Criminal Activity Detection from videos under Low Light Condition Using Deep Neural Network

by

Ilham Hoque Nafim
20101375
Somiya Azadi Tonni
20101187
Humira Akter Runa
20301073

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
October 2024

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Ilham Hoque Nafim

20101375

Somiya Azadi Tonni

20101187

Humaira Akter Runa

20301073

Approval

The thesis/project titled "Criminal Activity Detection from videos under Low Light Condition Using Deep Neural Network" submitted by

1. Ilham Hoque Nafim (20101375)
2. Somiya Azadi Tonni (20101187)
3. Humaira Akter Runa (20301073)

Of Summer, 2024 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on October, 2024.

Examining Committee:

Supervisor:
(Member)

Dr. Md. Ashraful Alam

Associate Professor
Department Of Computer Science and Engineering
BRAC University

Co-Supervisor:
(Member)

Md. Tanzim Reza

Senior Lecturer
Department Of Computer Science And Engineering
BRAC University

Program Coordinator:
(Member)

Dr.Md. Golam Rabiul Alam

Professor
Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Dr.Sadia Hamid Kazi

Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

Criminal activity detection from footage, especially in low-light circumstances, offers a considerable problem due to reduced visibility, noise, and detail loss. In this research, we present an approach for detecting criminal actions in low-light videos using deep learning models. The specific difficulty of low-light conditions, which result in limited visibility and noisy data, is handled using modern video pre-processing techniques to improve video quality. Our strategy improves video classification accuracy by using both spatial and temporal information, preserving essential visual signals. We use transfer learning to adapt pre-trained models such as VGG-16, MobileNetV2, NASNetMobile, LSTM and a Conv-LSTM so that they can generalize effectively in low-light settings. The modified UCF Crime dataset simulates low-light situations, and our results illustrate that the suggested technique improves detection accuracy while remaining efficient. This study paves the path for more dependable surveillance systems capable of working in harsh environments. After working through all these models the best performing model we got was the ConvLSTM model where we got an accuracy of 77%.

Keywords: Deep learning, Criminal Activity Detection, Threat detection, Low Light Condition, Human Behavior Analysis, UCF crime, VGG-16, MobilenetV2, CNN, NasNetMobile, Conv-LSTM

Acknowledgement

Firstly, all praise to the Almighty Allah who guided us and blessed us with his blessing to complete our thesis without any major interruption.
Secondly, We want to express our heartfelt gratitude and thanks to our respected Supervisors Dr. Md.Ashraful Alam, sir, and Md.Tanzim Reza sir for their kind support and continuous encouragement throughout the journey of the thesis. We are grateful for their time and effort in nurturing our academic growth.
Finally, we want to extend our heartfelt thanks to our parents for their incredible guidance and support. We truly couldn't have made it this far without them. Their love, encouragement, and prayers have been with us every step of the way, and now we're just about to graduate!

Table of Contents

List of Figures

Chapter 1

Introduction

Criminal activities in poor lighting conditions are a growing concern for law enforcement agencies amidst increasing security challenges and advancements in technology. Many instances of video surveillance exist in public areas; however, the current setup of such systems remains very traditional, considering poor lighting conditions that do not enable proper vision. Most of today's surveillance systems operate in traditional mode, wherein the data is on offline video feeds with no real-time detection. Although traditional methods are employed, most of them utilize either rule-based systems or subjective human decisions. Therefore, the low-light scenario is a difficult condition to exactly detect criminal activities that may point to certain security gaps with delays in responses. Additionally, the active method of monitoring considerably reduces the potential to allow the triggering of real-time alerts, proactive notifications, and continuous monitoring in critical situations where the visibility is low[26]. Besides, this application of human operators reduces the analysis to inconsistencies and biases, hence less effectiveness in the detection of crimes[7]. These limitations signify the development of advanced and intelligent systems that are able to detect abnormalities in real-time, even in low-light conditions. Besides detecting more crime and less burdening security personnel, efficiency and effectiveness would also be improved in the protection of environments under low light conditions[6].

This thesis proposes a deep learning-based framework for detecting criminal activity in videos under low-light condition. We utilize five state-of-the-art deep learning neural networks : CONVLSTM, VGG-16, Mobilenet-V2, NasnetMobile and LSTM. The choice of these models is strategic, leveraging their unique architectural strengths to extract nuanced features from diverse and big datasets. These models excel at extracting subtle features from video data, making them ideal for analyzing human expressions, posture, movement, and interactions. We design a system for collecting data on human behavior through various sensors, including cameras, microphones, and motion detectors[10].
The key contributions of this work are:

1. Using Spatial and Temporal characteristics: Detecting criminal behavior in video needs the capture of both spatial and temporal characteristics. Spatial features come from each frame, while temporal features collect movement and activities across frames. To accurately model these factors, we employ a combination of CNN (for spatial features) and LSTM (for temporal features)[9].

2. The data is then preprocessed to extract relevant features and prepare it for model training. The preprocessed data is fed into the chosen models, where it is analyzed for anomalies and deviations from established baseline behavior patterns. The models are trained on labeled datasets containing examples of normal and malicious behavior, enabling them to accurately identify potential threats.

3. To further refine our technique, we use transfer learning, which allows the models to adapt to the specific problems of low-light video data without the need for substantial retraining.

4. The UCF Crime dataset, which is often used to detect criminal activities, has been enhanced to imitate low-light settings. This allows us to test our models' performance in realistic low-light settings, providing that the results apply to real-world surveillance systems.

This work addresses the integration of deep learning techniques, video enhancement, and transfer learning for better criminal activity detection from low-light videos. The study not only tackles present inadequacies in surveillance systems, but it also suggests a more efficient strategy to dealing with difficult real-world [5]situations. Our findings indicate that enhanced video preprocessing, spatial-temporal feature extraction, and transfer learning can greatly improve detection accuracy in low-light circumstances, paving the way for more dependable surveillance systems such as Corporate Security, Critical Infrastructure Protection, Airport Security, and Border Control ensuring the safety and well-being of individuals and vital infrastructure.[18]

The following chapters of this paper are the Problem statement, Research Objective, Literature Review, Working Plan, Dataset, Model Explanation and Usage, Model Implementation, Result and Analysis, and Conclusion arranged as follows.

1.1 Research Problem

Current personnel security systems largely rely on static rules or subjective human assessments especially in low-light environments, leaving them vulnerable to evolving threats and susceptible to biases and inconsistencies. While existing video surveillance systems capture anomalies, their post-factor analysis limits their effectiveness in preventing immediate harm. Consequently, this leads to a need for a proactive personnel security solution that can detect illegal or criminal activities and behavioral deviations in real-time under low-light circumstances, triggering timely interventions and significantly reducing the risk of security breaches. [16]

Traditional security measures, which tend to depend only on human surveillance, are restricted in their ability to monitor every aspect of human action especially when you are all day looking at a monitor there is a high chance that in case of low-light conditions, one will not able to properly address the situation as it is really hard to understand what is happening. [17]
The problem at hand is to detect criminal activity within the location, and deploy

a system that can successfully analyze the situation at any condition[11].

In this thesis, we focus on enhancing the security of any organization or environment through the application of Convolutional Neural Networks (CNNs). Factories, being bustling hubs of activity with numerous employees, visitors, and goods, are potential hotspots for various security threats. These threats can range from theft of valuable items to acts of sabotage, like putting on fire intentionally, unauthorized access to confidential information or trade secrets, conflicts among employees, or external threats leading to physical harm, Negligence, or intentional acts that compromise the safety of the workforce, such as tampering with safety equipment[12].

Existing surveillance systems frequently rely on rule-based algorithms and heuristics, which have limited adaptability to the ever-changing landscape of human activities[3]. Furthermore, these systems frequently generate a large number of false alerts, increasing the cognitive stress on security workers and increasing the possibility of major incidents being missed.

Despite advances in deep learning, there is still a significant gap in the implementation of advanced neural network designs intended specifically for low-light video analysis. Current research has primarily focused on daytime surveillance or has failed to effectively address the specific issues presented by low-light circumstances. Furthermore, while certain models have demonstrated promise in feature extraction and temporal analysis, no complete evaluation of the performance of numerous architectures—such as VGG-16, MobileNetV2, NASNetMobile, and CNN-LSTM—under these conditions exists. Deep learning models powered by neural networks can handle massive volumes of visual data, and if it is handled properly we believe it is possible to detect even in the low-light condition which has the potential to revolutionize personnel security systems.

To date, there is a large gap between research and practical implementation of deep learning models for detecting criminal activity videos under low-light conditions. Bridging this gap is a key requirement for protecting these essential assets, not just a technological undertaking.

As a result, the core aim of this thesis is to create, construct, and test a deep learning-based model capable of detecting criminal activities under low-light conditions. This model must function in real-time, easily connect with existing surveillance systems, and significantly eliminate false alarms.

1.2 Research Objectives

The fundamental goal of this research is to make deep learning models to adapt to low-light conditions. We aim to detect threat levels in various organizations or public spaces accurately. There are the following specific research objectives that have been outlined to attain this broad purpose.

1.**Literature review:** To conduct a comprehensive examination of the available

literature on existing low-light condition research systems and also to determine the benefits and drawbacks of current approaches for video activity detection and management under low-light conditions.

2. **Data Collection and Processing:**To collect a comprehensive data set of footage of criminal activities or similarly available data sets for taking care to cover a variety of crime classes. Data should be prepossessed to reduce noise and useless information.

3. **Model Selection:** Not all models can perform best on low-light conditions we need to figure out which model is performing on low-light conditions or in similar conditions well and for that, we may have to train lots of models or use the pre-trained model on the similar dataset to see which suits our needs best..

4. **Feature Extraction:** Implementing strategies for extracting features from video frames, with a focus low light conditions will allow the model to distinguish between normal and low-light conditions.

5. **Transfer Learning:** Establishing that the proposed model can transfer learn and can learn from different dataset and can give accurate results.

6. **Detecting Low-Light condition:** Integrating a criminal activity detection model that will able to detect criminal activity under low light also

7. **Evaluation and Validation:**Evaluating the generated model's performance rigorously using a variety of data sets, including simulated threat scenarios and historical surveillance footage. Examine the accuracy, sensitivity, specificity, and false alarm rate of the model.

8. **Integration with existing System:** Ensuring that the deep learning model is seamlessly integrated with the current monitoring and security infrastructure in public spaces.

This thesis intends to make a substantial contribution to the field of personnel security systems

Chapter 2

Literature Review

The integration of deep learning in security systems has the potential to improve security at a significant level. Criminal activity detection in videos, especially in low-light settings, has emerged as a significant study topic in computer vision and surveillance systems. Traditional video surveillance methods, which depend on human surveillance or simple detection of movement, are progressively being substituted with automated systems that use machine learning and deep learning techniques. These automated solutions offer increased accuracy, efficiency, and scalability in spotting suspicious behavior.[27] Insider sabotage and suspicious activities inside the organizations will be detectable, especially in low-light environments more accurately than the current human-oriented security measures. Applying the Deep learning model by using different kinds of natural networks will be able to detect the slightest of things that are very hard to identify by humans also it works at such a speed and can calculate data so efficiently and accurately that it will produce not just the better result but also will do something which humans are not capable physically. Several research has investigated the use of deep learning for video classification, with models such as Convolutional Neural Networks (CNNs) demonstrating substantial effectiveness in identifying patterns from video frames. Recurrent Neural Networks (RNNs), specifically Long Short-Term Memory (LSTM) networks, have also been utilized to observe the temporal dynamics of videos, enabling greater comprehension of actions and activities over time[4]
In this part, previous relevant work in the deep learning field where human behavior and actions are analyzed to detect suspicious behavior has been taken into consideration as the work is still going on in these fields there is still a lack of data sets. But in the case of behavioral analysis, there are many relative fields where behavior analysis is in use, and also much research has been conducted on human behaviors and their suspicious or criminal activities. We will try to find relevant methods from these research papers to infuse into our model and get the most accurate result.

2.1 Related Works

The research[24] introduces a novel approach to assessing human poses in extremely low-light conditions, addressing important data gathering and prediction accuracy issues. The authors present the ExLPose dataset, which consists of 2,556 pairs of aligned low-light and well-lit photos, allowing for accurate posture classification due to the better visibility of the well-lit images. A dedicated camera system was created

to record these paired images concurrently, resulting in high alignment and accurate annotations. The suggested method uses a Learning Using Privileged Information (LUPI) framework, in which a teacher model processes well-lit images to provide extra supervision to a student model processing low-light inputs. This novel approach improves the resilience of pose estimation under changing lighting conditions.

The authors propose[4] a unique framework that uses deep multi-view representational learning to distinguish actions from several video streams simultaneously, improving robustness against noise and degradation common in low-light conditions. A spatial-temporal based-on features correlation filter is at the heart of their approach, allowing numerous actions to be detected and recognized simultaneously. The system uses C3D features to capture spatiotemporal dynamics and a deep fusion method to combine data from several visual spectrum. Comprehensive trials on nocturnal action datasets reveal that this strategy is effective, with significantly better action recognition results than conventional approaches. This work emphasizes the possibility of establishing action detection techniques that work well under adverse lighting conditions, paving the path for uses in many sectors such as surveillance.

In this paper[8], researchers investigate advances in intelligent camera surveillance, with an emphasis on the detection and evaluation of human behavior utilizing an innovative two-channel deep convolutional neural network (DCNN) model. Researchers, replicate the brain's visual processing utilizing independent channels for spatially (static) and temporally (dynamic) information, which allows for more accurate feature extraction. This dual-channel technique greatly surpasses classic single-channel models in identifying human activities, as evidenced by trials on the KTH cognitive dataset. The suggested approach successfully records joint motion information, resulting in excellent recognition precision while preserving low computing overhead, which makes it appropriate for real-time applications across a variety of areas, including security and human-computer interaction[24].

The paper[2] introduces a dual-stream network-based cloud-assisted IoT architecture for Human Activity Recognition (HAR) in conditions of low light. The method solves the issues of detecting actions in complicated, low-light environments while optimizing compute capacity for edge sensors. First, a lightweight CNN improves low-light video frames while detecting human activity in specific frames. These improved frames are subsequently analyzed by a dual-stream network that combines CNNs and transformers to extract spatiotemporal data at both short and long distances. The optimized Parallel Sequential Temporal Network (OPSTN) incorporates a squeeze-and-excitation attention system to minimize data volume and increase activity categorization accuracy. The system excels existing algorithms on three sample datasets (HMDB51, UCF50, and YouTube Action), especially under difficult situations, displaying higher levels of precision and effectiveness[21].

This research paper[26] focused on using Inception-v3 transfer learning approaches to accurately detect abnormal behavior in video surveillance footage. By applying both pre-training and fine-tuning transfer learning approaches, the Inception v3 network classifies keyframes as normal and abnormal behaviors. The study utilized the UCF Crime dataset for model training and evaluation, using metrics like accuracy, recall, precision, and F1 score. Moreover, the fine-tuned Inception-v3 model outperformed the pre-trained model in terms of accuracy, recall, precision, and F1 score. The enhanced performance of the fine-tuned Inception v3 model indicates that this

approach could be highly beneficial for practical applications in video surveillance for detecting abnormal behaviors.[23]

In this research [7] a system was developed to differentiate abnormal and normal events based on the analysis of movement information from video sequences. The HMM method was utilized to learn the histograms of the optical flow orientation of the video frame. By comparing the captured video frames with the existing normal frames, the system identifies the similarities between them. Eventually, the system was tested and validated on various datasets such as the UMN dataset and PETS[27]

This research[28] represents an efficient Crime Monitoring System (CMS) that can detect crime in real time through camera surveillance. This system (CMS) mitigates human limitations such as inattentiveness, slow reactions, etc. The CMS integrates CCTV with deep learning and image processing, operating across three stages such as weapon detection using YOLOv5, violence detection with MobileNetv2, and face recognition via specialized algorithms. The system utilizes an image dataset for overall operations and a video dataset for training the violence detection model, using frame-by-frame images from videos. The performance results for weapon detection, violence detection, and face recognition are quite impressive. The CMS effectively identifies crime incidents and triggers timely alerts, demonstrating its significant potential to enhance security and safety through advanced surveillance technologies[14].

The study [25] offers a novel approach that makes use of modern deep-learning methods to identify anomalies in videos, such as violent conduct. The goal is to make real-time monitoring systems more successful in avoiding and addressing crimes by evaluating the temporal (when) and spatial (where) components of video frames using a 3D CNN and Convolutional Block Attention Module (CBAM) to increase accuracy on significant regions of the video frames. In this paper, they proposed a method compiled with different techniques such as Frame merging, Data structure, and training evaluation. The Frame margin combines several video frames into a single image to save memory and increase the model's capacity to handle larger amounts of data. Moreover, the Data Structure technique is responsible for joining the frames and creating an input structure that facilitates the model's understanding of intricate behaviors across time. Lastly, In the Training and Evaluation method UBI-Fights, RWF-2000, UCSD Ped1 and Ped2, and other datasets are used on violent behavior identification to train the model. This implemented method exceeds existing models and one of the datasets, UBI-Fights accomplished 99.33% accuracy. This research paper concludes this proposed method can be used in several real-time situations, such as keeping an eye on public areas or guaranteeing student safety in schools. Further research could be done on the merged frames and attention module technique to improve anomaly detection in other scenarios.

In this paper [13] The authors implemented a lightweight and efficient model that detects anomalies in surveillance films using Convolutional Neural Networks (CNN) and a residual attention-based Long Short-Term Memory (LSTM) network, which is essential for public security in modern cities. This research paper proposed Frameworks: A CNN is implemented to extract spatial characteristics from video frames. On the other hand, The residual attention-based LSTM network, which is intended to identify abnormalities in the video sequences, is then given these features. Here

overall, CNN is utilized for feature extraction and LSTM is applied for sequence learning, the model's capacity to identify unusual activity is improved. The authors minimize the computational expense by using MobileNetV2, a lightweight CNN, to pull out features. The residual attention-based LSTM enhances the identification of abnormal sequences through the selective concentration of pertinent features and the disregard of additional data.

This model [31] is being implemented on several benchmark datasets, such as Avenue, UMN, and UCF-Crime to assess the model's performance. It achieved higher accuracy compared to other traditional techniques. The UCF-Crime dataset obtains 78.43% accuracy, the UMN dataset has 98.20% accuracy, and the Avenue dataset shows 98.80% accuracy (an increase of 8.62%). Moreover, this paper [5] concludes the suggested framework detects anomalies in surveillance footage successfully and accurately because the residual attention-based LSTM and MobileNetV2 models make it appropriate for real-time applications in smart surveillance systems. To further improve performance, future research will investigate additional deep learning models, such as 3D models and graph neural networks.

The goal of this study [29] is to discover traits, that emerge during snatching. They applied the VGG19 neural network to find anomalies in videos from their Snatch 101 dataset.

When compared with alternative approaches, the suggested methodology is more accurate and efficient (81% accuracy). However, manually identifying particular objects or individuals in surveillance footage is difficult, especially in spotting unusual activities in crowded places to stop crimes. They use of the Snatch dataset contains recordings of both incidental snatching and typical behavior and these photos are being processed using the VGG19 algorithm to find abnormalities.

In this paper [20] suggested a model combining convolutional and Long Short-Term Memory (LSTM) layers to detect violent events from surveillance footage and these are routed via a Telegram bot to law enforcement for prompt action. Here, CNNs are used for spatial features while LSTMs are implemented for temporal relationships in the proposed system. After being divided into frames, surveillance footage is sent to a MobileNet v2 classifier. Violent frames are improved and given to police via Telegram along with their location, while non-violent frames are destroyed. Moreover, CNNs work well with image data, while LSTMs perform well with sequences. However, here, they implemented MobileNet V2 architecture as it is suitable for real-time violence detection because this technique uses depth-wise separable convolutions for efficiency and accuracy and detects the anomaly activities from the dataset successfully.As a result, the MobileNet v2 model accomplished 96% training and 95% testing accuracy within the 1000 videos which performed better at detecting violence in real-time compared to earlier models like CNN-LSTM.

In this paper [30] they followed some steps to implement the method including image classification they created grayscale graphics from incoming video data and adjusted it to intensity values using lighting improvement techniques. Moreover, For the Selection and Extraction of Features process, they exploited both spatial and temporal detectors and extracted local motion information at the pixel level using a differential evolution algorithm for optimization. It also uses a 2D cellular network to enhance the detection process.The approach was evaluated on the UCSD

Anomaly Detection Dataset with MATLAB.It outperformed other techniques in terms of speed and accuracy, showing an average detection rate of 98.6%. However, On a home computer, the method executes in three seconds.

This research [1] provides a model for detecting anomalous incidents in surveillance footage, using ResNet50 and a recurrent neural network (ConvLSTM). This proposed model utilizes the real-time camera information of subjects and situations. It achieves 81.71% accuracy on the UCF-Crime dataset to detect theft, aggressive behaviors, and firefighting. This model aims to increase the precision and effectiveness of detecting abnormal activities by combining ConvLSTM for video sequence analysis with ResNet50 for feature extraction.This model is followed by preprocessing the data and splitting it to calculate the frames, then it pull out the features through the ResNet50 model. It also exploits the ConvLSTM technique to detect suspicious behaviors successfully.

This paper [15] applied machine learning techniques including Convolutional neural networks (CNNs), long short-term memory networks (LSTMs), support vector machines (SVMs), and recurrent neural networks (RNNs) to identify unusual human behavior in video surveillance data. In this paper, the main goal is to find out the outcomes by assessing and comparing each model's performance. The work consists of an experimental evaluation with recently recorded video footage from Axis Communications AB and the UCF-crime dataset. The algorithms' performance in identifying anomalies is measured by the assessment metrics ROC-AUC (Receiver Operating Characteristic - Area Under Curve) and EER (Equal Error Rate). After the implementation of each model, CNNs achieved the highest accuracy with a ROC-AUC of 0.8346 and an EER of 0.2439, LSTMs showed the accuracy with a ROC-AUC of 0.7797 and an EER of 0.3039, SVMs accomplished the lower performance with a ROC-AUC of 0.7281 and an EER of 0.3780 due to it limited scalability issues and sensitivity to parameter tuning and RNNs achieved a ROC-AUC of 0.7602 and an EER of 0.3292, because of its vanishing gradient problem and limited memory capacity[19]. According to the study's findings, CNNs perform resilience against noise and their capable of handling spatial variables makes them the best algorithm for detecting behavioral anomalies in video surveillance data. However, Their need for huge datasets and powerful computers is a disadvantage, though. On the other hand, long short-term memory banks (LSTMs) are useful for capturing temporal patterns.[22]

In conclusion, we can conclude that all of the provided research papers consist of different techniques and methods that are relevant to the objective of our thesis. In our research paper, we may utilize these papers to achieve our desired outcome, detecting the potential threats by analyzing our low-light dataset. Moreover, the related works provide a solid foundation and understanding for integrating various architectures to find the suitable one for our thesis research, guiding us toward achieving the desired outcome of a robust and efficient crime detection system under low-light. By building upon the insights and methodologies presented in the reviewed papers, we can tailor our approach to effectively address the challenges and requirements of detecting anomalous human behaviors with various deep-learning approaches.

Chapter 3

Dataset

Dataset is the center of a research without proper dataset no research is fruitful. For our research we needed a dataset which is very robust and big. As we are working on the security day-to-day basis of the people the model needs to be able to handle big amount of data. So testing our model with large quantity of data is a very crucial part of our research

3.1 Dataset Choosing

For our work, we have decided to use the UCF Crime dataset. Our main objective is to classify videos, especially crime-related videos where any particular class of crime is happening and we want to do it under low light and low resolution means even if the video quality is bad or things are not visually properly visible it will be able to detect the crime class. So we basically need a dataset where we can get crime-related videos and there is no better dataset than the UCF Crime dataset. UCF crime dataset is currently the largest dataset containing almost 128 hours worth of videos of real-life crime incidents which has been divided into 14 general classes. All this footage is captured by video surveillance from schools, roads, general stores, parks, and many other places where the real crime happened. The incident recorded here can happen to anyone at any time and shows the biggest security threats in our daily lives.

3.2 Charectaristic of Dataset

The UCF Crime dataset is the most versatile dataset of all it contains more than 1900 video clips which are taken in both indoor and outdoor environments using various surveillance cameras so that it can detect abnormal or illegal activities in real-world situations. All the video dataset has been divided into 14 classes of which 13 belong to pure crimes and threats which include 13 distinctive classes of anomaly events or threatful events, including abuse, arrest, arson, ambush, Road accident, Burglaries, Explosions, Fighting, Robbery, Shooting, Stealing, Shoplifting, and Vandalism.There are 950 uncut real-world surveillance videos and 950 regular videos. 70% of the videos are under 3 minutes which makes the dataset very useful for video categorization. The dataset is divided into two parts as usual training and testing from which 75% are used for training and 25% used for testing.Each of the

14 classes found in the dataset is represented in fig-1

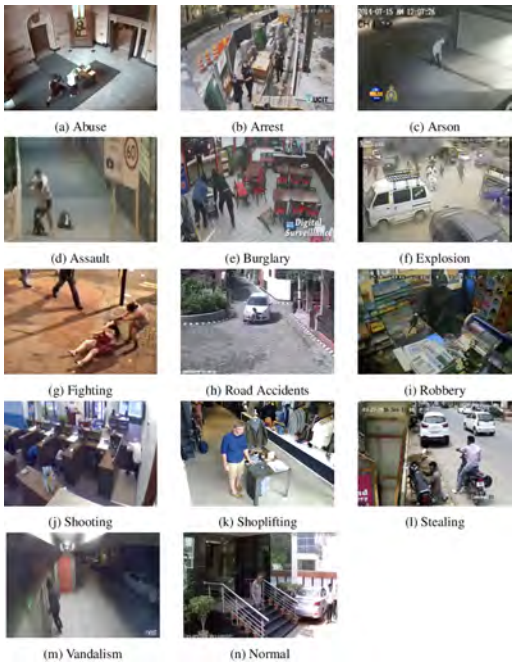

Figure 3.1: UCF-Crime Dataset

3.3 Dataset Processing

For the dataset processing, we have to go through multiple processes to make them work well with the model architecture. Below is the explanation of the whole dataset-preparation process

3.3.1 Frame Resize and maximum sequence length limit

We wanted to avoid unwanted computational overhead so we resized all the frames extracted from the video. Now we know video is a collection of a series of frames over a certain period and we can extract any amount of frames from the videos but the problem is computational power which is a limitation. We can extract only the amount of frames that our device can compute and because of that, we have limited the maximum number of training frames to pass as one sequence for each class of dataset to 2000 frames per video.

```
data_dir = "/kaggle/input/"
image_height , image_width = 64, 64
max_seq_length = 2000
```

Figure 3.2: Frame Resize and Sequence length Limit

3.3.2 Creating List of classes

We have created a list of classes from our UCF crime dataset and for our purpose, we have taken twelve unique classes from our dataset which are "Abuse", "Arrest", "Arson", "Assault", "Burglary", "Explosion", "Fighting", "NormalVideos", "Shoplifting", "Shooting", "RoadAccidents", "Vandalism". The two other "Robbery" and "Stealing" are excluded as the data inside them was quite similar to the "Burglary" and "Shoplifting" and thus it can cause data fragmentation.

3.3.3 Extracting Video Frames

The function called extract frame is aimed at helping to extract portions out of a video, hence making it possible to compress the video before the training, which in turn saves a lot of training time. For this purpose, OpenCV contains a class called VideoCapture whose primary function is to capture video file, video sequences, images and other related media formats. The further rescaled extracted video frames will meet the requirements of (image height and image width) after which a proper evaluation of volume is done. It has to be pointed out that the frame's pixel values are in the range of 0 to 255. Therefore, after the original values are collapsed, the frames would be transformed in such a way that the effective range of all the pixel values would lie between 0 and 1 to maintain the effect of scale-invariant post-normalization. After completing that process of normalization, two thousand frames of all the classes available are randomly taken and added to a features list, the corresponding class for each of these frames is put in a labels list and then the labels of each class are transformed into a multi-class one hot encoded numerical representation.

3.3.4 Training and Test dataset generation

Following this preprocessing, we have two NumPy arrays: features (24000, 64, 64, 3), where 3 is the number of RGB channels, (64, 64) is the frame dimension, and 24000 is the total number of frames (total no of frames = no of frames in each class * no of class). Another NumPy array is labeled (24000,). We split the features and labels NumPy arrays to create a training and test dataset and kept shuffle = True.

3.3.5 Creating Low-Light Condition Dataset

For applying low light we have used openCv cv2.convertScaleAbs function where we have used scaling factor alpha for pixel intensity of the frame of video. A value less than 1 of alpha reduces the brightness of the frame significantly, makes it darker, and simulates low-light conditions. We have taken 0.3 as the value of alpha which converted the videos of our dataset in proper low light condition.

Figure 3.3: Normal UCF Video frame Vs. Low-Light UCF video frame

we preprocessed the new dataset by applying clahe function of openCV for contrast enhancement, We used the openCV Gaussian blur function to denoise the video frames and then we applied global histogram equalization to enhance the contrast of the video frames by adjusting the intensity distribution so that it become more uniform across all pixel values by spreading out the most fre- quent intensity levels it improves the visibility of image features in underexposed or low-contrast regions.

Chapter 4

Methodology

4.1 Working Plan

We will first preprocess our UCF crime dataset with resizing and Normalizing then will generate frames from the videos. After the frames are generated, we will split the frames in the Train and Test dataset and then train them through the Deep Learning models for feature extraction and understanding spatial-temporal movement in a video after the training we will save the model in .h5 format and then we will again preprocess the dataset by applying low light and low resolution them we will run the dataset through the saved model to see how it performs under the low light and low-resolution condition.

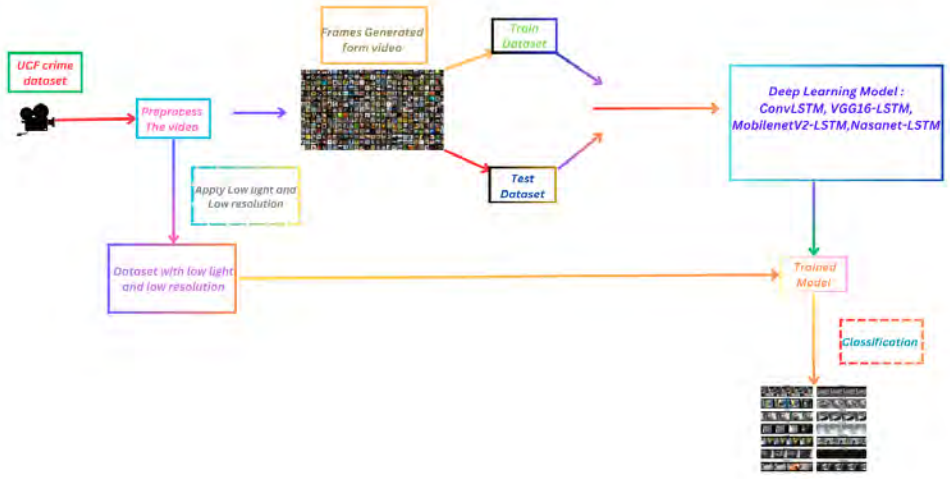

Figure 4.1: Proposed working plan for anomaly detection

4.2 Model Explanation and Usage

In this research, We applied the following Deep Learning architectures ConvLSTM, VGG-16, MobileNet V2 and NasanetMobile. These architectures were used to extract the spatial-temporal information from the video.

4.2.1 CONVLSTM2D

We used the CONVLSTM2D model architecture first which we got inspiration from the model defined in the paper "Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting" by Xingjian Shi et al []. So as you can understand from the name CONVLSTM it is a hybrid model of CNN and LSTM.In the regular Long Short Term Memory (LSTM) we use dense fully connected layers to connect the different gates of LSTM but in the case of CONVLSTM we bring convolutional layers from CNN and instead of dense fully connected layers we use Convlutina Filters to connect the different gates of LSTM. Now if we go further deeper into this the LSTM has three gates forget gate, input gate and Output gate. In LSTM we input the previous cell output with a feature map and multiply them together and for the multiplication, we use a feed-forward Multi-Layer Perceptron or inshort a dense layer. Now in the CONVLSTM2D instead of the dense layer for the multiplication, we are using Convolutional Kernels of size (3,3) and then move it across the feature map so that it can process the frames and extract the spatial-temporal features from it. After processing the data we got an output of 64*64*3 size feature map. This feature map is then passed through the dropout layer with a dropout rate = 2.0. So now all this is done the rest of the process is like a regular machine learning process where we applied flatten and passed it to a dese layer with 256 neurons with Relu activation function and then for the classification of data we passed it through another dense layer with softmax activation with 12 neurons and 0.3 dropout

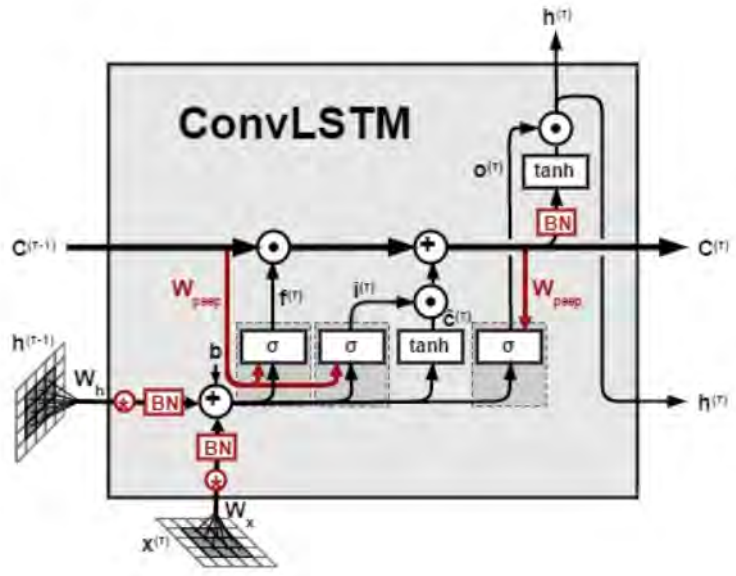

Figure 4.2: CONVLSTM2D Architecture

4.2.2 VGG-16

VGG-16 is a type of CNN which considered as one of the best computer vision models to date. It is a 16-layer network in which all convolutional layers has kernels/filters of size 3x3 with stride = 1 and padding = 'same'. Max pool layers have kernels of size (2x2) and stride = 2. VGG-16 is an ideal object detection and classification which can classify 1000 images of 1000 different categories. In our research, we used VGG-16 as a feature engineering tool. So instead of using the full VGG-16 network, we have dropped the last dense connected layers instead of we have used global average pooling which will generate one feature map for every category of the classification problem. So basically what we have done is we removed the dense layer from the top of the feature map and instead of them we have used the average of each feature map and then put them in the softmax layer. The reason we are doing this hassle is because global average pooling avoids overfitting as there is no parameter to optimize. It only sums out the spatial information as it is location-invariant

Model: "model"

Layer (type)	Output Shape	Param #
input_2 (InputLayer)	[(None, 64, 64, 3)]	0
block1_conv1 (Conv2D)	(None, 64, 64, 64)	1792
block1_conv2 (Conv2D)	(None, 64, 64, 64)	36928
block1_pool (MaxPooling2D)	(None, 32, 32, 64)	0
block2_conv1 (Conv2D)	(None, 32, 32, 128)	73856
block2_conv2 (Conv2D)	(None, 32, 32, 128)	147584
block2_pool (MaxPooling2D)	(None, 16, 16, 128)	0
block3_conv1 (Conv2D)	(None, 16, 16, 256)	295168
block3_conv2 (Conv2D)	(None, 16, 16, 256)	590080
block3_conv3 (Conv2D)	(None, 16, 16, 256)	590080
block3_pool (MaxPooling2D)	(None, 8, 8, 256)	0
block4_conv1 (Conv2D)	(None, 8, 8, 512)	1180160
block4_conv2 (Conv2D)	(None, 8, 8, 512)	2359808
block4_conv3 (Conv2D)	(None, 8, 8, 512)	2359808
block4_pool (MaxPooling2D)	(None, 4, 4, 512)	0
block5_conv1 (Conv2D)	(None, 4, 4, 512)	2359808
block5_conv2 (Conv2D)	(None, 4, 4, 512)	2359808
block5_conv3 (Conv2D)	(None, 4, 4, 512)	2359808
block5_pool (MaxPooling2D)	(None, 2, 2, 512)	0
global_average_pooling2d (Gl	(None, 512)	0

Total params: 14,714,688
Trainable params: 0
Non-trainable params: 14,714,688

Figure 4.3: VGG-16 Model Summury

As you can see the trainable params is 0. This happens because we are using a pre-trained VGG-16 layer so before training the model we had to freeze it so that it can't be updated during back-propagation

4.2.3 MobilenetV2

MobilenetV2 is CNN architecture invented by Google Researcher in 2018 for efficient embedded vision tasks. Its input layers take height, width, and channel as input. Depthwise separable convolution is heavily utilized in this model where it is divided into two parts depthwise convolution and pointwise convolution. In depthwise a single convolutional filter applies for each input channel separately which reduce the number of parameters which reduce the computational cost. On the other hand, pointwise convolution applies 1x1 convolutional filter so that it can combine the channels from the depthwise convolution and increase network capacity. The difference between MobilenetV2 and its original version is that it introduces Inverted Residuals with Linear Bottlenecks. Inverted residual starts with a light bottleneck layer with which is depthwise separable convolution and then it connects with a linear bottleneck which maintains information flow through the network efficiently. linear bottlenecks apply linear activations instead of RELU cause RELU is a non-linear activation function and can cause information loss. And then finally a global average pooling layer is applied which reduces the spatial dimensions of the feature maps to a single value per feature map by computing the average value of each feature map.By performing global pooling average we are computing the average value of each feature map which reduces the spatial dimensions of the feature maps to a single value per feature map

Input	Operator	t	c	n	s
$224^2 \times 3$	conv2d	-	32	1	2
$112^2 \times 32$	bottleneck	1	16	1	1
$112^2 \times 16$	bottleneck	6	24	2	2
$56^2 \times 24$	bottleneck	6	32	3	2
$28^2 \times 32$	bottleneck	6	64	4	2
$14^2 \times 64$	bottleneck	6	96	3	1
$14^2 \times 96$	bottleneck	6	160	3	2
$7^2 \times 160$	bottleneck	6	320	1	1
$7^2 \times 320$	conv2d 1x1	-	1280	1	1
$7^2 \times 1280$	avgpool 7x7	-	-	1	-
$1 \times 1 \times 1280$	conv2d 1x1	-	k	-	

Figure 4.4: Overall Architecture of MobilenetV2

In our research, just like VGG-16 we are using MobilenetV2 as a feature extractor. We are using a pre-trained MobilenetV2 layer so before training the model we had to freeze it so that it can't be updated during back-propagation

4.2.4 NasNetMobile

NasNetMobile is a CNN-based architecture that uses a cell-based search space. The model begins with a few convolutional layers, which do the initial feature extraction. The first convolutional layer uses batch normalization and ReLU activation. The core of the architecture is based on repeated normal cells and reduction cells. Normal cells extract features from frames and maintain the same spatial resolution from beginning to end. On the other hand reduction cells downsample the input feature map which reduces the spatial dimensions while increasing the depth of the feature maps. The process is quite similar to the other processes we have seen in CNN like in VGG-16 we had max-pooling and stride also in ConvLSTM and MobilenetV2. NasNetmobile also uses depthwise separable convolutions just like MobilenetV2 as it reduces the number of parameters and brings down the computational complexity significantly. Global average pooling is also used here as it reduces the spatial dimensions of the feature maps to a single value per feature channel and averages all spatial locations into a single number.

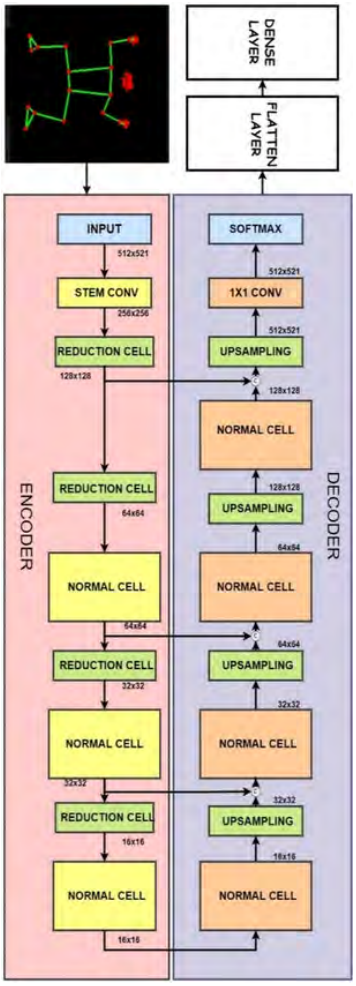

Figure 4.5: Architecture Of NasNetMobile

4.2.5 LSTM Model

So what we saw in the case of VGG-16, MobilenetV2, and NasNetMobile is that we are using them only for spatial feature extraction but for video classification we also need temporal sequence learning. The spatial feature will understand not only the objects in a video but also the body movement or what kind of activity is happening in a video to understand that what we need is temporal sequence. In case of CONVLSTM2D, the model itself is a hybrid model by combining convolutional Layer and LSTM so it can extract both spatial and temporal features using this one model. But for the other models, we have used them only for spatial feature extraction. We could also use them for temporal extraction also but it would not have given us our desired result as they are the model specialized for image datasets, not video datasets they are only good at extracting features and remembering them from images or frames but in the video, we need the temporal dependence which means multiple frames together creating a long or short sequence of an incident and that incident need to be stored so that if similar something sequence come to the model it can then compare and can classify or learn from it and LSTM is the right model for this work cause it's gating mechanism can control the flow of information. The mechanism consists of cells and gates (input, forget, and output) that selectively retain or discard information at each time step, allowing the model to learn from both recent and distant past inputs.

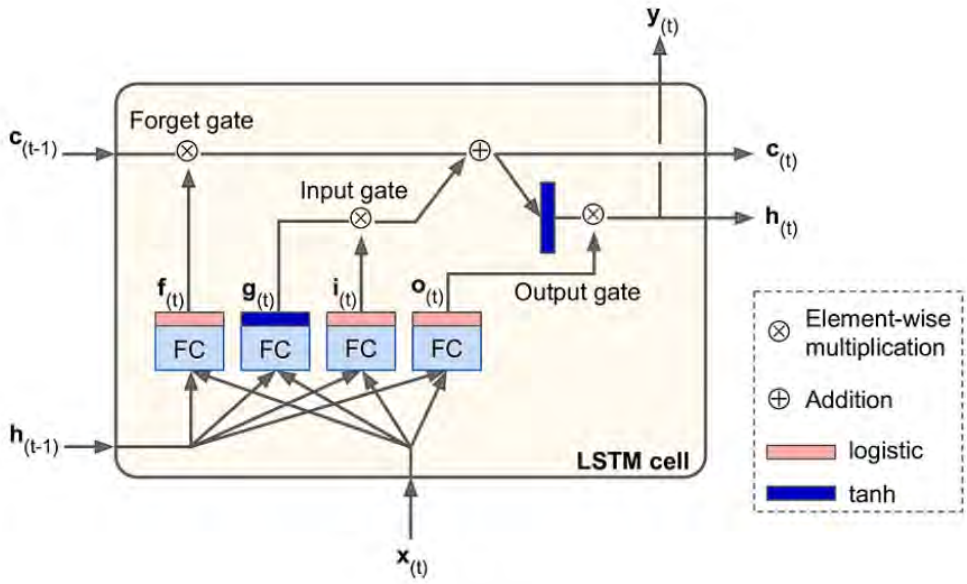

Figure 4.6: Architecture of LSTM Cell

Chapter 5

Implementation and Result

5.1 Implementation and Result of ConvLSTM

5.1.1 Training on Normal dataset and Testing on Normal and low-light Dataset

In the implementation part at first, we have preprocessed our dataset by resizing them, breaking them into frames, and then dividing them into train and test classes. 80% data used for Training and 20% for Testing. Then we have one-hot-code encoded the labels. After preparing the dataset first we trained the dataset through our CONVLSTM2D model. We have used early stopping call back with patience 7 and used categorical cross-entropy for calculating model loss. We have used a validation split of 0.2 or 20% of the train data. Then we saved the model in .h5 format. For the display of results, we used matplotlib.pyplot, which gave us the graph for both accuracy and loss training and validation.

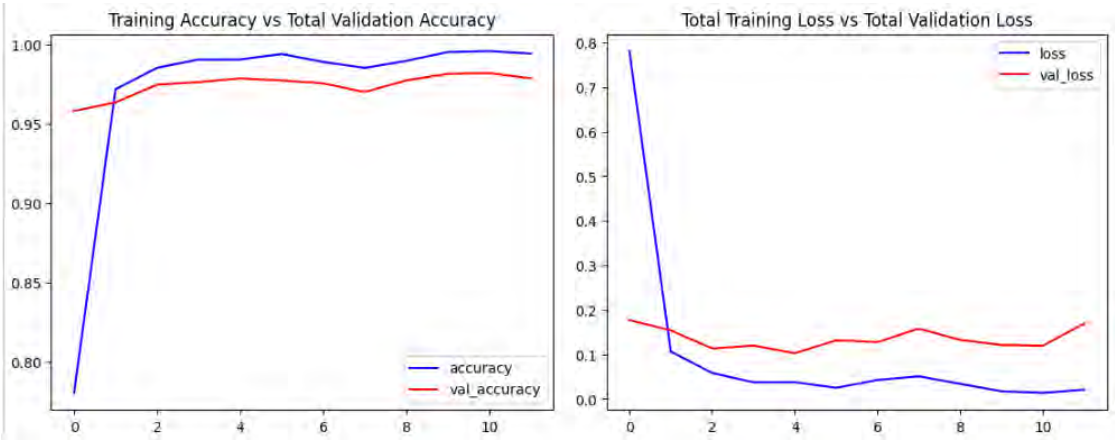

Figure 5.1: Training Result on Normal Dataset

Now after we saved the model as .h5 format file we used to test it with both of our datasets Normal UCF dataset by which we trained this model and another one in a preprocessed UCF Dataset where we applied low-light. We passed the raw videos through the model to see if a model trained in a normal Dataset can successfully classify two different types of datasets. So after running the model through the both of the dataset we get the below results

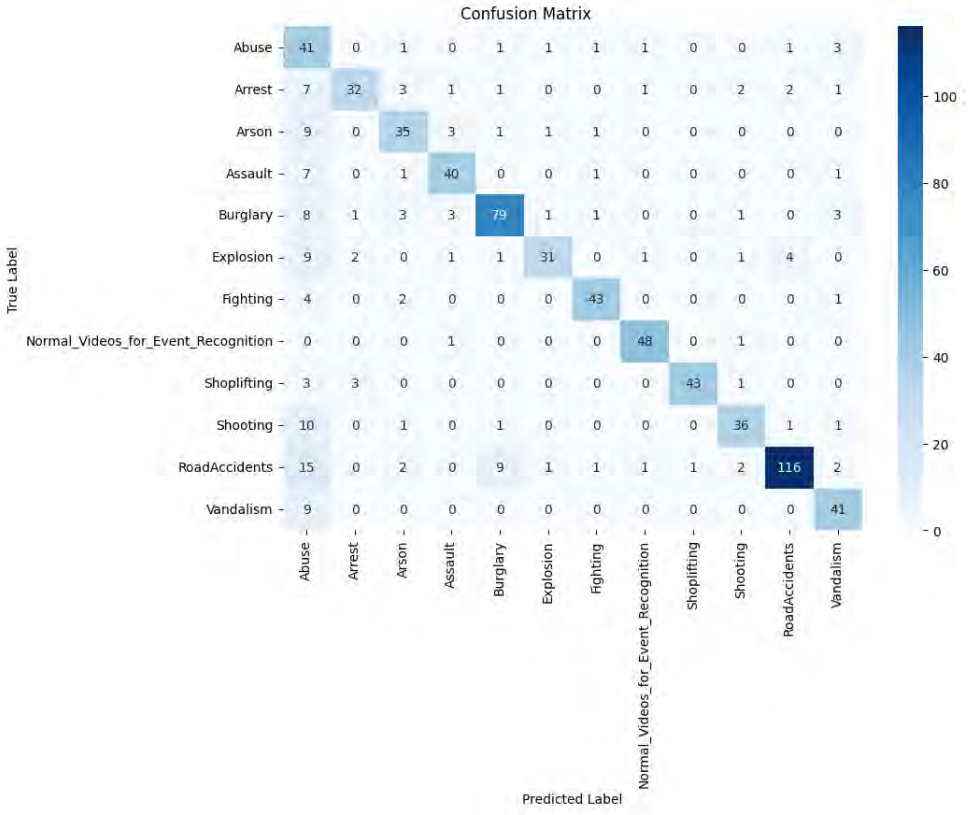

Figure 5.2: Confusion Matrix for Normal Dataset

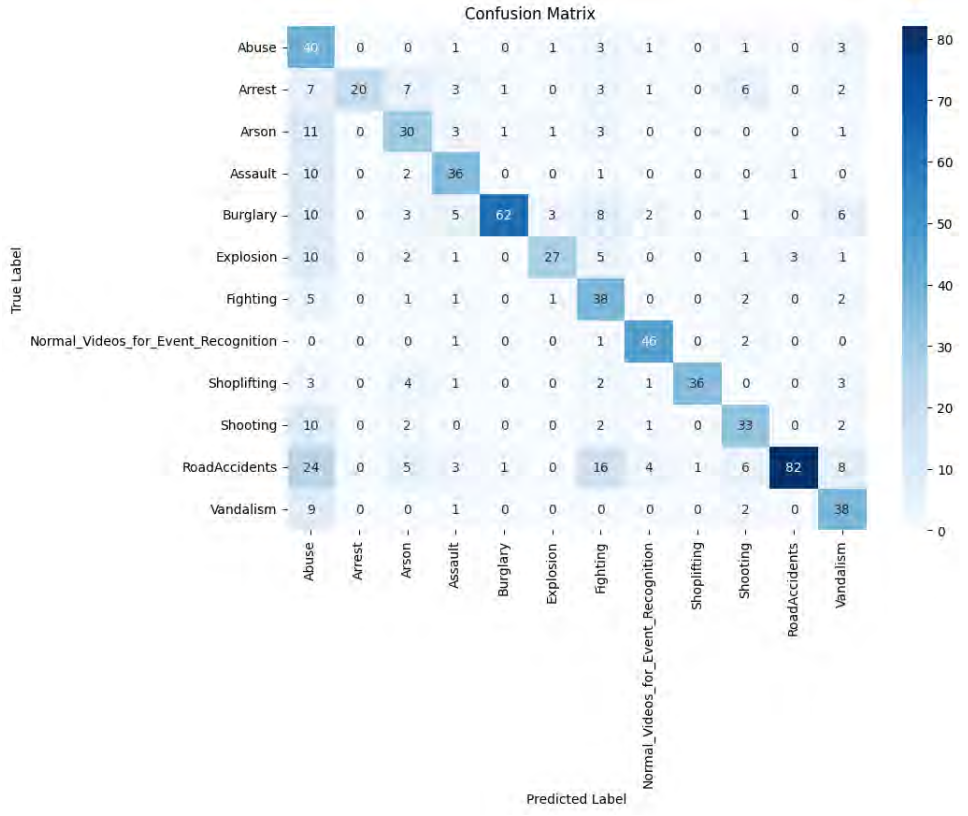

Figure 5.3: Confusion Matrix for Low-Light Dataset

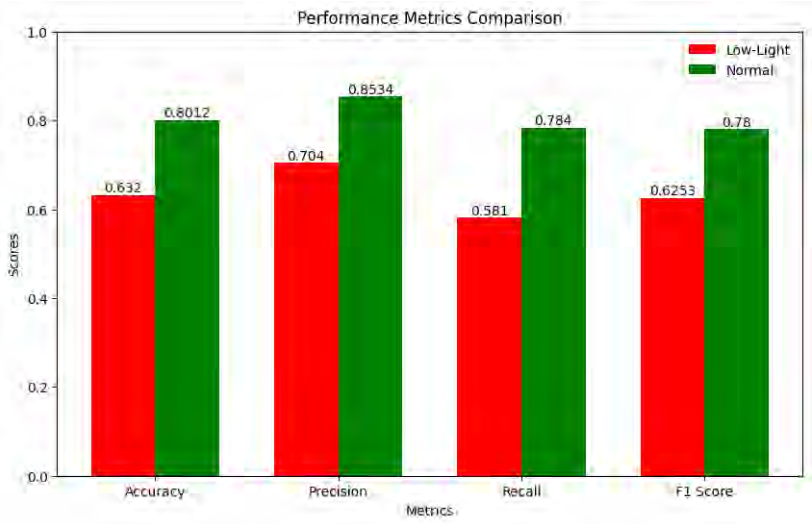

Figure 5.4: Accuracy, Precision,Recall, F1

5.1.2 Transfer Learning on Low Light Dataset and Testing on Normal and Low-Light Dataset

After analyzing the confusion Matrix and the other matrix we can understand that our CONVLSTM trained on Normal model is not performing well on the Low-Light dataset. This means the ConvLSTM model must also be trained on the Low-Light dataset so that it can perform well in the Normal dataset and the Low-Light datset and to achieve that, we can use transfer learning. Now for transfer learning, we will do everything the same what we did in the first place. We have preprocessed our dataset by resizing them, breaking them into frames, and then dividing them into train and test classes. 80% data used for Training and 20% for Testing. Then we have one-hot-code encoded the labels. After preparing the dataset first we trained the dataset through our CONVLSTM2D model. We have used early stopping call back with patience 7 and used categorical cross-entropy for calculating model loss. We have used a validation split of 0.2 or 20% of the train data. After that, we have

used the keras load function to load the model we trained before on Normal dataset. Now with the same configuration, optimizers and classifier we again train the model with the preprocessed low-light dataset by keeping the learning from the normal dataset intact. For that, we have frozen the 4 out of 6 layers of the CONVLSTM model and kept open only flattened and dense layers. After the training, we saved the model the same way in .h5 format. Now the transfer learning is done we again did the same thing we tested the model on both datasets.
So, after the training we can see that the performance have significantly improved and our model CONVLSTM performs well in both of the datasets

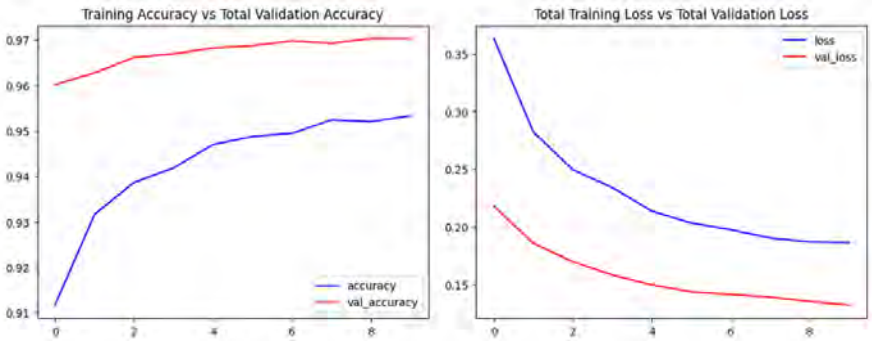

Figure 5.5: Training Accuracy and Loss for Low-Light Dataset

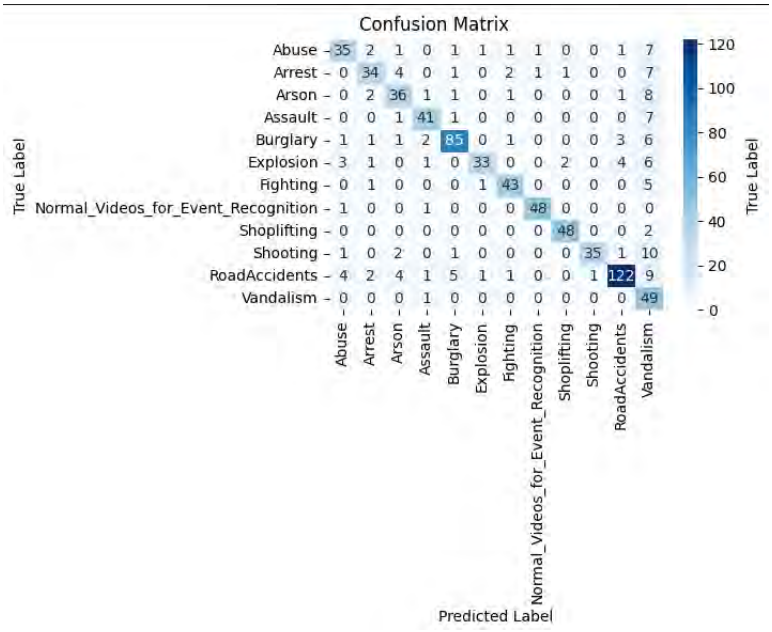

Figure 5.6: Confusion Matrix for Normal Dataset

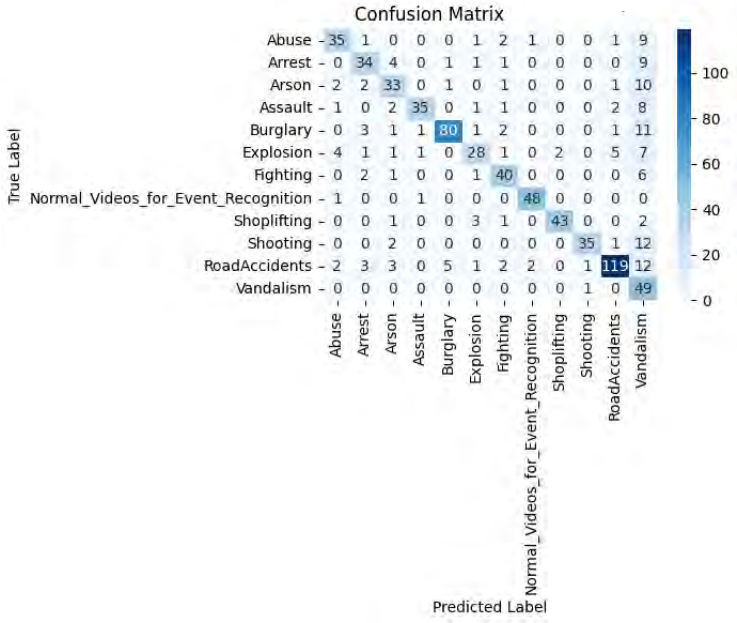

Figure 5.7: Confusion Matrix for Low-Light Dataset

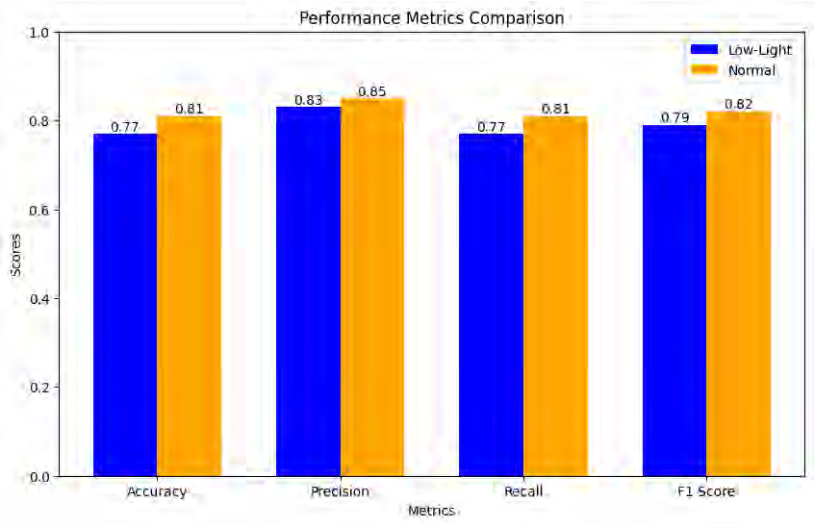

Figure 5.8: Accuracy, Precision, Recall, F1

5.2 Implementation and Result of VGG16-LSTM

5.2.1 Training on Normal dataset and Testing on Normal and low-light Dataset

In case of VGG-16 we are using the VGG-16 model for only feature extraction we have disabled some features of the model like removing the top dense layer from the feature map by making include-top = False. Also instead of dense layer at last we are using global average pooling (Explained in 4.2.2) that too imported using keras library.VGG16-model.trainable = False is used so the pre-trained model can't be updated during back-propagation. After this, we used the Keras TimeDistributed layer and passed the VGG-16 model to the Timedistributed layer which applied to every temporal slice of an input. By doing this we are encoding the video frames for the temporal sequence learning. Now after the model go through the time-distributed layer and we will pass it through our custom LSTM model for temporal sequence learning. the dataset processing is same as explained in the CONVLSTM part. After training, we saved the model and ran it through the datasets and got the below results

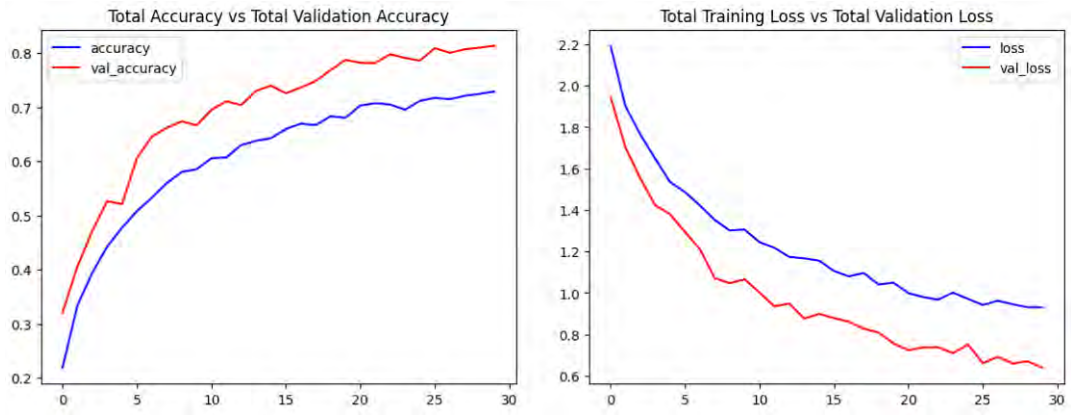

Figure 5.9: Training Accuracy and Loss for Normal Dataset

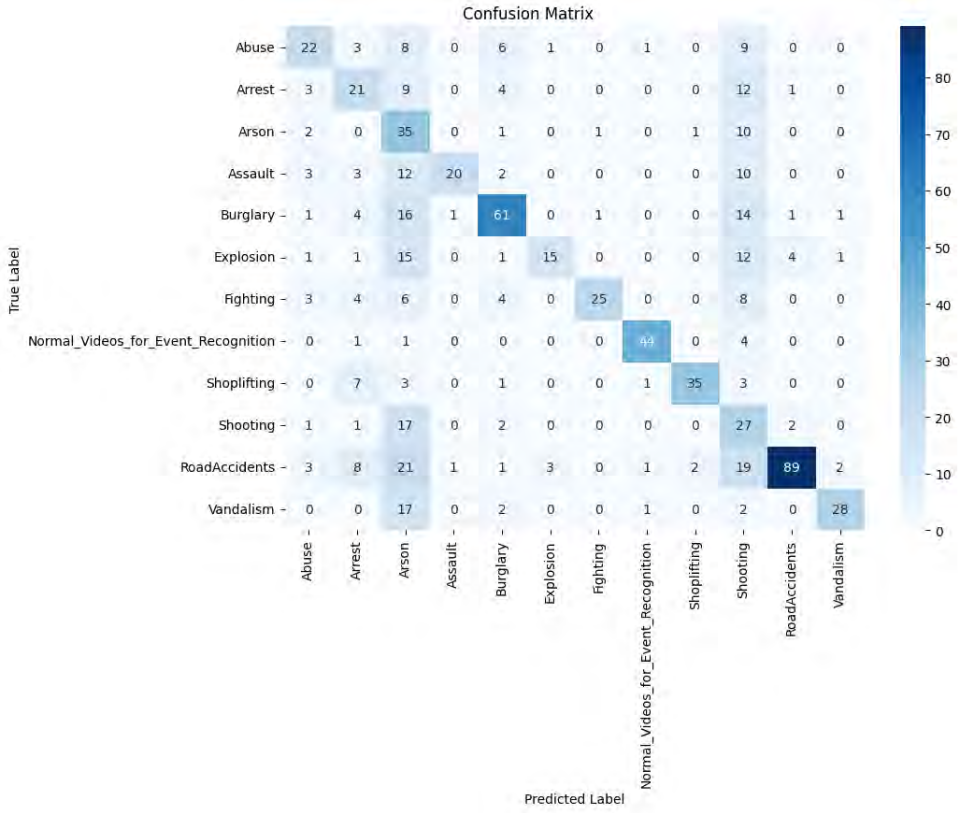

Figure 5.10: Confusion Matrix for Normal Dataset

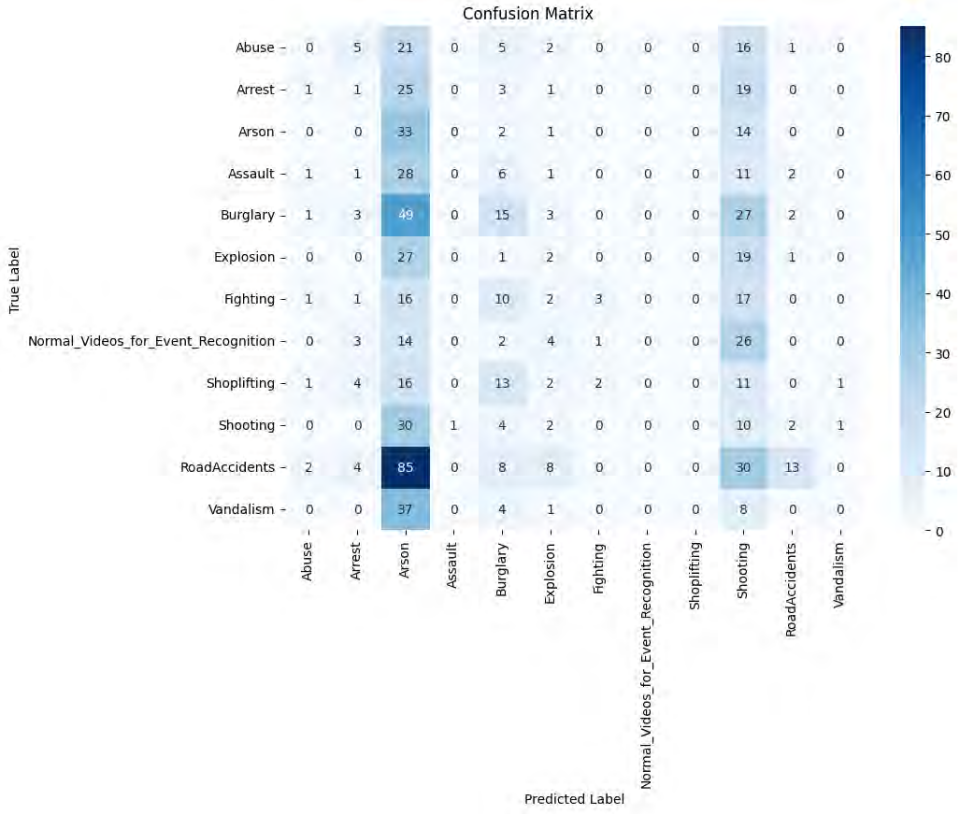

Figure 5.11: Confusion Matrix for Low-Light Dataset

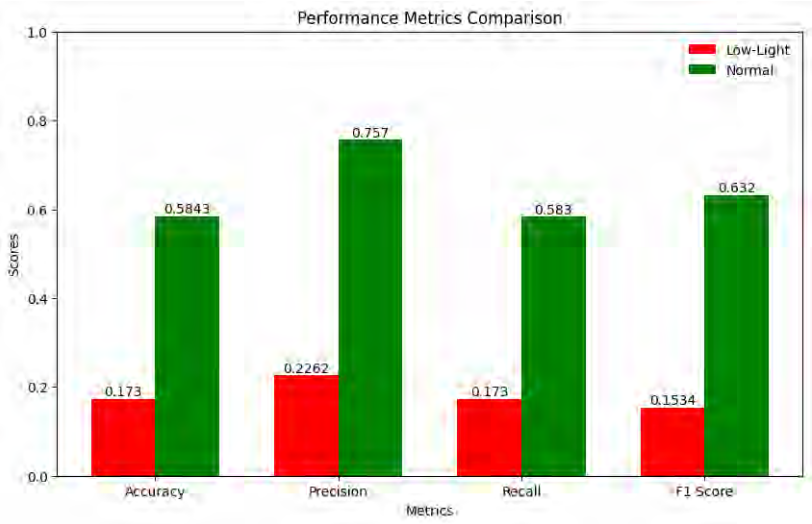

Figure 5.12: Accuracy, Precision, Recall, F1

5.2.2 Transfer Learning on Low Light Dataset and Testing on Normal and Low-Light Dataset

Now in the case of transfer learning we loaded the VGG16-LSTM model file and we froze the first 4 layers of the model after that we changed the optimizer from Nadam to Adam and we also changed the epoch to 50. Then after training we saved the model and run it through the Normal and low-light dataset

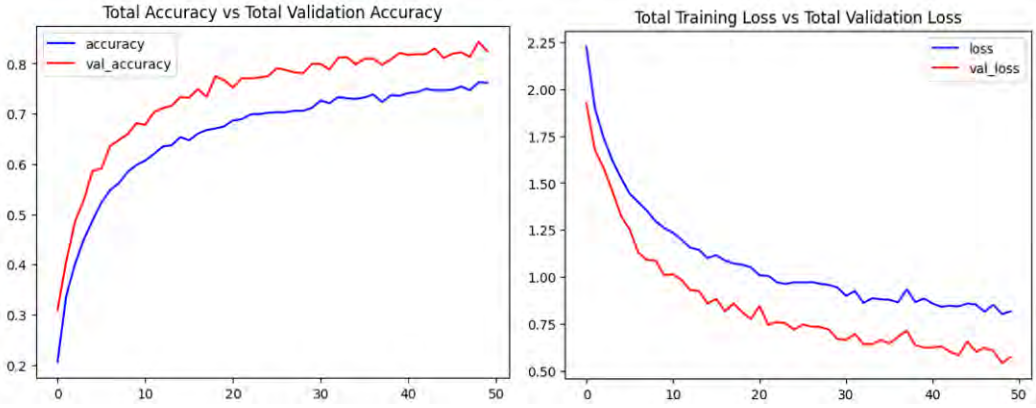

Figure 5.13: Training Accuracy and Loss for Low-Light Dataset

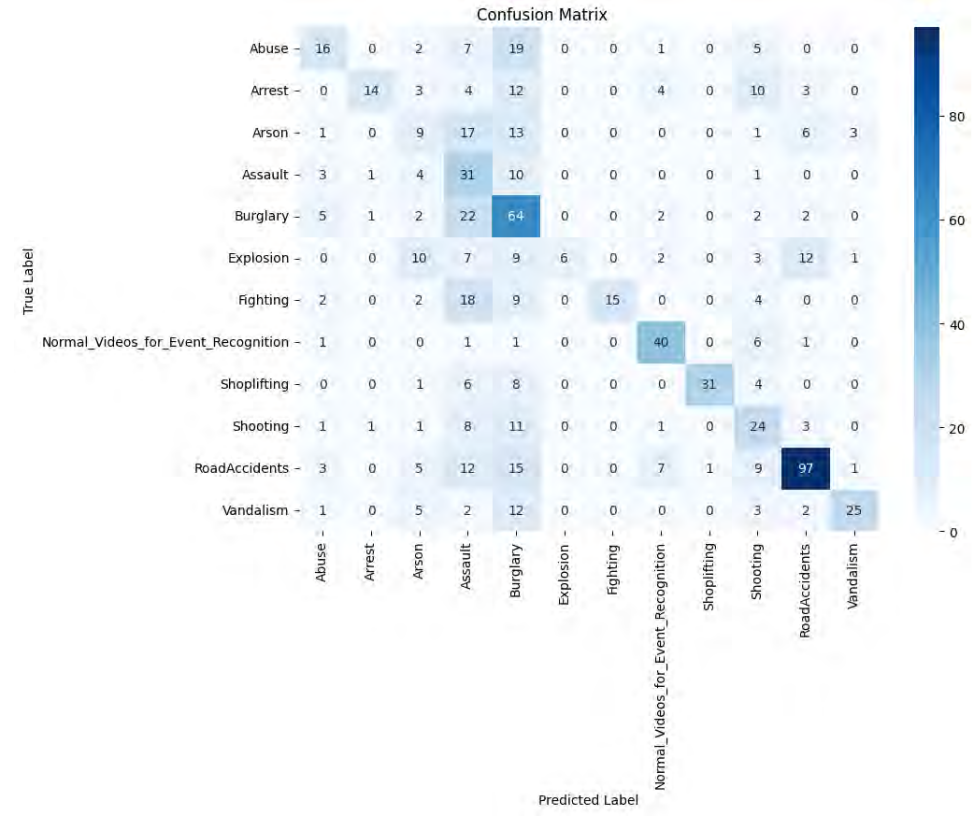

Figure 5.14: Confusion Matrix for Normal Dataset

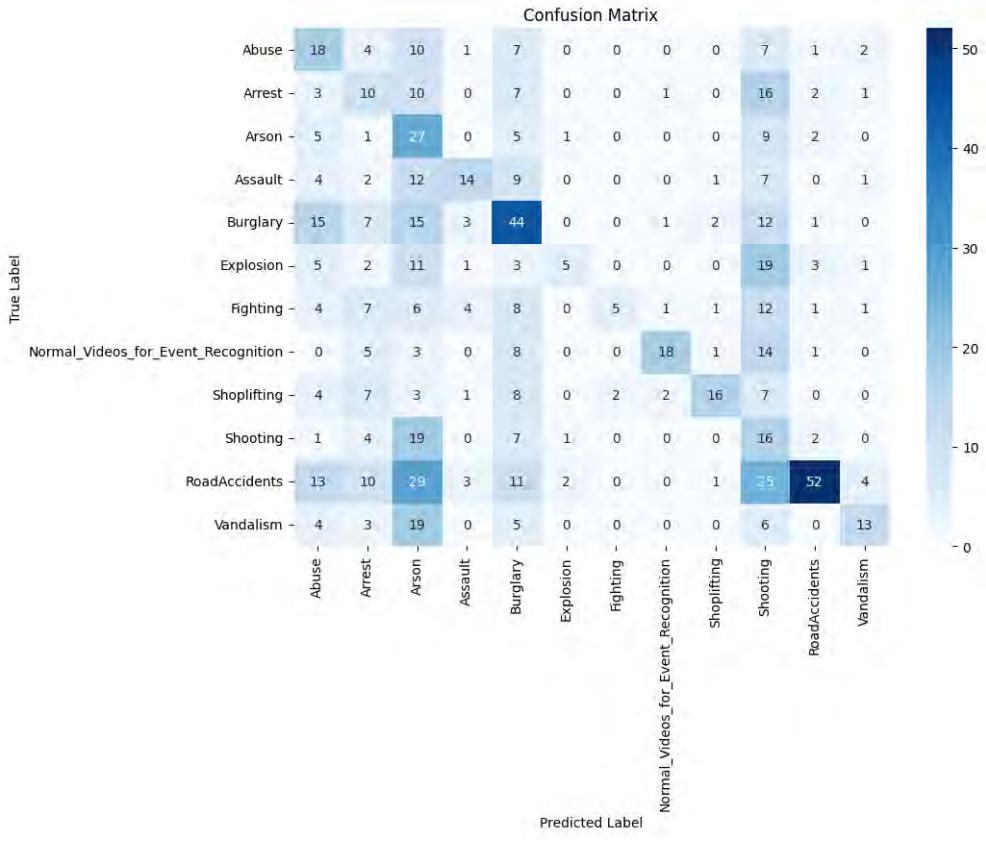

Figure 5.15: Confusion Matrix for Low-Light Dataset

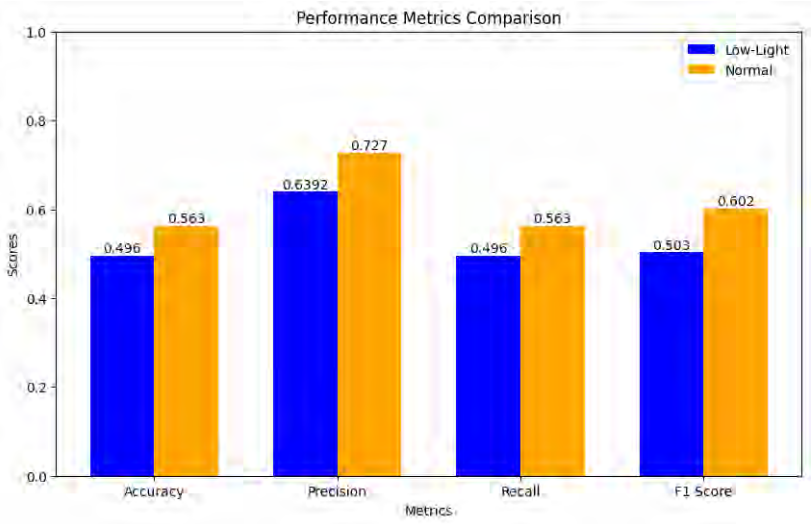

Figure 5.16: Accuracy, Precision,Recall, F1

So as we are seeing in the result there is a significant improvement in the result from transferring. It proves that the transfer-learning can detect low-light videos at least better than the only normal dataset-learned model

5.3 Implementation and Result of MobileNetV2-LSTM

5.3.1 Training on Normal dataset and Testing on Normal and low-light Dataset

So just like VGG-16 here also we used MobileNetV2 for the feature extraction and then ran them through the LSTM model for temporal sequencing. We have used the Nadam optimizer here with a 0.002 learning rate and ran the model for 30 epochs. After the model is trained we save the model and run it through the normal dataset and low-light dataset. Below is the following result.

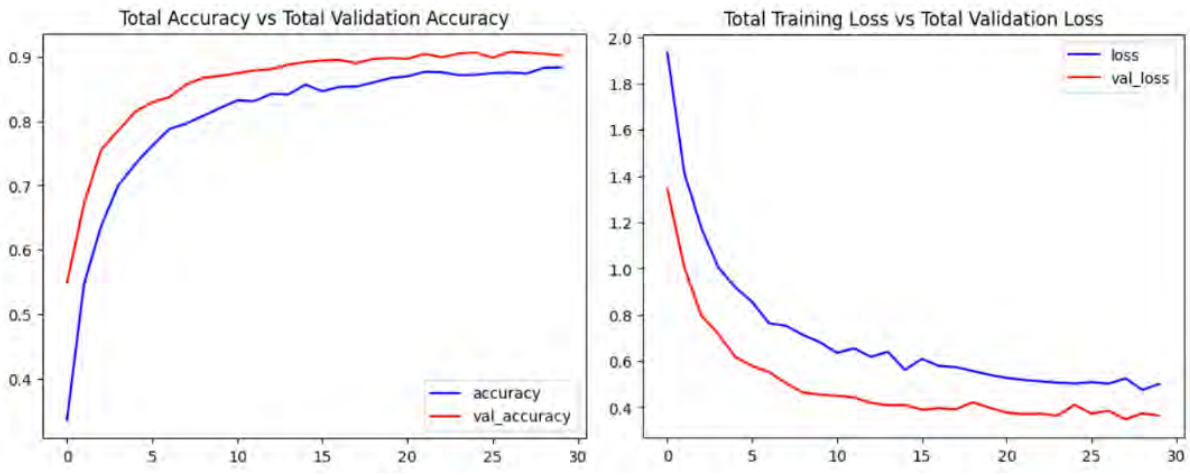

Figure 5.17: Training Accuracy and Loss for Normal Dataset

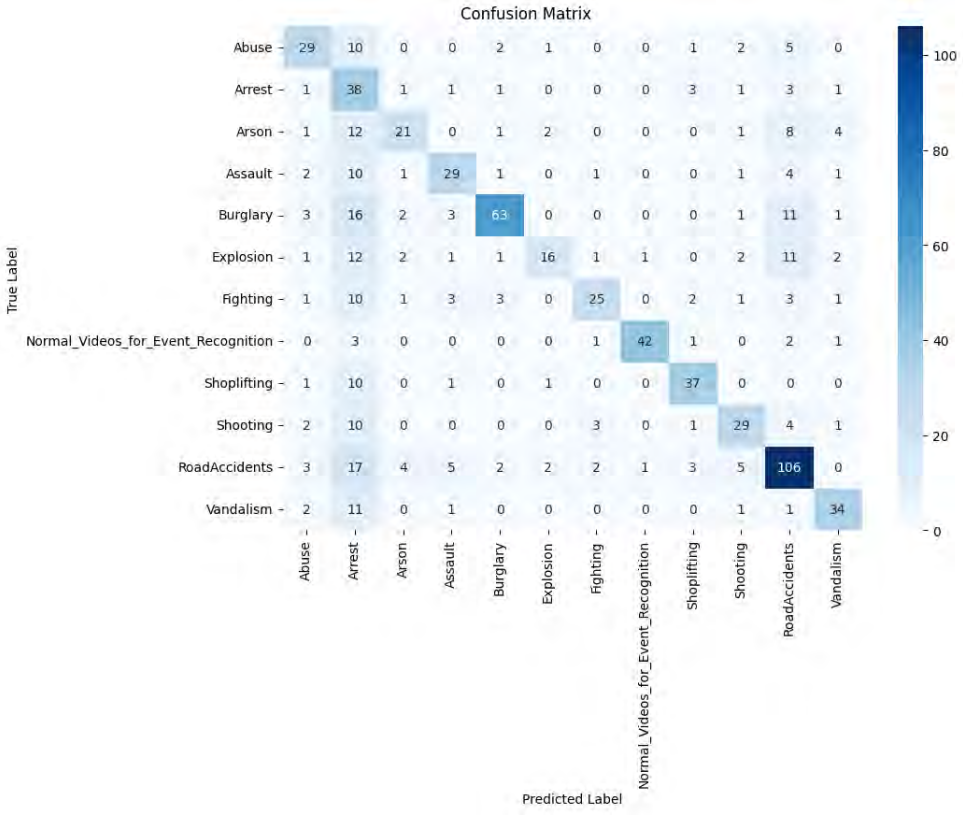

Figure 5.18: Confusion Matrix for Normal Dataset

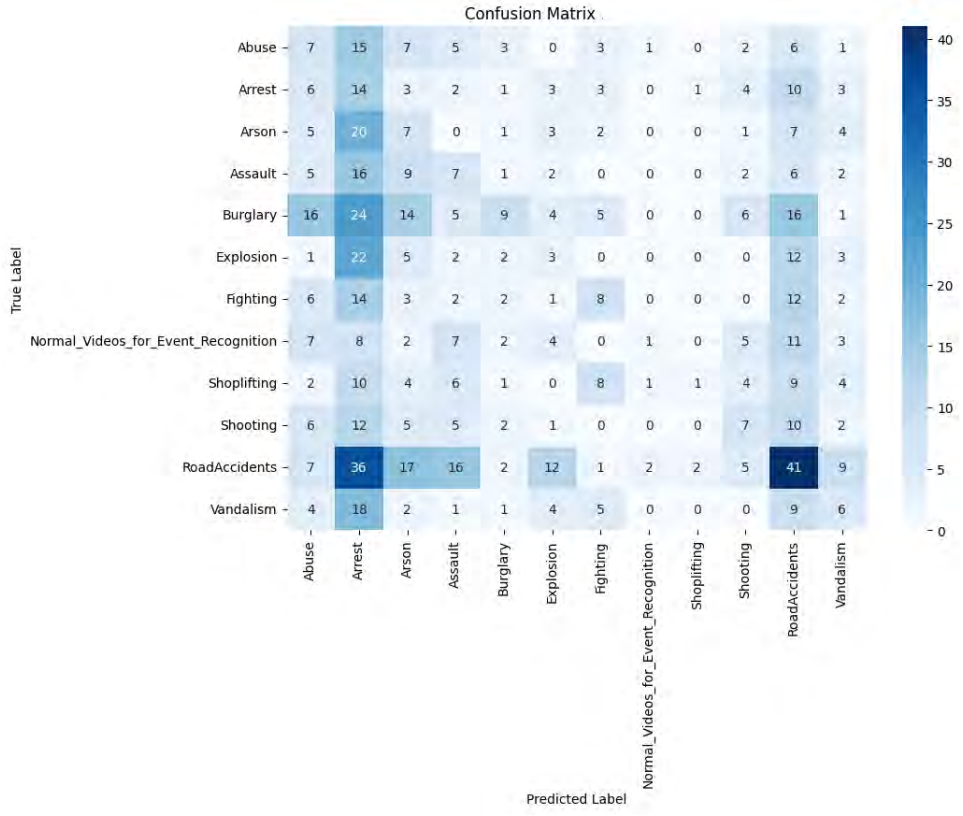

Figure 5.19: Confusion Matrix for Low-Light Dataset

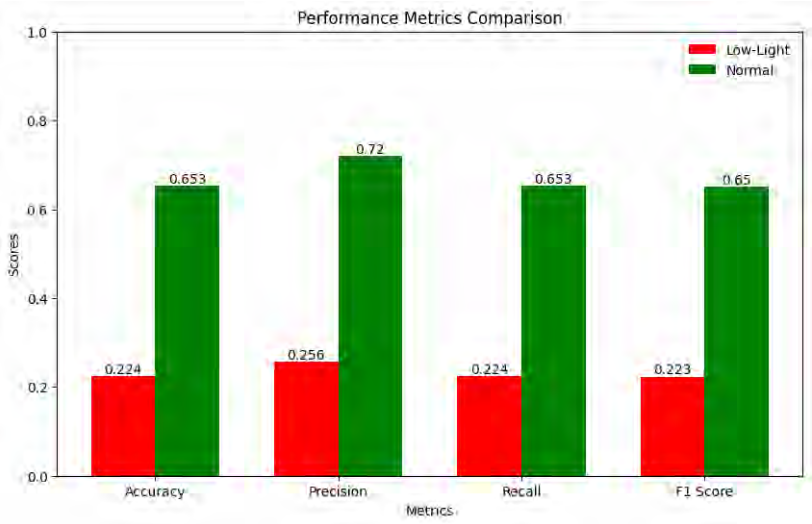

Figure 5.20: Accuracy, Precision, Recall, F1

5.3.2 Transfer Learning on Low Light Dataset and Testing on Normal and Low-Light Dataset

Now in the case of transfer learning, we loaded the MobileNet-LSTM model file and we froze all the layers of the model except the last 3 layers and we changed the learning rate from 0.002 to 0.003, and kept the optimizer same as before ran 30 epochs just as before. Then after training, we saved the model and ran it through the Normal and low-light datasets and got the below results

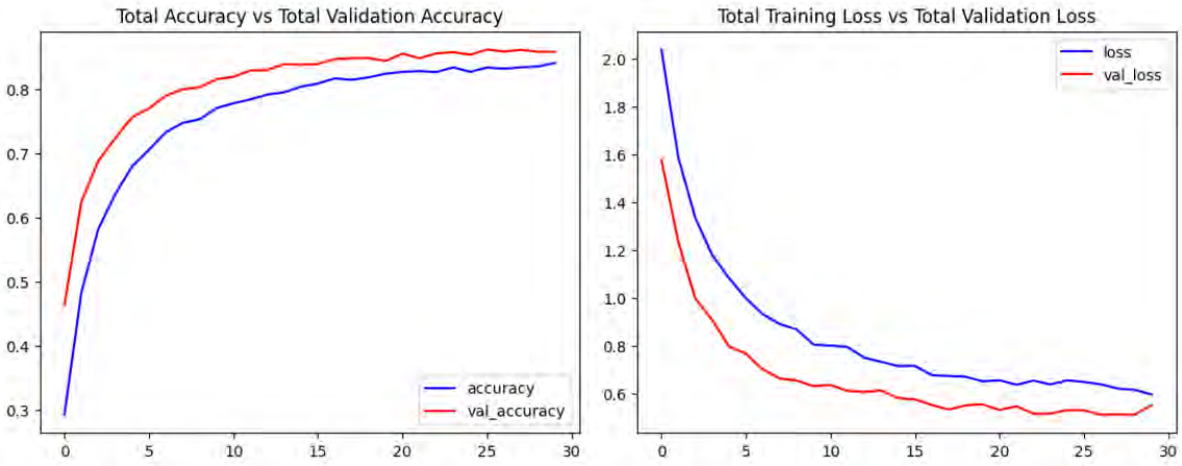

Figure 5.21: Training Accuracy and Loss for Low-Light Dataset

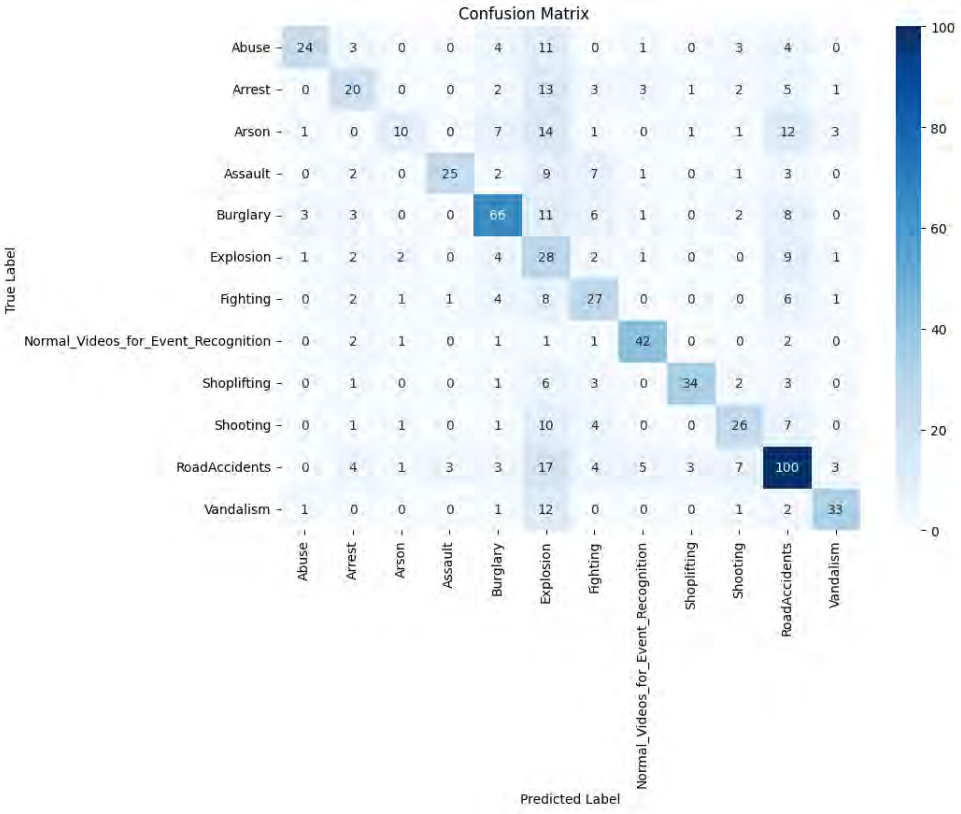

Figure 5.22: Confusion Matrix for Normal Dataset

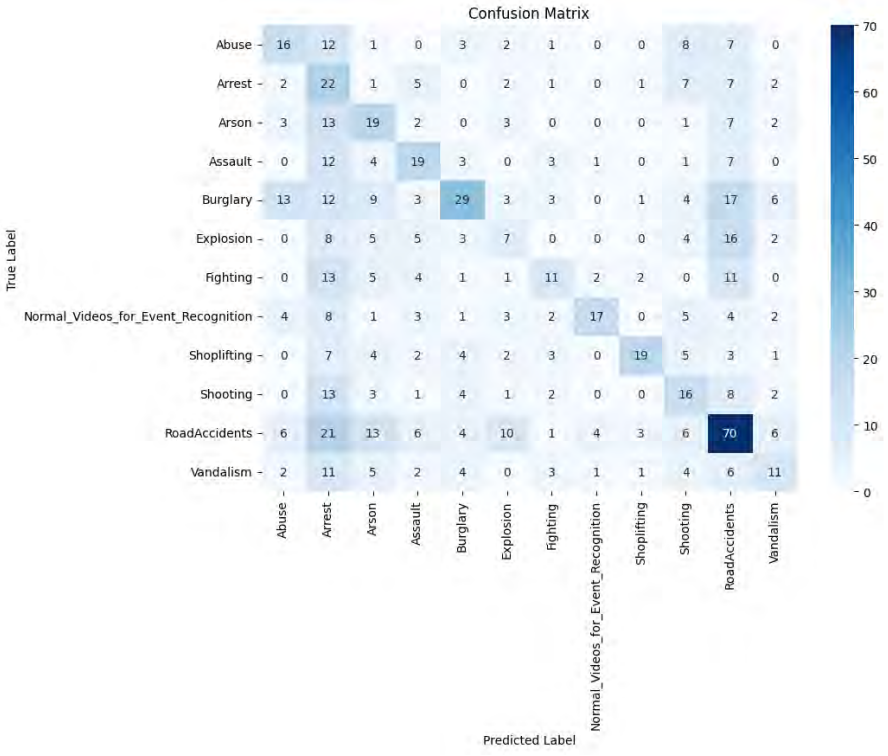

Figure 5.23: Confusion Matrix for Low-Light Dataset

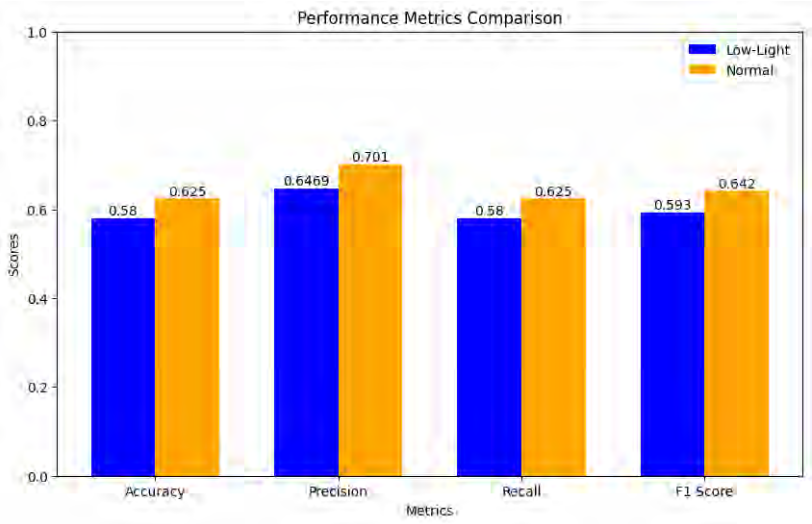

Figure 5.24: Accuracy, Precision, Recall, F1

5.4 Implementation and Result of NasNetMobile-LSTM

5.4.1 Training on Normal dataset and Testing on Normal and low-light Dataset

In the case of NasnetMobile-LSTM, we extracted the feature using NasnetMobile just like VGG-16 and MobilenetV2 and after that, we passed them to the LSTM Model, we trained our model using the normal dataset. We used the Nadam optimizer and the learning rate was 0.002 and we ran the model for 30 epochs. After the model is trained we save the model and run it through the normal and low-light dataset. Below is the following result

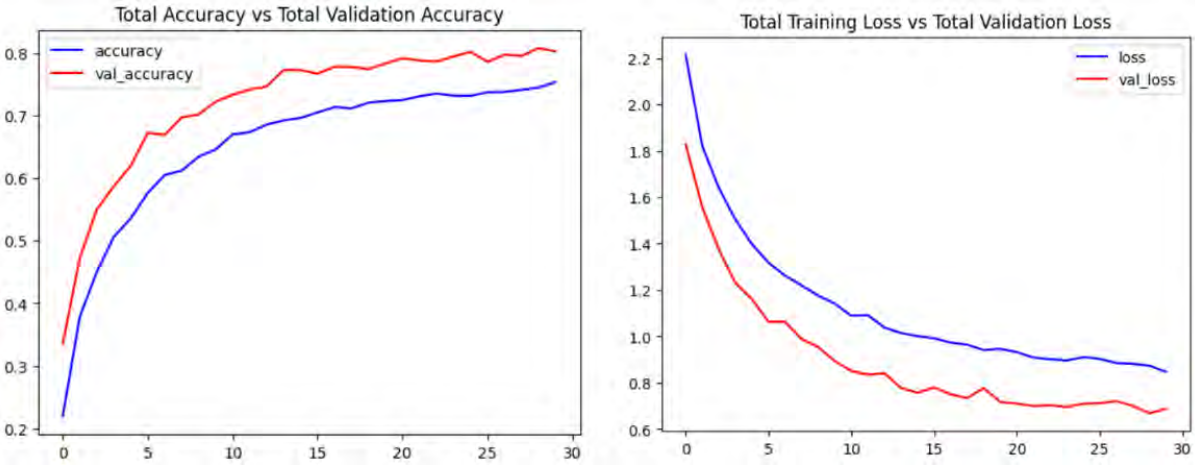

Figure 5.25: Training Accuracy and Loss for Normal Dataset

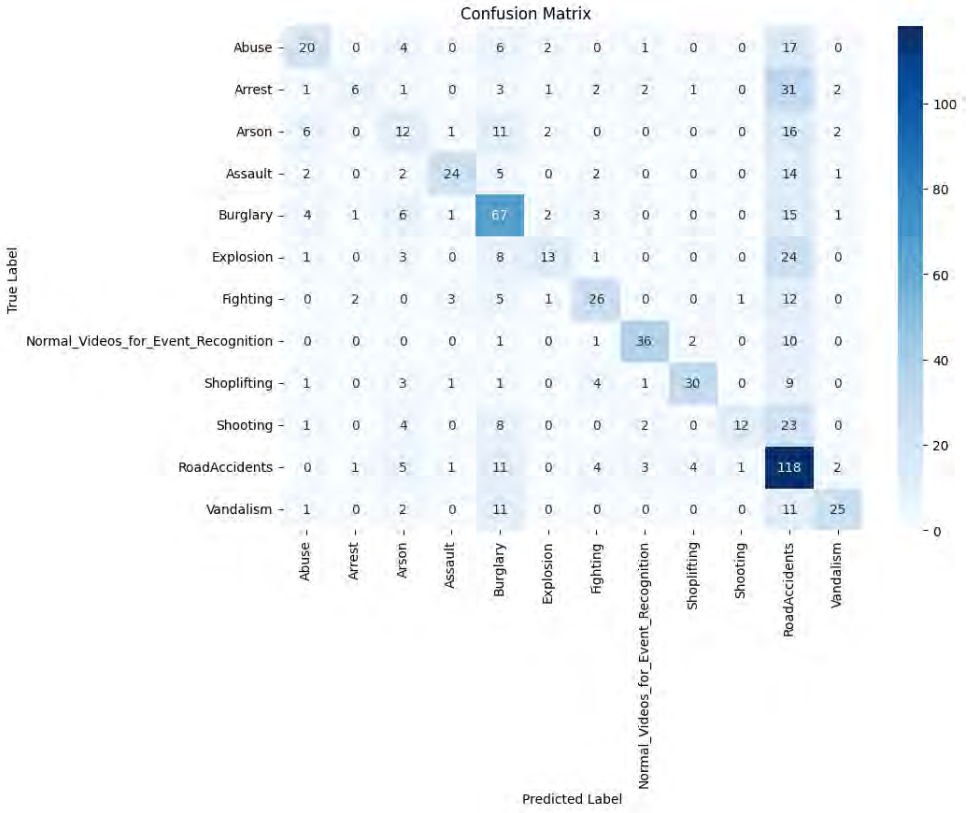

Figure 5.26: Confusion Matrix for Normal Dataset

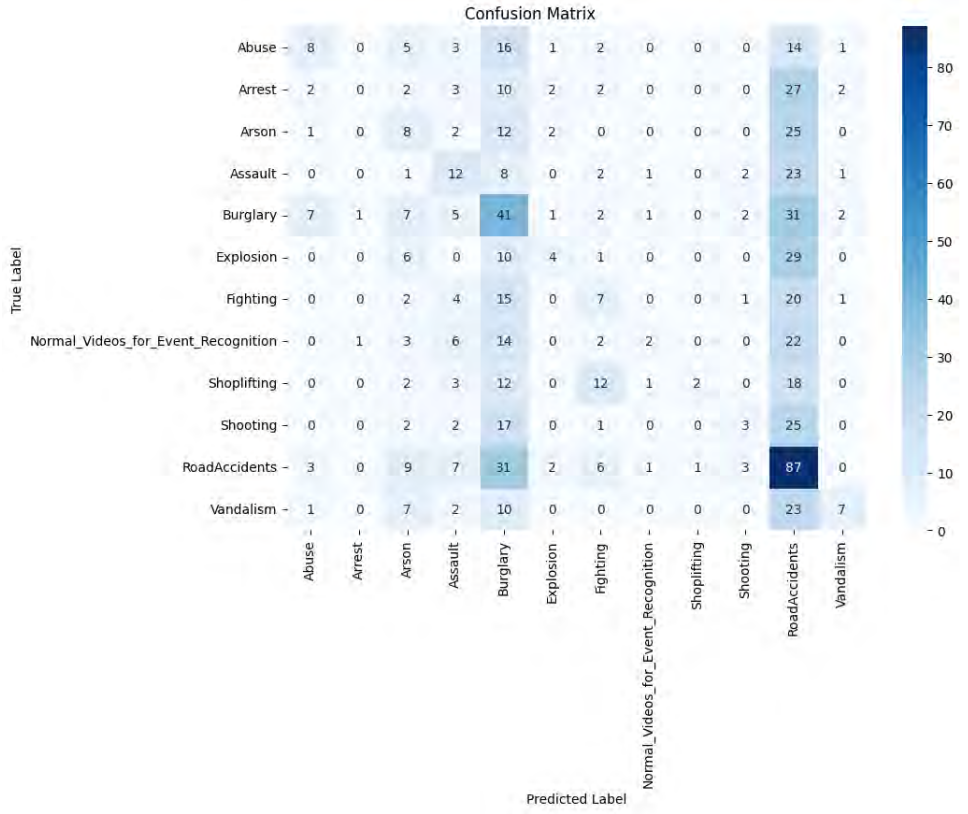

Figure 5.27: Confusion Matrix for Low-Light Dataset

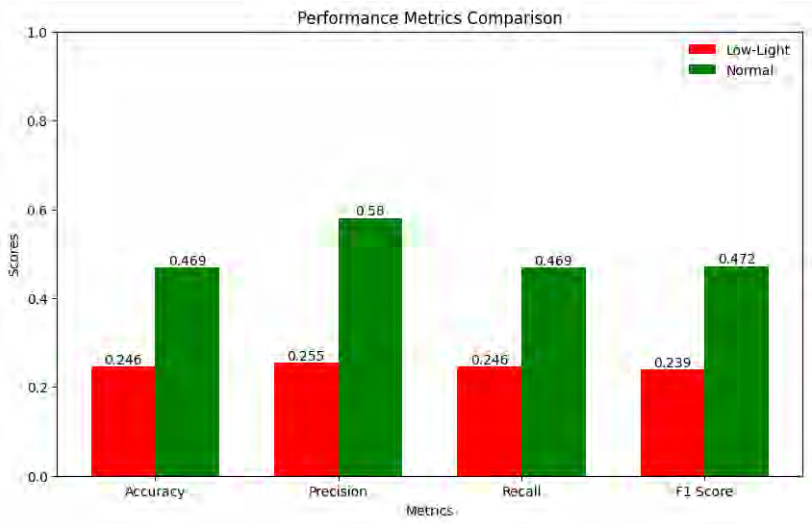

Figure 5.28: Accuracy, Precision, Recall, F1

5.4.2 Transfer Learning on Low Light Dataset and Testing on Normal and Low-Light Dataset

So we can see that the model failed to perform on the Low-Light dataset now as like before we will transfer learn our saved model on the Low-Light dataset. In the case of transfer learn we have disabled the initial layers of the Nasnet and kept only the last three reductions Normal and Reduction Cell for feature extracting then we froze all layers of LSTM except the last 3 layers and then we trained the model on the Low-Light dataset. Just like before we used a nadam optimizer with 0.002 learning and then we used 30 epochs for training. Below is the result of the transfer learning model for NasnetMobile

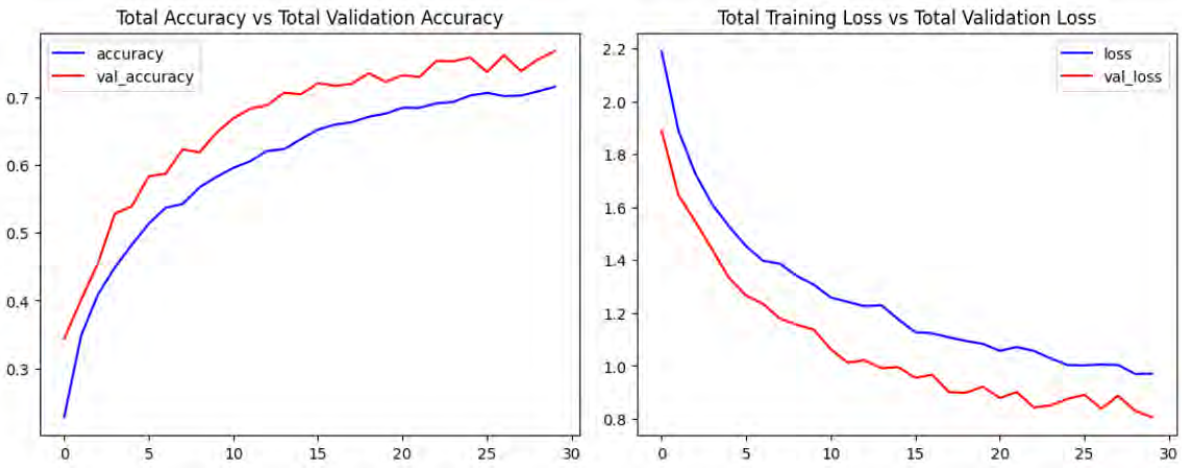

Figure 5.29: Training Accuracy and Loss for Low-Light Dataset

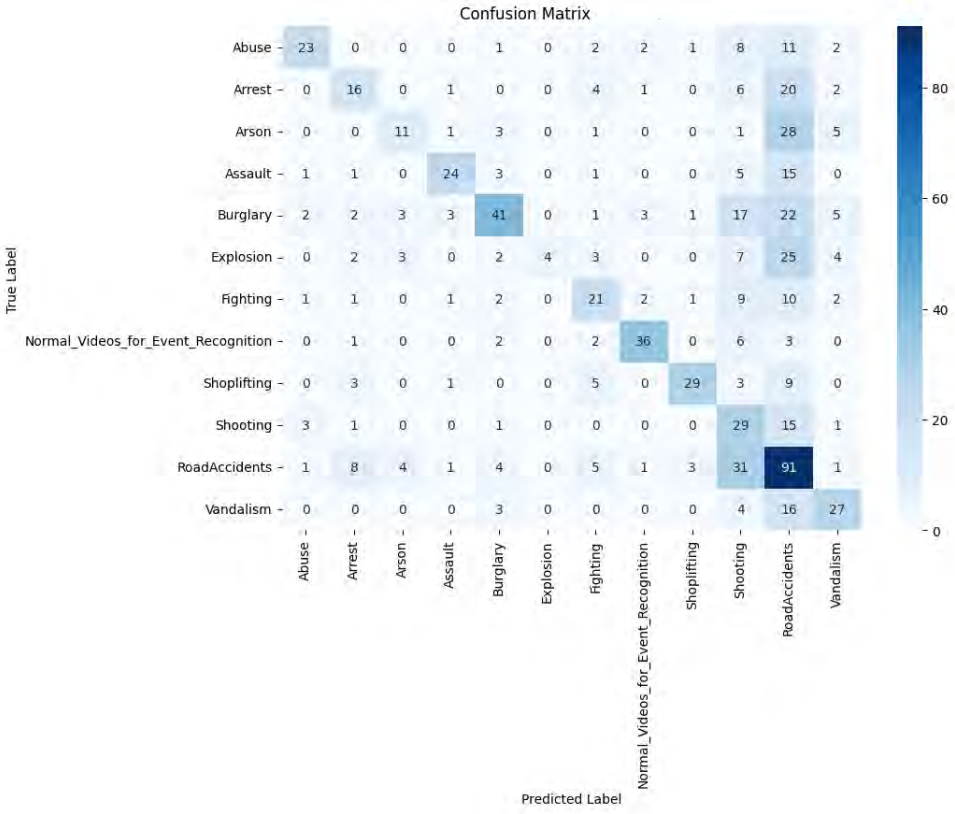

Figure 5.30: Confusion Matrix for Normal Dataset

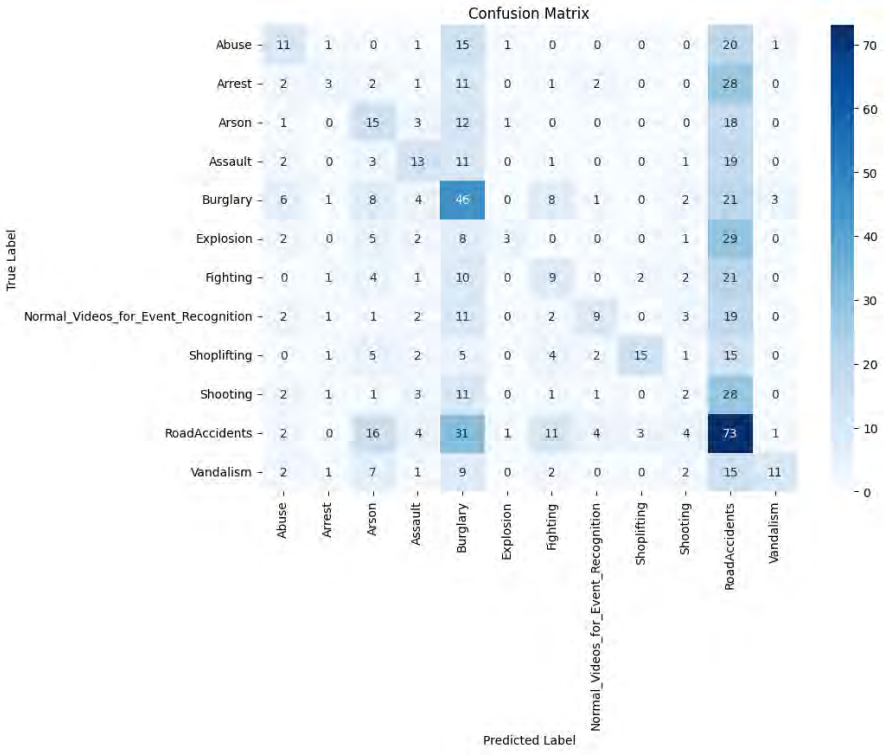

Figure 5.31: Confusion Matrix for Low-Light Dataset

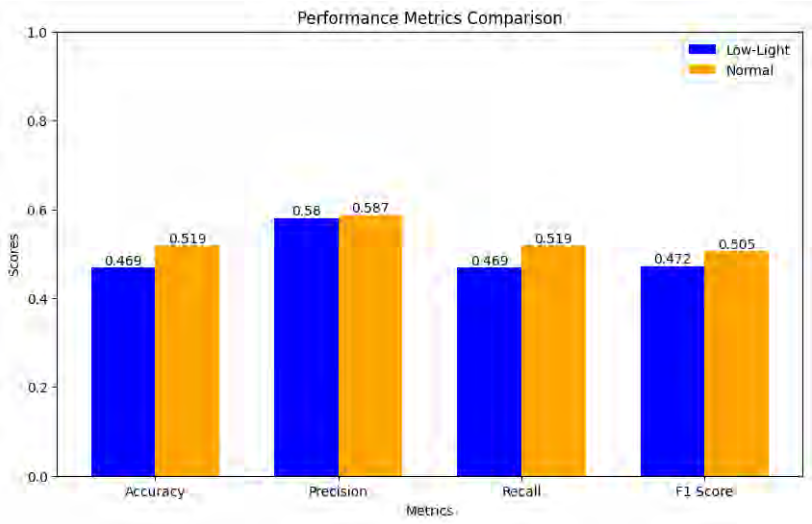

Figure 5.32: Accuracy, Precision, Recall, F1

5.5 Analysis

Now let's analyse our result and see what exactly happening here. So basically we first trained our models with the Normal UCF crime dataset without adding any extra pre-processing and then after training, we test them on the Normal dataset as well as the dataset we made with low-light conditions of the UCF crime dataset. From the result, we find out that the models are not able to perform up to the mark in low-light dataset but it is performing well in the Normal UCF dataset which is obvious as the model is trained on the Normal UCF dataset. Now what we need to do is train the model using the low-light dataset but also keep intact the information models learned from the UCF normal dataset and for that we frize up the layers of the models to retain the learning of the Normal UCF dataset and then add the newly learning components from the low-light dataset and finally will able to detect both type of condition successfully. So now let's analyze what performance we get from our implementation.

First we trained our CONVLSTM model with the Normal dataset and from that we got about 98.34% training accuracy and about 97.65% validation accuracy and in case of loss training loss was only 3.263% and validation loss was near 20% which is quite low. Now after that we run the model through the UCF normal dataset and Low-light dataset and got quite satisfactory result that both the model perform quite well through the model didn't perform that much well in case of the low-light dataset. The class-wise classification was lower than the normal dataset classification. Then we can see the video wise classification accuracy, precision, recall and f1 score where we can see that the UCF normal dataset was ahead of the low-light dataset. The score are visible in the fig:5.4 which show the clear difference.

Now let's see the performance when we transfer learned the model with low-light dataset. So after training the model with the low-light dataset, we got about 95% training accuracy and about validation accuracy was about 97%. The training loss was also quite moderate about 20% of training loss and 14% of validation loss.Now if we look at the confusion matrix we can see that the model did better classification

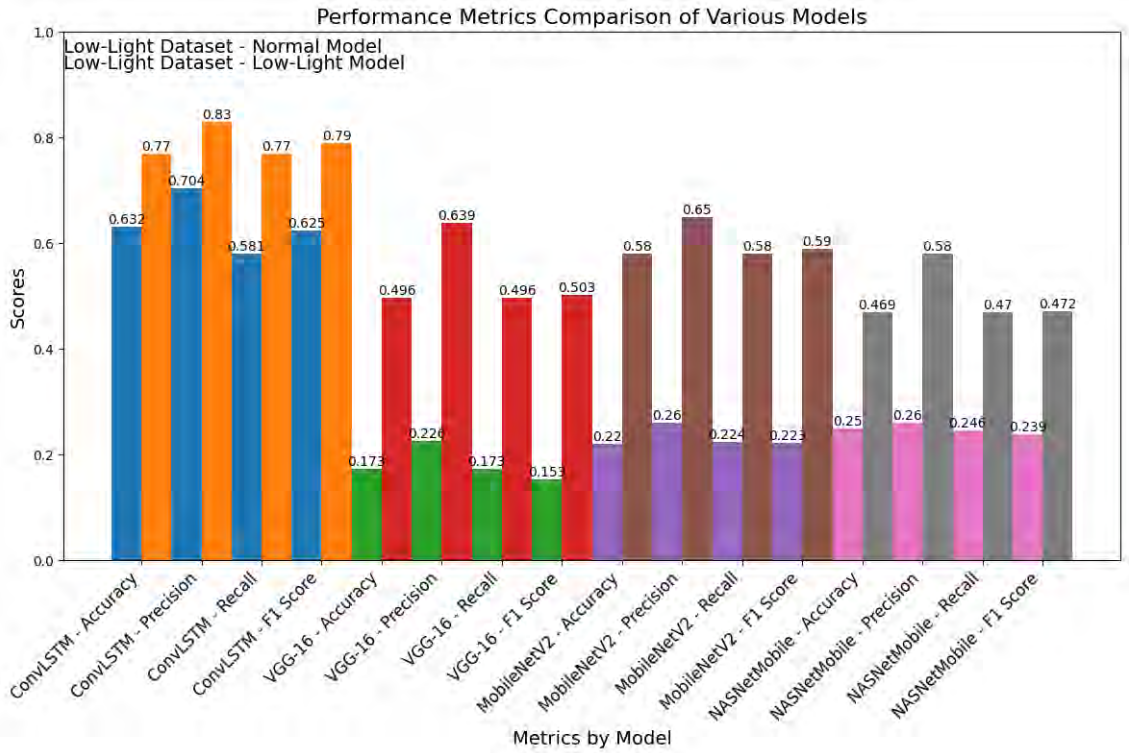

Figure 5.33: Overall Performance Analysis

at both of the model. Correct prediction per class has increased and it is much better than the model we trained only with UCF normal dataset.Now if we look at the accuracy, precision, recall and F1 score of the models performance in both the normal and low-light dataset (Fig:5.8) we can see that the performance improved significantly then the previous one. Especially in case of a low-light dataset the accuracy has jumped from 63.2 % to 77% which is really significant. Also in case precision, recall, and f1 it has seen significant jump in the result. So we can say that our transfer learning method is working and ConvLSTM can detect both the normal and low-light dataset better than before.

Now come to the VGG-16-LSTM, so just like CONVLSTM we trained our model in the Normal dataset fast and we got about 73.31% training accuracy and about 80% validation accuracy. The validation loss was about 95% and the the validation loss was about 72% which showed us the model is a bit underperforming in the Normal dataset. So after training the model, we run it through the Normal dataset and Low-Light condition dataset we see in the confusion matrix that the model couldn't able to detect and classify the low light condition at all but in the Normal dataset it performed well and classified video well though there are miscalculation also. The accuracy, precision, Recall and f1 score (Fig:5.12) justify the result we saw in the confusion matrixes.

In the case of transfer learning after training the model with the low-light dataset, we got about 75% training accuracy and about 80% validation accuracy and in case of loss the training loss was about 93.25% and validation loss was about 62.09% so we can see that the model perform sightly well than what we got from the normal dataset training. Now after that we run the model through the both datasets and see that the confusion matrix in the case of normal dataset has little but ill-

performed but still classified well and in the case of low-light dataset this time though the classification is not well enough but better than what we got in only normal dataset trained model this time it can classify significant amount of video through the misclassification rate is also high. The confusion matrix result reflects on the Accuracy, precision, recall and f1 graph though its performance decreases a little bit in case of the normal dataset it performed well in the low-light dataset. The detection accuracy increased from 17.3% to 49.6% which is a significant jump in performance.

In the case of MobilenetV2, After training the model with the Normal UCF crime dataset we got about 85.28% training accuracy and 87.10% validation accuracy and training loss was 58.32% and validation loss was about 0.428%. So the model trained well we can say that from the result now after we ran it through the dataset we can see it performed quite well in the Normal dataset but in the low-light dataset it performed poorly it wasn't able to classify at all. Kind of a similar case like VGG-16. In accuracy, Precision, Recall and f1 (fig: 5.20) the result of the confusion matrix reflects.

Now after this, we trained our model in a low-light dataset just like the previous models and then we got about 82.45% Training accuracy and about 85.11% validation accuracy. The training loss was about 62.31% and the validation loss was about 61.78% after that we ran the trained model on both of the datasets and the confusion matrix performed quite well. The confusion matrix of the normal dataset was quite the same as the previous one but the confusion matrix on the low-light dataset improved significantly as it was able to detect a significant amount of videos though there are also misclassification a lot but it is way better performed than what we saw with the normal dataset trained model. As we look at the accuracy, precision, recall, and f1 we can see a significant amount of improvement as the accuracy jumped from 22.4% to 58% (fig:-5.24), and the precision, recall, and f1 score also increased double as we compare both the figures. So here also we can see that the transfer learning is working significantly well.

At last we have NasNetmobile-LSTM where after training the model with the normal dataset we got about 76.36% training accuracy and about 79.39% validation accuracy and in case of loss training loss was about 91.12% and the validation loss was about 73.90% this shows that the model didn't train that well with the normal dataset. Now when we ran the model through the normal and low-light dataset we get the confusion matrix which shows it performed well in case of Normal dataset though there is lot of misclassification and the number of detected videos per class is also low for maximum classes but it still done quite well and in the case of the confusion matrix of the dataset it done what it's previous two models have dome completely failed to recognize the low-light videos. The accuracy, precision, Recall, and F1 (Fig:28) reflect the confusion matrix

Now if we talk about transfer learning after we train the model on a low-light dataset we get about 70% of training accuracy and about 77.32% of validation accuracy and in case of loss, we get about 1.15% of training loss and about 80% of validation loss which is quite high. So after running the model through both normal and low-light

datasets we get the confusion matrix for both the datasets and we see that it didn't perform much well at all though the low light condition performed a little bit better than the previous one and able to classify some amount of videos but that too not for all classes. Only particular classes were predicted well. The accuracy, precision, recall and F1 score also reflect the result of the confusion matrix (5.32 Figure) This comparison analysis shows that the main objective of our model which was to be able to detect and classify and detect videos under low light conditions has been achieved. Though it is not significant but it is a big improvement in this sector. From the analysis, one thing is clear ConvLSTM is the best-performing model. We can see that if performed well in the low-light dataset even with not trained with the low-light dataset also we see that after trained with the low-light dataset it performed really well. In case of the other three model one thing is clear the model improved the performance under the transfer learning as though it is not good but the improvements was great and it shows us that we can even further improve the things. so for testing our work we have load our transfer learned CONVLSTM2D model and run it on a video and we get the following result in case of normal video and low-light video from the dataset

Figure 5.34: Testing model under Normal and Low-Light video

Chapter 6

Limitation, Conclusion and Future Work

6.1 Limitations

While our approach to criminal activity identification in low-light circumstances has demonstrated promising results, yet some drawbacks need to be addressed:

While implementing the preprocessing approaches, enhanced the visibility and video quality, the models still struggled in severe low-light circumstances where the video was nearly or completely dark. Even advanced deep learning systems have difficulty detecting important characteristics in video material that is fully dark. This shortcoming can result in missed detections, in which the model fails to identify a criminal action accurately. Future research may need to look into further developed image-enhancing techniques or alternative sensor technologies (such as infrared or thermal imaging) to overcome this issue. Low-light surroundings are prone to noise and diminished visual detail, which can confuse models and result in false positives when typical behaviors are incorrectly categorized as illegal. For instance, shadows or bad lighting may distort actions in a way that might appear deceptive to the model. This sensitivity to noise remains a key concern, since it may result in over-detection and excessive alarms, compromising the system's reliability in practical applications. Moreover, Despite using transfer learning, we had significant difficulty maximizing the learning rate for models such as NASNetMobile, MobileNetV2, and VGG-16 during training. The learning rate remained stable, preventing the models from enhancing their performance during training. This issue reduced the models' capacity to converge quickly and reach optimal performance, slowing down the training process. Despite the use of pre-trained weights, inefficient learning rate adjustment resulted in extended training times and inferior performance at specific epochs.
To overcome the slow learning rate, we required to switch from the Nadam optimizer to the Adam optimizer to achieve faster convergence. While Nadam (Nesterov-accelerated Adam) is commonly used for its theoretical advantages in adaptive learning rate, it did not produce the required results in this situation. The Adam optimizer enabled faster convergence and a more reliable learning process. However, while this optimization swap improves efficiency, it also highlights a shortcoming in some models' ability to adjust to learning rate and optimizer setups.
Finally, During the training phase, considerable computational and hardware limita-

tions were encountered, given the complexity and scale of the deep learning models used. Models such as VGG-16, NASNetMobile, and the Conv-LSTM hybrid are resource-intensive and require significant GPU power for handling video data, particularly when applied to big datasets like the modified UCF Crime dataset. The increased memory and processing requirements for these models resulted in lengthier training periods, even when using powerful cloud GPUs.

6.2 Conclusion

To conclude, This research proposes an approach to detecting criminal activity in low-light videos using advanced deep-learning neural networks. As surveillance systems become more integrated into public safety and security, the ability to reliably recognize suspicious behavior in complicated lighting settings is critical.

In this study, we investigated the efficacy of numerous cutting-edge deep learning architectures, including VGG-16, MobileNetV2, NASNetMobile, and a hybrid CONV-LSTM model. We wanted to improve the models' capacity to use both spatial and temporal variables by using sophisticated video preparation approaches aimed at improving image quality. We also converted the widely used UCF Crime dataset to simulate low-light conditions. This allows us to test the effectiveness of our models under realistic low-light scenarios, ensuring that the results apply to real-world surveillance systems. Using transfer learning, we were able to adapt these pre-trained models to our specific purpose, considerably decreasing the requirement for retraining while still producing encouraging results. Our results show that integrating spatial and temporal feature extraction approaches improves the accuracy of criminal activity identification in low-light conditions. Despite difficulties which included concerns with learning rate optimization, and noise sensitivity in extreme low-light conditions, our models indicated the potential for reliable criminal activity detection. The use of transfer learning proved advantageous, allowing us to leverage prior knowledge while addressing the limits of our dataset and model design. However, this study also identified many constraints that must be addressed in order to develop the area further. The limits encountered throughout the training process, notably those involving computational resources and learning rate slowdown, highlight the need for more efficient algorithms and model architectures. The findings show that, while traditional models may struggle with accuracy in low-light conditions, advanced deep-learning models can greatly improve detection skills.

Notably, the Conv-LSTM model appeared as the most efficient architecture, successfully combining spatial feature extraction and temporal analysis to boost detection rates of suspicious activity. This hybrid technique provides a more detailed understanding of behaviors throughout time and outperforms the other architectures, making it ideal for real-time surveillance applications. Overall, this research incorporates the expanding field of research on the interface of computer vision, deep learning, and security applications. By discovering efficient ways for detecting criminal and suspicious behavior in low-light situations, we seek to pave the path for the creation of more intelligent and responsive surveillance systems. These innovations can improve public safety measures while keeping ethical issues at the forefront of deployment strategies.

In conclusion, this study not only tackles the immediate issues of low-light video categorization, but also provides the framework for future advancements in surveillance technology, ultimately aiming for systems that are more accurate, efficient, and ethical in their deployment.

6.3 Future Work

Based on the findings and limits reported in this study, several crucial areas for future research might be pursued to improve the effectiveness of criminal activity detection in low-light circumstances. Despite the advancement and acceleration of new technology, analyzing anomalies and suspicious human behavior under completely dark circumstances is still in the progress in computer vision research area. First, there is a need for better preprocessing approaches that make use of modern picture-enhancing methods, such as the integration of infrared and thermal imaging technologies. Such developments should considerably improve the models' capacity to extract useful characteristics from extremely low-light conditions when typical RGB video input is insufficiently detailed. Another key area is the combining of multi-modal data sources, such as audio feeds and sensor data, to gain a more complete knowledge of criminal behavior and improve detection accuracy. Finally, ethical considerations must be emphasized when deploying automatic surveillance systems; future research should focus on building frameworks that assure responsible use, preserve individual privacy, and eliminate model biases. By incorporating explainable AI (XAI) approaches, we may increase openness in these systems' decision-making processes, promoting confidence among stakeholders and assuring accountability in key applications such as public safety. Overall, these efforts will help to develop stronger, accurate, and ethical surveillance technology.

Bibliography

[1] S. Rahman, J. See, and C. C. Ho, "Action recognition in low quality videos by jointly using shape, motion and texture features," in *2015 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, IEEE, 2015, pp. 83–88.

[2] X. Wang, L. Gao, J. Song, and H. Shen, "Beyond frame-level cnn: Saliency-aware 3-d cnn with lstm for video action recognition," *IEEE Signal Processing Letters*, vol. 24, no. 4, pp. 510–514, 2017. DOI: 10.1109/LSP.2016.2611485.

[3] W. Sultani, C. Chen, and M. Shah, "Real-world anomaly detection in surveillance videos," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 6479–6488.

[4] A. Ulhaq, "Action recognition in the dark via deep representation learning," in *2018 IEEE International Conference on Image Processing, Applications and Systems (IPAS)*, IEEE, 2018, pp. 131–136.

[5] R. Nawaratne, D. Alahakoon, D. De Silva, and X. Yu, "Spatiotemporal anomaly detection using deep learning for real-time video surveillance," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 1, pp. 393–402, 2019.

[6] N. Tufek, M. Yalcin, M. Altintas, F. Kalaoglu, Y. Li, and S. K. Bahadir, "Human action recognition using deep learning methods on limited sensory data," *IEEE Sensors Journal*, vol. 20, no. 6, pp. 3101–3112, 2019.

[7] T. Wang, M. Qiao, A. Zhu, G. Shan, and H. Snoussi, "Abnormal event detection via the analysis of multi-frame optical flow information," *Frontiers of Computer Science*, vol. 14, Aug. 2019. DOI: 10.1007/s11704-018-7407-3.

[8] K. Zhang and W. Ling, "Joint motion information extraction and human behavior recognition in video based on deep learning," *IEEE Sensors Journal*, vol. 20, no. 20, pp. 11 919–11 926, 2019.

[9] J. T. Zhou, J. Du, H. Zhu, X. Peng, Y. Liu, and R. S. M. Goh, "Anomalynet: An anomaly detection network for video surveillance," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 10, pp. 2537–2550, 2019.

[10] R. J. Franklin, V. Dabbagol, *et al.*, "Anomaly detection in videos for video surveillance applications using neural networks," in *2020 Fourth International Conference on Inventive Systems and Control (ICISC)*, IEEE, 2020, pp. 632–637.

[11] R.-H. Hwang, M.-C. Peng, C.-W. Huang, P.-C. Lin, and V.-L. Nguyen, "An unsupervised deep learning model for early network traffic anomaly detection," *IEEE Access*, vol. 8, pp. 30 387–30 399, 2020.

[12] G. Pang, C. Yan, C. Shen, A. v. d. Hengel, and X. Bai, "Self-trained deep ordinal regression for end-to-end video anomaly detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 12 173–12 182.

[13] Z. Yu and W. Q. Yan, "Human action recognition using deep learning methods," in *2020 35th International Conference on Image and Vision Computing New Zealand (IVCNZ)*, IEEE, 2020, pp. 1–6.

[14] B. Degardin and H. Proenca, "Human behavior analysis: A survey on action recognition," *Applied Sciences*, vol. 11, no. 18, p. 8324, 2021.

[15] C. Li, C. Guo, L. Han, *et al.*, *Low-light image and video enhancement using deep learning: A survey*, 2021. arXiv: 2104.10729 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2104.10729.

[16] X. Ma, J. Wu, S. Xue, *et al.*, "A comprehensive survey on graph anomaly detection with deep learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 12, pp. 12 012–12 038, 2021.

[17] R. Nayak, U. C. Pati, and S. K. Das, "A comprehensive review on deep learning-based methods for video anomaly detection," *Image and Vision Computing*, vol. 106, p. 104 078, 2021.

[18] W. Ullah, A. Ullah, I. U. Haq, K. Muhammad, M. Sajjad, and S. W. Baik, "Cnn features with bi-directional lstm for real-time anomaly detection in surveillance networks," *Multimedia tools and applications*, vol. 80, pp. 16 979–16 995, 2021.

[19] Y. Zhang, Y. Chen, J. Wang, and Z. Pan, "Unsupervised deep anomaly detection for multi-sensor time-series signals," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 2, pp. 2118–2132, 2021.

[20] R. Al Sobbahi and J. Tekli, "Comparing deep learning models for low-light natural scene image enhancement and their impact on object detection and classification: Overview, empirical evaluation, and challenges," *Signal Processing: Image Communication*, vol. 109, p. 116 848, 2022.

[21] C.-Y. Luo, S.-Y. Cheng, H. Xu, and P. Li, "Human behavior recognition model based on improved efficientnet," *Procedia computer science*, vol. 199, pp. 369–376, 2022.

[22] D. R. Patrikar and M. R. Parate, "Anomaly detection using edge computing in video surveillance system," *International Journal of Multimedia Information Retrieval*, vol. 11, no. 2, pp. 85–110, 2022.

[23] K. Hu, J. Jin, F. Zheng, L. Weng, and Y. Ding, "Overview of behavior recognition based on deep learning," *Artificial intelligence review*, vol. 56, no. 3, pp. 1833–1865, 2023.

[24] A. Hussain, S. U. Khan, N. Khan, I. Rida, M. Alharbi, and S. W. Baik, "Low-light aware framework for human activity recognition via optimized dual stream parallel network," *Alexandria Engineering Journal*, vol. 74, pp. 569–583, 2023.

[25] I.-C. Hwang and H.-S. Kang, "Anomaly detection based on a 3d convolutional neural network combining convolutional block attention module using merged frames," *Sensors*, vol. 23, no. 23, p. 9616, 2023.

[26] S. Jebur, K. Hussein, and H. Hoomod, "Abnormal behavior detection in video surveillance using inception-v3 transfer learning approaches," *IRAQI JOURNAL OF COMPUTERS, COMMUNICATIONS, CONTROL AND SYSTEMS ENGINEERING*, vol. 23, no. 2, pp. 210–221, 2023, ISSN: 1811-9212. DOI: https://doi.org/10.33103/uot.ijccce.23.2.16. eprint: https://ijccce.uotechnology.edu.iq/article_178247_fa40556919c4e6a798516f569c300340.pdf. [Online]. Available: https://ijccce.uotechnology.edu.iq/article_178247.html.

[27] S. Lee, J. Rim, B. Jeong, *et al.*, "Human pose estimation in extremely low-light conditions," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 704–714.

[28] M. Mukto, M. M. Mahmud, I. Haque, *et al.*, "Design of a real-time crime monitoring system using deep learning techniques," *sensors*, 2023.

[29] V.-A. Nguyen and S. G. Kong, "Multimodal feature fusion for illumination-invariant recognition of abnormal human behaviors," *Information Fusion*, vol. 100, p. 101 949, 2023.

[30] M. Mao, A. Lee, and M. Hong, "Deep learning innovations in video classification: A survey on techniques and dataset evaluations," *Electronics*, vol. 13, no. 14, 2024, ISSN: 2079-9292. DOI: 10.3390/electronics13142732. [Online]. Available: https://www.mdpi.com/2079-9292/13/14/2732.

[31] K. Rezaee, S. M. Rezakhani, M. R. Khosravi, and M. K. Moghimi, "A survey on deep learning-based real-time crowd anomaly detection for secure distributed video surveillance," *Personal and Ubiquitous Computing*, vol. 28, no. 1, pp. 135–151, 2024.