm,’’ *Pattern Anal. Appl.*, vol. 23, no. 2, pp. 611–623, May 2020.

[7] W. Lao, J. Han, and P. De With, ‘‘Automatic video-based human motion analyzer for consumer surveillance system,’’ *IEEE Trans. Consum. Electron.*, vol. 55, no. 2, pp. 591–598, May 2009.

[8] A. G. Ferguson, ‘‘Predictive policing and reasonable suspicion,’’ *Emory Law J.*, vol. 62, no. 2, p. 259, 2012.

[9] C. Beck and C. McCue, ‘‘Predictive policing: What can we learn from Wal-Mart and Amazon about fighting crime in a recession?’’ *Police Chief*, [1] W. Sultani, C. Chen, and M. Shah, ‘‘Real-world anomaly detection in surveillance videos,’’ in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6479–6488.

[10] K. J. Bowers and S. D. Johnson, ‘‘Who commits near repeats? A test of the boost explanation,’’ *Western Criminol. Rev.*, vol. 5, no. 3, pp. 12–24, 2004.

https://doi.org/10.62643/ijerst.2024.v20.i3.pp250-262

ISSN 2319-5991 www.ijerst.com

Vol. 20, Issue 3, 2024

[11] Seattle Police Department, ''SPD 2021 year-end crime report,'' Seattle, WA, USA, 2021.[Online].Available:https://www.seattle.gov/documents/Departments/Police/Reports/2021_SPD_CRIME_REPORT_FINAL.pdf

[12] (2019). *FBI.* [Online]. Available: https://ucr.fbi.gov/crime-in-the-u.s/2019/crime-in-the-u.s.-2019/topic-pages/robbery

[13] B. Fawei, J. Z. Pan, M. Kollingbaum, and A. Z.Wyner, ''A semi-automated On tology construction for legal question answering,'' *New Gener. Comput.*,vol. 37, no. 4, pp. 453–478, Dec. 2019.

[14] R. Thompson, ''Understanding theft from the person and robbery of personal property victimisation trends in England and Wales,'' Nottingham Trent Univ., Nottingham, U.K., Tech. Rep. 2010/11, 2014.

[15] P. J. Cook, ''Robbery violence,'' *J. Criminal Law Criminol.*, vol. 78, no. 2, pp. 357–376, 1987.

[16] J. D. McCluskey, ''A comparison of Robbers' use of physical coercion in commercial and street robberies,'' *Crime Delinquency*, vol. 59, no. 3, pp. 419–442, Apr. 2013.

[17] D. F. Luckenbill, ''Patterns of force in robbery,'' *Deviant Behav.*, vol.1, nos. 3–4, pp. 361–378, Apr. 1980.

[18] T. Ishikawa and T. T. Zin, ''A study on detection of suspicious persons for intelligent monitoring system,'' in *Proc. Int. Conf. Big Data Anal. Deep Learn. Appl.* Singapore: Springer, 2018, pp. 292–301.

[19] J. R. Medel and A. Savakis, ''Anomaly detection in video using predictive convolutional long short-term memory networks,'' 2016, *arXiv:1612.00390.*

[20] M. Shah, O. Javed, and K. Shafique, ''Automated visual surveillance in realistic scenarios,'' *IEEE Multimedia Mag.*, vol. 14, no. 1, pp. 30–39, Jan. 2007.

[21] A. Biswas, S. C. Ria, Z. Ferdous, and S. N. Chowdhury, ''Suspicious human-movement detection,'' Ph.D. dissertation, Dept. Comput. Sci. Eng., BRAC Univ., Dhaka, Bangladesh, 2017.

[22] R. Arroyo, J. J. Yebes, L. M. Bergasa, I. G. Daza, and J. Almazán, ''Expert video-surveillance system for real-time detection of suspicious behaviors in shopping malls,'' *Expert Syst. Appl.*, vol. 42, no. 21, pp. 7991–8005, Nov. 2015.

[23] R. Nawaratne, D. Alahakoon, D. De Silva, and X. Yu, ''Spatiotemporal anomaly detection using deep learning for real-time video surveillance,'' *IEEE Trans. Ind. Informat.*, vol. 16, no. 1, pp. 393–402, Jan. 2020.

[24] F. Wu, G. Jin, M. Gao, Z. HE, and Y. Yang, ''Helmet detection based on improved YOLO V3 deep model,'' in *Proc. IEEE 16th Int. Conf. Netw.,Sens. Control (ICNSC)*, May 2019, pp. 363–368.

[25] Y. Liu, X.-K. Wang, W.-H. Hou, H. Liu, and J.-Q. Wang, ''A novel hybrid model combining a fuzzy inference system and a deep learning method for short-term traffic flow prediction,'' *Knowl.-Based Syst.*, vol. 255, Nov. 2022, Art. no. 109760.vol. 76, no. 11, p. 18, 2009.