

Leveraging Sequential Deep Learning Models for Detecting Multitude of Human Action Categories

by

KAZI AL REFAT PRANTA

23341120

FAHAD MOHAMMAD REJWANUL ISLAM

20101443

KHANDAKAR FAHIM AHMED

23241110

PRINCE SAHA

19301212

NAIMUR RAHMAN

20101484

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
September 2023

© 2023. Brac University
All rights reserved.

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:



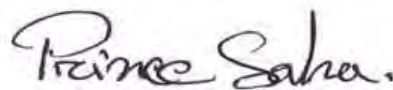
KAZI AL REFAT PRANTA
23341120




FAHAD MOHAMMAD REJWANUL ISLAM
20101443



KHANDAKAR FAHIM AHMED
23241110



PRINCE SAHA
19301212



NAIMUR RAHMAN
20101484

Approval

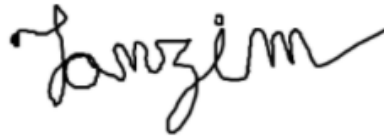
The thesis titled “Multiple Human Action Recognition From Video Data Using Deep learning” submitted by

1. KAZI AL REFAT PRANTA(23341120)
2. FAHAD MOHAMMAD REJWANUL ISLAM(20101443)
3. KHANDAKAR FAHIM AHMED(23241110)
4. PRINCE SAHA(19301212)
5. NAIMUR RAHMAN(20101484)

Of Spring 2023, has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on September 18, 2023.

Examining Committee:

Supervisor:
(Member)



Tanzim Reza
Lecturer
Department of Computer Science and Engineering
Brac University

Co-Supervisor:
(Member)



Rafeed Rahman
Lecturer
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

In today's world, where science and technology are constantly evolving day by day, people are drawn to tangible experiences and visual representations. There's a growing effort to teach machines about human movements and postures to enable smart decision-making. This has led to increased interest in the field of human action recognition (HAR) among researchers globally. Our research focuses on implementing advanced technologies to address criminal activities, specifically emphasizing Human Activity Recognition (HAR). Moreover, our dataset includes 1275 videos, covering 20 different actions involving both violent and non-violent behaviors. In addition, we have developed a pipeline that utilizes YOLO-v8 to extract background, followed by models for accurate video classification. two models, conv-lstm and lrcn, were incorporated into our deep learning pipeline. Through our observations, we found that the LRCN model outperformed the other model, achieving an accuracy of 62% and an F1 score of 60% for the 20 classes, for 17 classes an accuracy of 63% and an F1 score of 66%. for binary classification LRCN got accuracy of 88% and an F1 score of 87%. Our research focusses the potential of advanced technologies to significantly improve Human Activity Recognition (HAR) in addressing various aspects of criminal activities in real-time scenario. This marks a substantial step forward in intelligent decision-making and public safety.

Keywords: Human Action Recognition; Machine Learning; Convolutional Long Short-Term Memory; Human Motions; Intelligent Decision-making; YOLO-v8; Long-term recurrent convolutional network; Visual Representations

Table of Contents

Declaration	i
Approval	ii
Ethics Statement	iv
Abstract	iv
Dedication	1
Table of Contents	1
1 Introduction	2
1.1 Research Problem	2
1.2 Research Objectives	3
2 Related Work	5
3 Data Collection and Analysis	12
3.1 Data collection	12
3.2 Data Preprocessing	12
3.3 Data Analysis	14
4 Model Architecture	18
4.1 Methodology	18
4.2 You Only Look Once v-8	18
4.3 Convolutional Long Short-Term Memory	18
4.4 LRCN	21
5 Result Analysis	24
5.1 ConvLSTM	24
5.2 LRCN	27
5.3 Comparison	32
6 Conclusion	37
Bibliography	40

Chapter 1

Introduction

Human action recognition or HAR is a very interesting field for research. In many countries specially in Bangladesh we can see that there can be many suspicious activities can be found such as- snatching, stealing, hitting etc. And there can be many violent acts we can find in our society. So, it is necessary for us to have a system which will be able to detect the suspicious acts or criminal acts. In many advanced country or technologically advanced country has some automated system which can detects human activity. For example, USA has that system where if a man try to do some criminal activity , the act will directly be generated and will be sent to their police office. So, we can assume that how the human act detection is important in the modern era. In some mid level country like Bangladesh where crime acts are maximum here it is necessary to deploy the system. We know that our country is technologically not that advanced so this recognition system must have to be usable for the controller. Also, there is another major point which is this human action detection can also play a vital role in monitoring a patient. For example, a patient is in life support here we can deploy the system in order to monitor the patient acts.[2] So, we already have major two ponts of this reasearch one is to detect the violent acts and the other one is to monitor patient acts. In this paper, we deal with some video data due to detection of human action where we not only detected actions but also detected violent and non -violent acts using some models using deep learning. In computer vision, this system is very necessary to deploy in order to detect vioolent acts as well as detecting patient acts. Therefore, HAR or human action recognition is very important to deploy in modern era.

1.1 Research Problem

This research is all about making the computers better at figuring out what people are doing in videos. The main aim is to improve accuracy by using deep learning, which is like teaching computers to learn on their own. The focus is on by making the computer understand different actions even when things like appearances, backgrounds, and viewpoints change. The main goal is to do this in real-time, meaning really quickly, while using data efficiently. All of these improvements are very important to help make computers better at understanding human actions, especially in areas like computer vision and human-computer interaction.

By detecting what people are doing in videos is important and necessary for some sectors such as- security systems, how we interact with computers, summarizing

videos, and analyzing sports. But it's tricky because people look different, backgrounds change, and sometimes things block the view or get in the way. This paper is trying to build a reliable system using deep learning to recognize various human actions from videos. The main focus will be on spotting violent actions accurately in real-life situations.

Traditional methods for spotting human actions usually rely on manually created rules, which might not capture all the details in videos. This research main target is to solve this problem by using deep learning to automatically by finding important features directly from the raw video data, making action recognition more accurate. Identification of human movements correctly is tough because people can look very different, and the way things are recorded can change how actions appear. This research is all about creating deep learning models that can handle these variations well, making action recognition more precise and useful for us.

Real-world situations add challenges such as- blocking the view or actions happening quickly, making it hard for computers to understand. The main goal of this project is to create deep learning models that can deal with these challenges, improving the ability to detect actions.

By creating deep learning models for recognition of actions usually needs a lot of labeled data, which means videos that are already marked with what actions are happening. Getting this data can be time-consuming and expensive. This study is exploring ways like data augmentation and semi-supervised learning to overcome this problem and make deep learning models better even with limited labeled data. Faster and real-time identification of human actions is crucial for things like security systems in fast-changing environments. This research wants to create smart systems that can recognize actions in real-time while still being accurate. So, this includes due to figuring out ways to make models smaller and less complex.

In simple terms, this research asks, **"How can we use deep learning to make computers better at understanding different human activities in videos, making it more accurate, faster, and improving computer vision?"**

The primary goal of our is to recognize human actions from videos or real time data using deep learning models and solve related challenges. The hope is that the findings will contribute to areas like artificial intelligence, computer vision, and how we interact with computers. This contribution aims to create a system that can quickly and accurately understand human activities in different and complex situations.

1.2 Research Objectives

The aim of our research is to introduces a human activity recognition system that can accurately recognise a wide range of actions in different movies. This objective will be accomplished by using many models specifically designed for certain conditions. We propose using Convolutional Neural Networks (CNN) which is used for extracting visual insights and Long Short-Term Memory (LSTM) which is used to learn temporal information, with the option of further including Recurrent Neural Networks (RNN) and Gated Recurrent Units (GRU). The main motive is to identify human acts from video dataset. The system uses established models to identify and categorize these movements which is either normal or abnormal, based on the specific circumstances of the person.

The research objectives are outlined as follows:

1. We need to gain a comprehensive understanding of the methodologies used for identifying human activities (HAR) and their functioning.
2. We need to develop a human action recognition system using cnn and lstm. 3. We must obtain a comprehensive comprehension of the fundamental ideas behind cnn along with lstm models.
4. We must conduct a thorough evaluation of the implemented model.
5. And proposing architectures to enhance the model's efficiency..

Chapter 2

Related Work

The Research conducted by the authors Liu et al. in which their methodology consisted of two different segments. One corresponds to VMHI, while the other pertains to FRGB. The VGG-16 CNN model architecture was used to analyze motion pictures in historical context for VMHI, whereas the Faster CNN was applied for FRGB. The latter included VGG-16, Kalman filter, RPN, pooling of ROI, and a layers of classification. The integration of this architecture resulted in the production of class labels and enhanced the ability to identify activities. The study used the Weizmann, Kth, and Utinteraction databases. The reference is from Liu et al. (2021). [13].

Jaouedi et al. conducted a study on motion tracking using the GRNN architecture. The video data was entered and the movements or activities were monitored using the Kalman filter and Gaussian mixture model (GMM). Subsequently, the tensor was fed into the GRNN model, which is structured with layers for input, hidden components (employing GRU), and output. The Kth and Ucf101 datasets were used to train this video classification model. The reference is from Jaouedi et al. (2020). [5]

Tsai et al. conducted a study on multi-person activity detection, originally using YOLO for object recognition in the input movies. The YOLO outputs were further analysed using the Deep Sort model, which uses sliding windows and a 3D conversion model to identify face features and actions. Optimised bounding boxes with nonmax suppression. The datasets included UCF101, kinetics, Kth, and TRECVID 2008. The reference is Tsai et al. (2020). [8].

The authors Kumar et al. used a model name the Gaussian Mixture (GMM) in conjunction with Kalman of probabilistic model features for feature identification, rather than relying on the Kalman filter. The Gated Recurrent Neural Network (GRNN) had been subsequently utilized to distinguish and classify specific actions or labels. This study used the Kth, Ucf, and Ucf101 databases. The source cited is Kumar et al. (2021). [12].

Zhang et al. used the VGG-16 Convolutional Neural Network (CNN) structure to first analyse visual inputs. This splited the video frames into two groups. High-level convolutional features and mid-level convolutional features. In order to understand time based information, the sequential attributes were then fed into short-term memory models, namely FC-LSTM and Conv-LSTM. Subsequently, the output from the model that extracted the features was sent to an another module named attention, which is dedicated for detecting and highlighting important character-

istics. This research firstly one attention module was used namely the Temporal Attention Module (TAM) and another attention module helped it that they called the Joint Spatial-Temporal Attention Module or JSTAM. The output of Convolutional LSTM or Conv-LSTM was inputted into JSTAM, whereas the result of FC-LSTM was allocated to TAM. The attention modules, obtained from the analysis of the primary component, were incorporated into a fusion module that produced an output with labels defined by softmax. This experimentation used the Hmdb51 and Ucf101 datasets.[9].

The project done by Muhammad et al. began by using input frames from the dataset. These photages went through Convolutional Neural Network (CNN), then max-pooling layer, batch normalisation, and an activation function of ReLU. Further refinement of the outcome involved processing through residual dilated CNN blocks and the integration of skip connections. Within the residual block were feature learning components that incorporated D-CNN, batch normalisation, and ReLU activation. The skip connection consisted of two deep convolutional neural networks (D-CNNs) with filter sizes of 33 and 11, including batch normalisation and ReLU activation. The use of D-CNN resulted in improved performance via the process of semantic segmentation and the capturing of implicit information. The Fusion module collects the outputs from the remaining blocks and skip connections, and directs the combined output to an attention module to generate a context vector. The use of a two directional Long Short-Term Memory (Bi-LSTM) architecture was implemented to extract sequential patterns from the attention module. The patterns were subjected to further processing using completely linked layers and Softmax activation, which led to the production of actions [15].

In this study, Dai et al. used a basic architecture that included a stream attention-based Long Short-Term Memory (LSTM) model. Each video input from the dataset was processed. There are two streams: one focusing on spatial-temporal features and the other on temporal features. The temporal attention module autonomously identifies salient regions throughout the frames of the videos in the temporal system. After the pooling layers of the spatial-temporal system, an LSTM is used to reveal concealed temporal linkages within the deep spatial map. The spatial system assigns various weights to items at different levels. After that the combination of the spatial-temporal features and temporal feature occurs. Next a Fully connected layers were utilized to generate labels or outputs by aggregating the collective outcome. The research used the Ucf11, and Ucf sports as data sources. They also used the Jhmdb dataset. [4].

Khan et al. first acquire input videos from the dataset throughout their investigation. The main structure consists of 26 layers of Convolutional Neural Networks (CNNs), including conv layers, Max pool layers, activation layers. The inputted frames traverse all 26 levels of the model, with characteristics extracted in the FC. Subsequently, a method that combines characteristics using high entropy is used to identify and prioritise features. The PDaUM module effectively discovers the most robust traits by combining the Poisson distribution with Univariate Measures. The PDaUM algorithm prioritises the most prominent characteristic and then inputs it into the ELM to ascertain the ideal labels. The study conducted by Khan et al. in 2021 made use of datasets like Ucf sports, Kth, Weizzman, and Hmdb51. [11].

Luvizon and his colleagues use a multi-task framework in their research, aiming to accurately ascertain the placements of humans in static photographs and identify

their actions in video sequences. The architectural design incorporates Convolutional Neural Networks (CNNs), which consist of a downsampling unit, an upsampling unit, and blocks of prediction. The crucial element, the prediction blocks, extract pictures or features from the previous pyramid, producing multitask characteristics that are used for predicting both posture and movement. Nevertheless, the significant computational expense linked to this design presents a constraint, impeding its practicality in real-time applications. [6].

Mazzia and his colleagues did a research where they introduced an attention-focused architecture called Action Transformer for the purpose of live action identification. The The Transformer architecture efficiently processes sequences of pose matrices by using a linear projection map, which converts the poses into a model with a higher number of dimensions. The researchers created three separate sets of the model, with each set consisting of 2, 5, and 10 samples, respectively. The present technique fails to effectively use the understanding of how different components of posture are interrelated, despite its positive qualities [23].

Gao and colleagues provide a novel paradigm in their work, using insights from nature to distinguish human activities from different perspectives. The model simultaneously learns adjustable weights for each camera, integrating fusion at both the score and feature levels. In order to facilitate the acquisition of vocabulary at the category level, the model produces a collection of instances pertaining to certain action classes. Nevertheless, an intrinsic constraint exists in its vulnerability to variations in input characteristics and the efficacy of dictionary learning, which directly affects the accuracy of classification [2].

The study conducted by Chakraborty et al. suggests the use of transfer learning on a convolutional neural network (CNN) that had pre-trained weights to recognise human activities using a small number of static pictures. Essential elements consist of data augmentation to overcome the dataset's constraints and transfer learning based on convolutional neural networks (CNN). Although the model outperforms existing cutting-edge techniques, its performance deteriorates in same contextual environments [10].

The research conducted by Yuan et al. focuses on identifying group activities using the Dynamic Inference Network. The Dynamic Relation and Walk modules are used on a spatiotemporal graph to forecast a relation matrix and dynamic walk offsets. This method produces a personalised and dynamic graph that outperforms the most advanced techniques, even when using limited computational resources. Nevertheless, the article would be enhanced by taking into account contextual signals derived from the visual realm (Yuan et al., 2021).

The work conducted by authors, Perrett et al. presents an approach to action classification, which combines a few-shot methodology with a CrossTransformer module of attention. This idea takes into account the resemblance between frames in a video and frames in a set of supports. A convolutional network is used to compute the D-dimensional representation of the input frame. The examination of results demonstrates that data augmentation has a good impact on convolutional neural network designs. Further investigation might examine spatiotemporal variations of TRX that are consistent with individual frames, perhaps revealing marginal improvements when using tuples in the suggested paradigm (Perrett, 2021).

The research conducted by Ray et al. rigorously examines a wide array of publications published in the last 10 years, up to 2022, with the objective of understanding

human behaviour using computer vision techniques. The main objective is to classify activity identification into generative, discriminative, and graph-based methodologies. The study determines that the effectiveness of transfer learning in human activity recognition (HAR) models relies on both, firstly the efficiency of this model in computational performance. and the match between the train and test datasets. Nevertheless, the research indicates a negative link between this association with the number or variety of courses. [32].

Zhang et al. did a research on action recognition using a Graph Convolutional Network (GCN) that relies on skeletal data. The suggested system functions by analysing and processing skeletal data. The system employs a Self-Attention Temporal Dependency Graph Convolutional Network (SATD-GCN). With that we had a Self-Attention Pooling (SAP) module. This technique utilizes attention to it self to remove unimportant vertices in the network after selecting the relevant ones. The temporal graph module adapts the pace of joint movement depending on activities, discerning between slight and substantial changes [30].

The work conducted by Hu et al. use the 2s-AGCN approach to analyse the movement patterns of skeletal data at different scales, focusing on their spatiotemporal properties. The deep spatiotemporal extraction module, in conjunction with the scalers time specimen module, improves the extraction of spatial and temporal data, hence increasing the field of the feature mask. The research investigates the efficacy of three feature fusion techniques—Module for extracting spatial features, methods for selecting feature fusion, and module for extracting temporal features—in capturing hierarchical deep spatial-temporal information.[21].

The work conducted by Khan et al. introduces a hybrid deep learning model designed for the purpose of identifying human activities. The LSTM network uses CNN layers to extract spatial data and capture temporal information by combining Convolutional with Long Short-Term Memory (LSTM) models. Here the CNN-LSTM architecture surpasses other models, attaining the highest level of accuracy [22].

Extracting distinct information from combined joint skeletons is a significant obstacle in action recognition, particularly in the domain of skeleton-based approaches. The study conducted by Song et al. presents a model called the Graph Convolutional Network (GCN), which aims to enhance performance by reducing unnecessary parameters. The research emphasises the significance of enhancing the efficiency of skeleton-based recognition. [26].

The study conducted by Wu et al. explores the use of LSTM with CNN, and GCMP for crowd action detection. It presents the use of GCMP (Group Context Motion Patterns) for classifying basketball movements. This method successfully captures motion information by analysing patterns of two groups of players. The dataset of NCAA categorises actions in three separate phases: Occurrence of the event, subsequent to the event, and preceding the event, in accordance with the GCMP paradigm. This method is first used, then CNN-based feature extraction is performed, and finally, LSTM is utilised to predict events [3].

The work conducted by Pang et al. suggests the use of skeletal data to identify human interactions. The research seeks to use graph topologies to represent the relationships between various anatomical elements. This is accomplished by using the Interaction Graph Transformer network. The IGFormer system integrates a Graph-based Inertial-Motion Sequence Alignment (GI-MSA) module, which repre-

sents human movement using a graph structure. The research introduces a Semantic Partition module. This module utilizes spatially and temporally extracted data. It then converts each human skeleton into a sequence of Body Part Time. The objective is to improve graph learning [25].

Plizzari et al. provide a research that demonstrates the use of the self-attention in the Transformer, specifically in (ST-TR) models. The objective is to efficiently collect and analyse mutual interdependencies. The study introduces the ST-TR method, which incorporates with TSA to understand the relationships between images and examine the spatial temporal linkages among human limbs inside a single image. In addition, it includes the SSA module for spatial temporal self-attention. The citation for this work may be found in [16].

Basak et al. tackle an issue in recognising human activity by using D-swarm net, which integrates four parameters: distance, velocity, angle, and angular velocity. The pairings are arranged on a stack and combined using a modified inception Resnet. The consolidated data is then sent to Automated Learning and Optimisation (ALO) for the final process of selecting relevant features [17].

Duan et al. explore the basics of Human Activity Recognition in relation to the skeletal system, using Graph Convolutional Networks (GCN). The authors provide PoseConvo3D, an innovative approach that employs a 3D CNN to address the constraints of GCN. The posture estimator produces a three-dimensional heatmap, which is then analysed using an approach that uses a layered configuration of heatmaps over five layers to achieve accurate pose analysis [19].

The work conducted by Yadav et al. focuses on a system that extracts geometrical and kinematic properties and combines them with the original skeleton coordinates. Significantly, the system places user privacy as a top priority by abstaining from using authentic images and instead depending only on coordinates [29].

In this work, Xiao et al. provide the temporal gradient as a novel modality to enhance the contingency of RGB feature extraction. This notion is considered innovative. Conventional methods such as fix match or mix match typically fail to consider the time-dependent changes and many modes of data, which may lead to less than ideal results in extracting features. Replacing RGB input frames with temporal gradient leads to dramatically improved outcomes, with a notable 25% rise in accuracy as reported in their study [28].

The research conducted by Muhamad et al. suggests a method to improve the efficacy of feature extraction in GRU, LSTM, and RNN layers using an augmentation technique. The proposal entails including four LSTM layers and two GRU layers into the approach, hence enhancing its capacity to detect and categorise behaviours seen in video streams. The sophisticated LSTM structure consists of four layers and utilises the input gate to regulate data transfer, the Forget gate to preserve data, and the output gate to determine the value of memory. GRU, which is equipped with two gates, employs the update gate to regulate the pace of information processing and the reset gate to govern the retention or disregard of information [24].

Guo et al. have introduced comprehensive methods to enhance a system's feature-learning efficiency. To improve the logical part of the learning the introduced a brand new model named AimCLR. The input undergoes three augmentations, with the most extreme augmentation being processed by an EADM system and stored in a memory bank. This architecture generates a new pattern of movement which again enhances the application of such features [20].

The research by Chen et al., tries to utilize all the possible information hidden within the frames to recognize human activities and classify them. The MMVIT adheres to VIT's fundamental concepts while expanding its scope to include a space-time modality within a four-dimensional volume. The model possesses the capability to leverage various visual modalities. These modalities include motion vectors, I-frames, residuals, and more. They contribute to the model's comprehensive understanding of visual information. This method provides a comprehensive video demonstration of the dissection process, isolating specific elements for further analysis [18].

Sun et al. examined the use of several individual data modalities in algorithms for Human Activity Recognition (HAR). The applied methodologies in the study were diverse. They included a Two-stream 2D Convolutional Neural Network (CNN), which focused on spatial and temporal information. Additionally, a Recurrent Neural Network (RNN) was employed to capture sequential patterns, and a 3D CNN provided insights into three-dimensional structures. Furthermore, Transformer-based methods were utilized for their attention mechanism and capacity to handle long-range dependencies in the data. The basic models included 2D CNNs, which were augmented with LSTM networks to capture sequential dependencies. Additionally, Graph Neural Networks (GNN) were incorporated into the models. Nevertheless, a significant constraint is the need for a heterogeneous dataset; supplementary data, comprising varied methods of gathering, is essential. Volunteers participate in activities that highlight the need of collecting data from several modes to build comprehensive standards for increasing the detection of human actions across multiple modes [27].

Muhammad and his colleagues used a Convolutional Neural Network (CNN) structure to extract unique characteristics from a variety of video material. The UFLBs and a dilated CNN with skip connection inclusion were used to further improve these capabilities. The inclusion of an attention layer significantly enhanced the performance of the LSTM model in every step. The features were then inputted into the BiLSTM network to capture temporal information. The main benefit of the article is its sophisticated motion features and convolutional technique, which allows for a smooth transition from 2D to 3D, resulting in highly accurate and dependable information. [14].

In this investigation, Singh et al. used the Temporal Shift Module (TSM) in addition with ResNet-18. Their achievement includes creating a two path temporal contrastive model (TCL) that use unlabeled videos at different speeds also maintaining a consistent representation of activity regardless of the speed. The ultimate goal was to introduce a solution to the problem of semi supervised action learning from the videos. To solve this they utilized a huge dataset of unlabeled data and a small labeled data.

The study conducted by Shiranthika et al. introduced the use of CNN (Convolutional Neural Networks) and LSTM (Long Short-Term Memory). One of the most note worthy task was to include Conv-2D layers with in the CNN model. They also optimized the who method to increase its speed and improve the accuracy. Here the LSTM solved the vanishing gradient problem of RNN. [7].

In a separate research endeavor, Ahmad et al. explored an innovative approach. This involved the integration of a multi-head convolutional neural network (CNN), de-

signed to extract diverse features. Additionally, a long short-term memory (LSTM) component was incorporated, enhancing the model's ability to capture and analyze sequential patterns in the data. This architecture worked best for identifying human activities video photages. Although many machine learning model uses techniques such as support vector machines (SVM) and K-nearest neighbours (KNN) to achieve better accuracy, but the use of such models requires the collection of data from wearable sensors. Here the multi-head Convolutional Neural Network (CNN) model is made of three CNNs extracted information better from data obtained from different sources. Further more, All three convolutional neural networks (CNNs) goes through compression, to increase its speed and usability in weaker devices. This novel approach generates better accuracy when compared to a single CNN model [1].

Chapter 3

Data Collection and Analysis

3.1 Data collection

Dataset created by the authors Borges, J., Queirós, S., Oliveira, B. et al. (2021) [29] had been utilized in this research for classifying 20 different violent and non violent actions. There are many video frames of two persons in this dataset and those persons are doing some action. So, this dataset mostly contains frames from videos that show different actions between two people in the video.[29] dataset is quite large and it is 25GB. Each video in the dataset, obtained from reference [29], lasts approximately 18 to 19 seconds. The dataset comprises a total of 668,992 frames, and each frame has dimensions of width 2048 and height 1536 pixels.



Figure 3.1: Sample Video Frames

3.2 Data Preprocessing

Our research employed YOLOv8 segmentation model developed by Ultralytics for the purpose of data preprocessing [31] At first, we utilized the YOLOv8 segmentation technique on the video frames to locate persons and segment their territories within the frames. Afterwards, we selected and kept only the segmented areas that corresponded to individuals, disregarding any other irrelevant information from the frames. The fragmented sections of individuals were kept as arrays including pixel



Figure 3.2: The phases of preprocessing

values. Subsequently, the procedure was applied on every frame in the video collection, guaranteeing the preservation of solely the segmented individuals while eliminating other frame particulars. Figure 3.4 exhibits a partitioned frame from which superfluous information has been eliminated. Figure 3.2 offers a complete overview of every stage of the preprocessing operation.

Figure 3.4 exhibits a partitioned frame from which superfluous information has been eliminated. Figure 3.2 offers a complete overview of every stage of the preprocessing operation.

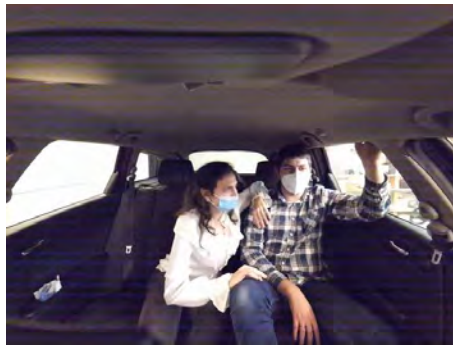


Figure 3.3: A frame from one video

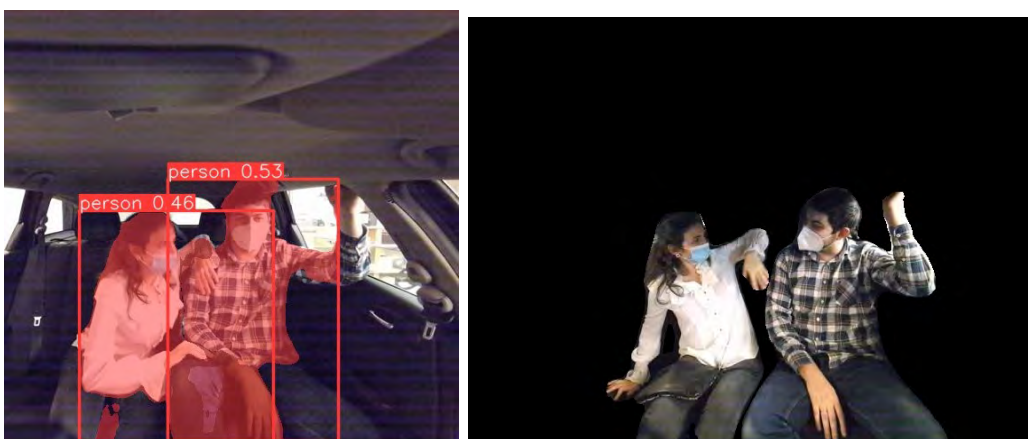


Figure 3.4: Human Segmentation and Background Removal

3.3 Data Analysis

This dataset is classified into twenty different levels, with each class representing a single or a series of actions. The dataset consists of 16 individuals, with 9 classified as male and 7 as female. Table 3.1 presents a comprehensive summary of the 20 classes, with each class encompassing many activities. Notably, classes 1 to 12 are associated with violent behaviors, while classes 13 to 20 involve non-violent actions 3.1.

Our analysis suggests a total of 1,275 films and 668,992 frames in the dataset [3.8]. It's remarkable that each class comprises approximately 60 movies, and within the dataset, two individuals are observed executing various tasks in the videos. In each class, all 16 individuals participate, shifting their positions. For instance, in class 1, an action like pushing and punching involves person 1 pushing and punching person 2, and the same action is witnessed with exchanged roles in other class 1 movies, leading in combinatorial role combinations. This results in an average of 62 videos every class due to the varied combinations of the 16 participants in two positions. Additionally, in section 3.8, we find that violent classes (class 1 to class 12) have around 35,000 frames each class, while non-violent classes (class 13 to class 20) average around 28,000 frames per class [3.8]. Figure 3.8 further indicates that each video in violent classrooms comprises roughly 575 frames, while non-violent classes have frame count of 450 for each video.

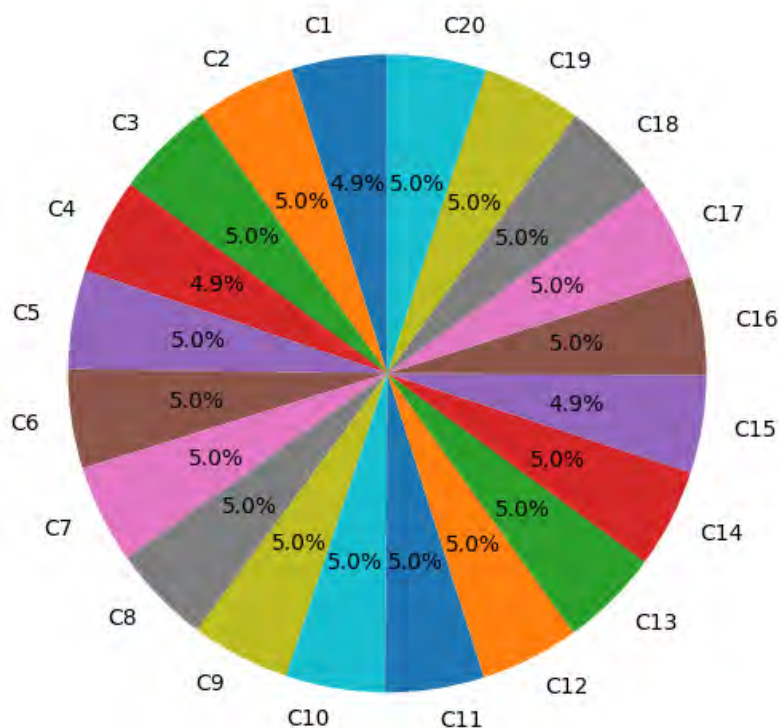


Figure 3.5: Videos per classe

From the given pie-graph in Figure 3.5, it indicates that each class constitutes same percentage of all of the classes. Our finding from the dataset reveals violent courses (class 1 to class 12) exhibit more activities compared to non-violent classes (class

13 to class 20). Specifically, practically every violent class has 4 actions, with class 2 having the highest at 5 actions. In contrast, non-violent classes (class 13 to class 20) normally have 3 actions, except for classes 19 and 20 which have 2 actions, and class 16 which has 4 acts [3.7]. Furthermore, the dataset encompasses sixty percentage of violent or aggressive activities and forty percentage of non-violent or casual activities, as represented in Figure 3.6.

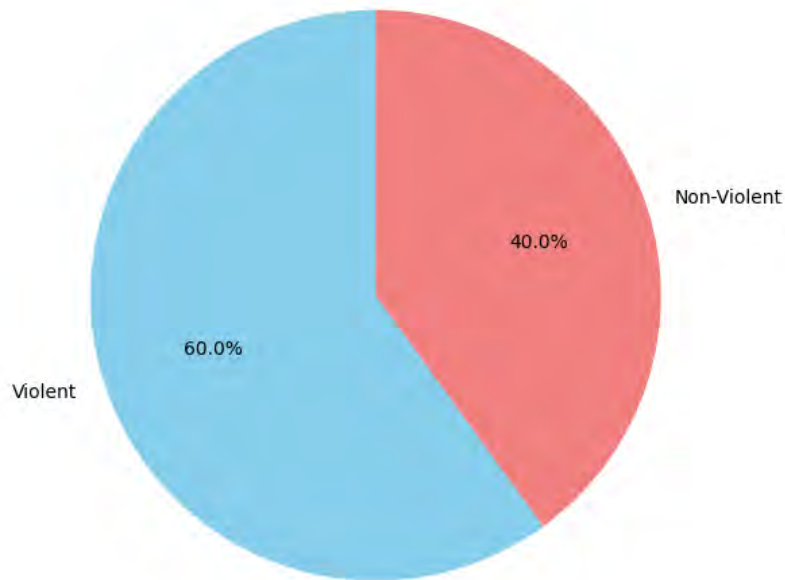


Figure 3.6: Non-violent and Violent classes

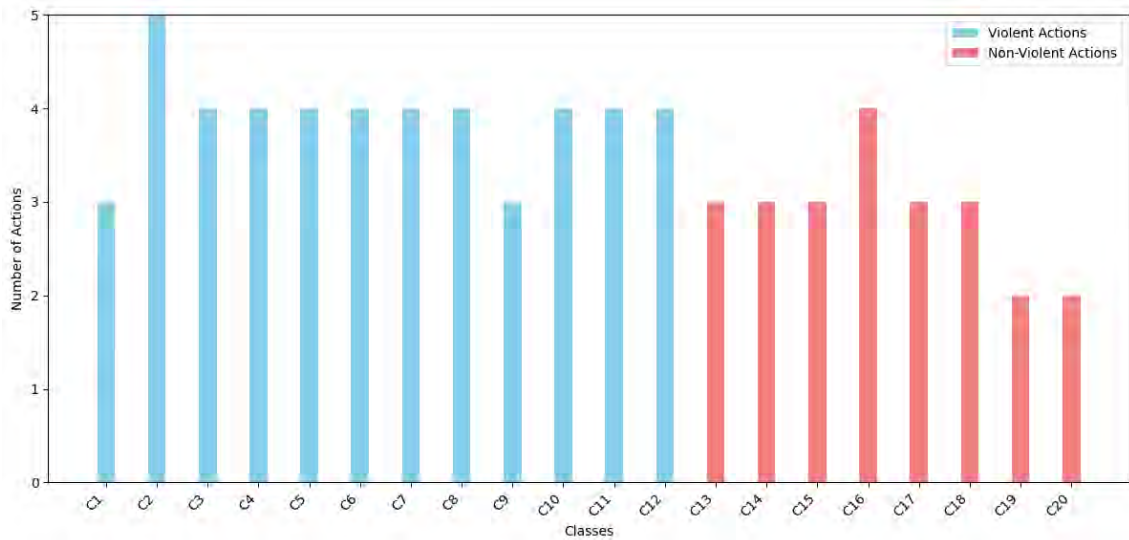


Figure 3.7: Multiple actions in each classes



Figure 3.8: Classes and Frames

Table 3.1: Classes and Actions

Class 1	Exchange of words (P1, P2) – Pushing (P1) – Striking (P1).
Class 2	Dance/song (P1) – Asking for a kiss (P1) – Rejecting a kiss (P2) – Slapping (P1) – Pulling/Pushing (P1).
Class 3	Arguing (P1, P2) – showing middle finger(P2) – Employing physical force (P1) through kicking – Applying force to the neck (P1).
Class 4	Discussion (P1, P2) – Intimidating with the possibility of striking (P2) – Slapping (P1) – Inciting (P1).
Class 5	Sexual Harassment: Approaching or intruding upon the other person (P1) – Caressing the hair (P1) – Physical contact with the body (P1) - Attacking with a backpack or purse (P2) .
Class 6	Greeting/Appreciating (P1, P2) – Showing something on a cellphone (P1) – Threatening with scissors (P1) – Requesting/Stealing a wallet (P1).
Class 7	Arguing (P1, P2) – Drawing a gun from a purse/clothing/backpack (P1) – Pointing a gun (P1) – Using a gun to hit or assault (P1).
Class 8	Arguing (P1, P2) – Taking out a knife from purse/clothing/backpack (P1) – Pointing the knife (P1) – Stabbing (P1).
Class 9	Approaching the other person (P1) – Threatening with a knife (P1) – Physical contact (P1).
Class 10	Engaging with phone (P1) – Engaging with phone (P2) – Slapping (P1) – Grabbing and striking (P2).
Class 11	Relaxing (P2) – Drinking (P1) – Throwing a bottle (P1) – Pushing (P2).
Class 12	Using phone (P2) – Checking or looking at the cellphone (P1) – Moving away (P2) – Pulling/Shoving (P1).
Class 13	Talking/engaging in conversation (P1, P2) - P2 begins to cry - Embracing (P1, P2).
Class 14	P1 requests P2 to take photos - P2 captures images - P2 shows P1 the resulting pictures.
Class 15	Applying lipstick - Styling hair (P1) - P2 taking a break.
Class 16	Sneezing once or more (P1) - Getting a tissue and using it to wipe the nose (P2) - Reading a book (P1).
Class 17	Yawning (P1) - Turning neck (P2) - Putting on headphones, enjoying music, singing/dancing (P2).
Class 18	Having a drink and meal (P1) - Taking photos with a smartphone (P2).
Class 19	Having a conversation or chatting (P1) - Coughing while using a laptop (P2).
Class 20	Taking notes on the notepad (P1), Applying hand sanitizer (P2).

Chapter 4

Model Architecture

4.1 Methodology

We utilized three particular models to classify our dataset of violent and non violent video dataset. Multiple human action detection was achieved by utilising the YOLO-v8, CNN, and LSTM model architectures. For our work, we utilised the INCAR Dataset, which consists of 1275 videos including 20 classes of videos. The classes covers different kind of violent and casual activities. We initially inputted the videos which transformed into different frames. After the conversion of video into individual picture frames, the study worked with various model architectures for educational purposes.

4.2 You Only Look Once v-8

YOLO v-8 is an fast and well performed computer vision model designed to efficiently detect and recognise objects in real-time. It's a lightweight and user-friendly programme, perfect for easily and efficiently identifying objects. We incorporated YOLO-v8 into our system after converting the video Dataset into individual frames. This model architecture performed the segmentation. We utilised the YOLO-v8 model's architecture on the frames. This model was used during the preprocessing part to segment the human with a padding of 30pixel near the palm area to cover any violent object hold by the persons. In total we had over 600k frames to go over. Our primary objective was to use this model architecture to identify humans and oder necessary objects and remove the background from frames to include important things to teach. Segmentation was performed based on the coordinate points of the human body, resulting in the removal of all other information except for the humans.

4.3 Convolutional Long Short-Term Memory

The Convolutional Long Short-Term Memory (ConvLSTM) is a unique neural network structure. It seamlessly integrates the benefits of Convolutional Neural Networks (CNNs) with the strengths of Long Short-Term Memory (LSTM) networks. This combination allows the model to effectively process spatial information using CNNs while retaining the capability to capture and learn temporal dependencies

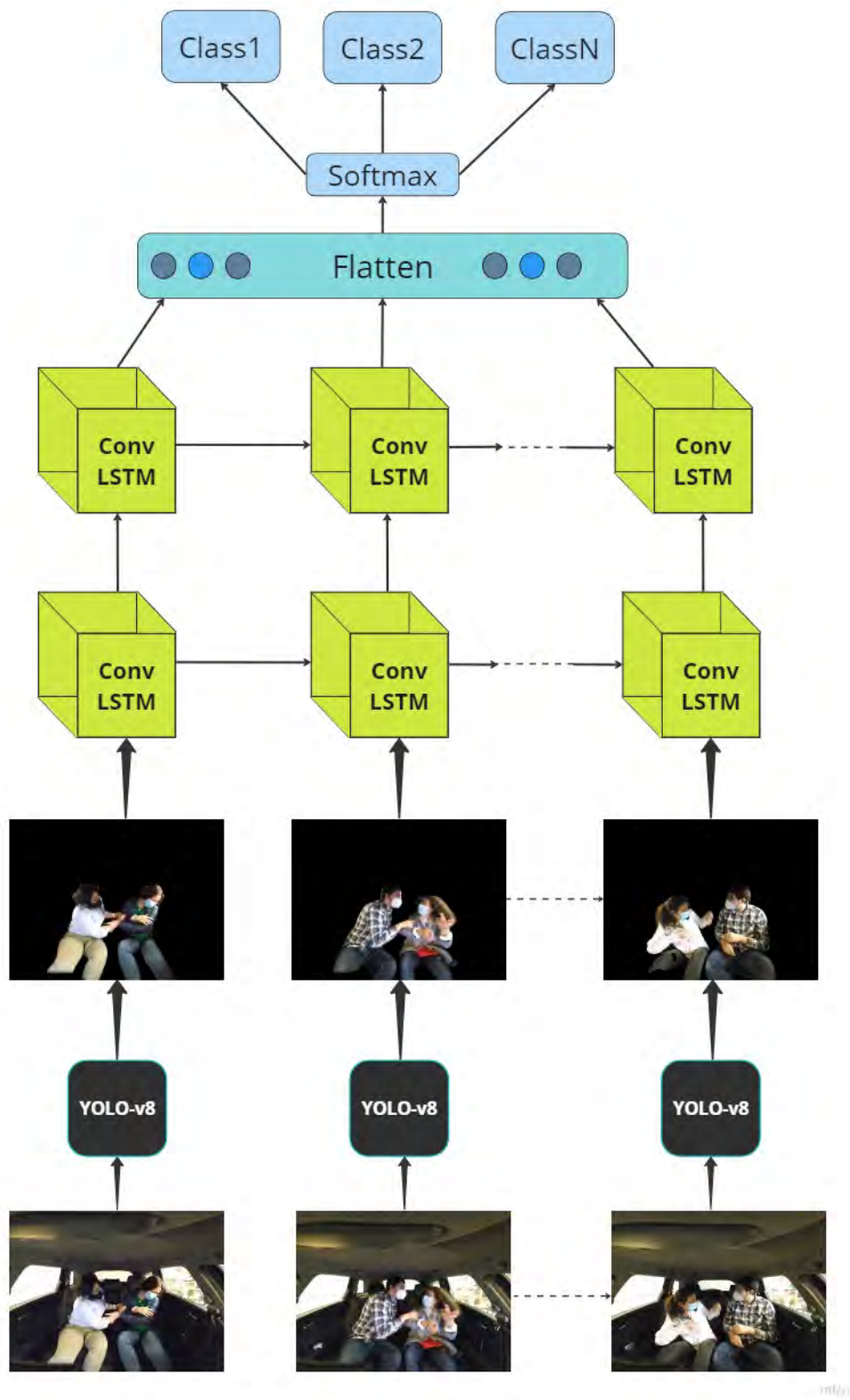


Figure 4.1: Pipeline with ConvLSTM

through LSTM architecture. Due to its capacity to effectively handle sequential or time-series data, LSTM is particularly well-suited for tasks involving videos, where pictures are linked together over time. ConvLSTM enhances the applicability of LSTM in such scenarios.

ConvLSTM is a very efficient model which mainly capture both spatial connections (object positions) along with temporal interactions (changes over time). ConvLSTM mainly combines the strengths of convolutional layers which excel at analysing spatial characteristics in data which are- pictures, with LSTM layers and these are adept at processing sequential information. This unique mix enables it to analyse sequences of data and acquire knowledge about the structure and characteristics of that data using convolutional techniques. ConvLSTM has shown remarkable use across several domains. For example, it functions as a flexible instrument in video analysis, aiding in the identification of activities and the monitoring of objects in films. I

As previously discussed, ConvLSTM is a concept that has been shown to be very successful in achieving its intended purpose. In our study, we used ConvLSTM, a blend of CNN and LSTM, to successfully identify films. Following the conversion of the films into individual frames, we used YOLO-v8 to only extract the human subjects and eliminate the backdrops. By decomposing the movie into individual frames, we are considering all the films as a collection of pictures. We are using the CNN component to extract the characteristics from the photos that have been manually selected by humans. Given that all the photos are interconnected in a consecutive arrangement, we use the LSTM model. When these two models are integrated and collaborate to address both geographical and temporal issues, they become very successful for achieving the objectives of our research. The CNN layer identifies and separates the patterns, edges, forms, and colours present in each of the input frames. It provides comprehensive spatial information that aids in understanding the events occurring inside a certain frame. Subsequently, we extract the visual characteristics from it. Subsequently, we input it into an LSTM model to get the precise prediction, which is accountable for capturing the temporal information. It measures the temporal variation of characteristics. Sequential processing determines the sequence of frames in a video and analyses the motion, movement, direction, tempo, and transitions within it. This is aiding in the prediction of actions and behaviours. The gates function as a unified entity that determines which information to retain and which to delete based on the previous frames. This feature enables the system to effectively preserve the contextual information extracted from videos.

As shown in Fig.4.4 the In this instance, ConvLSTM has four 2-dimensional convolutional LSTM layers. Following each layer, there is a three-dimensional max pooling layer. Subsequently, we had a time-distributed dropout. The model consists of fourteen differnt layers, with twelve of them being hidden levels. These hidden layers include a total of 193020 trainable parameters. The input layer transmits twenty frames each video, each having a resolution of 100×100 pixels and including 3 channels for the RGB colours. The first ConvLSTM2D layer consists of four filters, with a kernel size of three by three. The activation function used for this layer is the hyperbolic tangent (tanh). The recurrent dropout rate is set to 0.2, and this layer produces sequences of outputs. The input tensor with dimensions $20 \times 100 \times 100 \times 3$ is transformed to an output tensor with dimensions $20 \times 98 \times 98 \times 4$ by this layer. In the subsequent layer, we included a max pooling layer with a 3-dimensional size

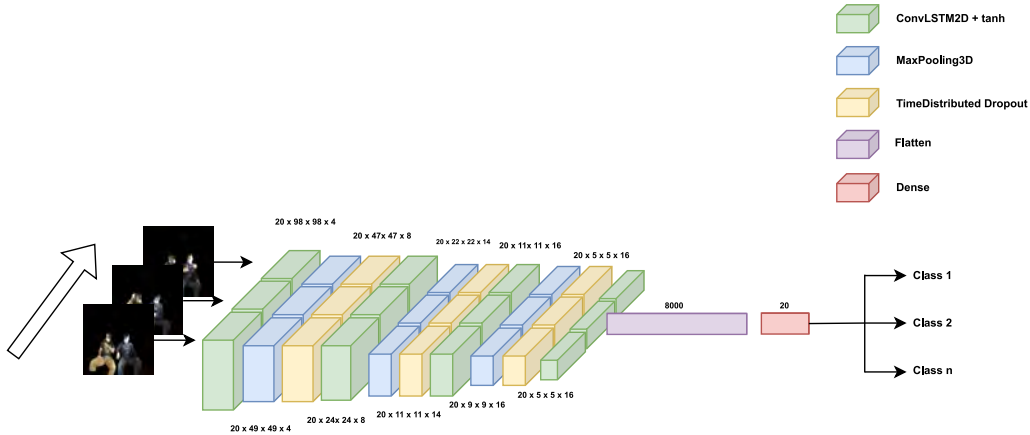


Figure 4.2: Model Architecture ConvLSTM

of one by two by two since we are working with time series data that includes an additional dimension. The max pooling operation decreases the dimensionality to $20 \times 49 \times 49 \times 4$. In the subsequent layer, a time distributed dropout with a dropout rate of 0.2 is used.

Next, we have an additional ConvLSTM2D layer with 8 filters, resulting in an output of $20 \times 47 \times 47 \times 8$. All the remaining hyperparameters were identical. Next, apply another 3D max pooling operation to further decrease the dimensions to $20 \times 24 \times 24 \times 8$. Additionally, apply temporally distributed dropout with the same dropout rate. The subsequent ConvLSTM2D layer consists of 14 filters of identical dimensions, which is then followed by 3D Max pooling and time distributed dropout. This finally results in a reduction to a size of $20 \times 9 \times 9 \times 16$. The subsequent layer consisted of 16 filters, along with the same max pooling and time distributed dropout, resulting in a dimension of $20 \times 5 \times 5 \times 16$. In the subsequent layer, we compressed all the elements and obtained a vector with a dimensionality of 8000. The final layer consists of a thick layer with a softmax activation function that maps to 20 distinct classes.

4.4 LRCN

The Long Short-Term Memory with Recurrent Convolutional Networks (LRCN) model utilises a ConvLSTM, which substitutes the fully linked layers of LSTM with a convolutional layer. This model incorporates both a CNN and an LSTM. CNN gathers characteristics from each frame and converts them into data points. These data points are then sent to the subsequent LSTM layer, which considers them as a temporal model. This model is more intricate than the ConvLSTM since it often has a greater number of parameters due to its use of two separate models. In our situation, the input consists of sequential data, namely frames. After being processed by the YOLO v8 model, the frames are then sent to the LRCN model. Subsequently, each frame undergoes convolutional neural network (CNN) processing, resulting in the extraction of all the characteristics. Then these characteristics are sent to the LSTM model. Which is responsible for recording the temporal order of the frames. The model network consists of four time-distributed convolutional layers, each followed by 2D max pooling and dropout. Following this process, the data under-

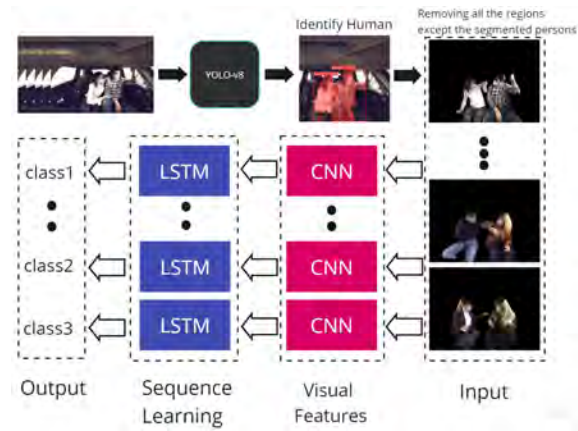


Figure 4.3: Pipeline of LRCN

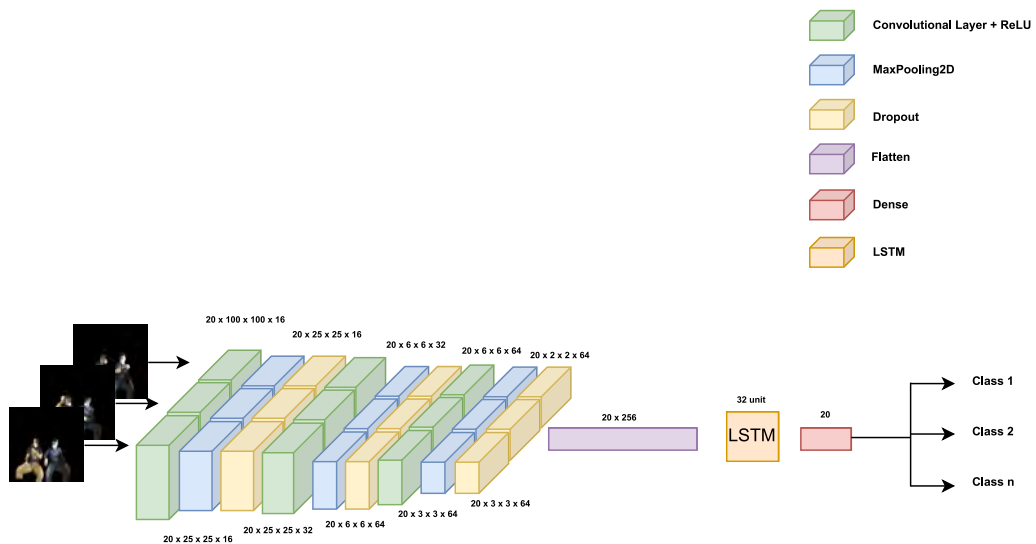


Figure 4.4: Model Architecture of LRCN

goes flattening, moving from the Convolutional Neural Network (CNN) to the Long Short-Term Memory (LSTM) layer, and then to a dense layer. The input layer maintains a stable configuration, providing twenty frames each video with a resolution of one hundred by one hundred pixels in three channels for RGB colours.

The input is transformed into dimensions of $20 \times 100 \times 100 \times 16$ in the time-distributed 2D convolutional layer. Every layer is equipped with sixteen filters, each having dimensions of three by three, and employs the Rectified Linear Unit (ReLU) activation function. The following time-distributed max pooling layer, with dimensions of 4×4 , produces an output of $20 \times 25 \times 25 \times 16$. Subsequently, a dropout layer is implemented with a dropout rate of 0.25.

The subsequent convolutional layer, consisting of thirty-two filters, transforms the input into dimensions of twenty by six by six by thirty-two. After applying another round of max pooling and dropout, the tensor is compressed to a size of $20 \times 3 \times 3 \times 32$. By applying another convolutional layer with sixty-four filters, which replicates the previous parameters, and then adding a final max pooling layer, the outcome is a tensor with dimensions twenty by two by two by sixty-four. Following an additional convolutional layer, the data is reshaped into a matrix of dimensions twenty by two hundred fifty-six and then proceeds through the LSTM layer, which consists of thirty-two units. Ultimately, it passes through a compact layer that utilises a softmax activation function, which then maps the output to the corresponding classes.

Chapter 5

Result Analysis

5.1 ConvLSTM

We took three different approaches to evaluate the model. First, we looked at all the classes together and found an accuracy of 59%. Then, since a few classes had issues, we excluded three of them and worked with the remaining 17, resulting in an accuracy of 66%.

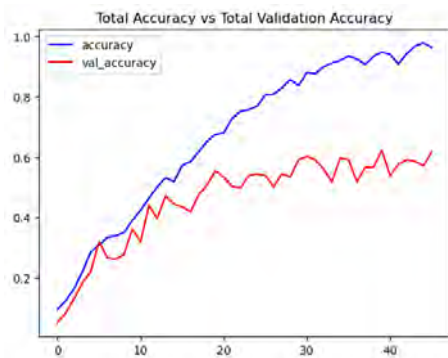


Figure 5.1: Training accuracy vs Validation accuracy (ConvLSTM)

Finally, we delved into binary classification based on the dataset's violent and non-violent actions, achieving an accuracy of 72%. Throughout these evaluations, we split the dataset into 75% for training and 25% for testing. Moreover, within the training data, we further divided it into 80% for training and 20% for validation. As illustrated by Fig. 5.2 in the confusion matrix with the size of 20X20 including all classes, class 7 has performed extremely well with the prediction rate of 92% over true class. We can also see that class 19 has also performed well with the prediction rate of 89% over the true class. But there is a major point to notice that few of classes over all classes performed worstly. In the matrix we can see that, class 3 has lowest prediction rate with 0% and this class is misclassifying with class 1. Since both these classes have same actions which are- debating, hitting, pushing, trying to hit etc these actions are common for this reason class 3 is misclassifying with class 1. Moreover , class 3 has major data corruption for that reason class 3 has very bad prediction rate which is 0%. In the matrix, we can also notice that class 2 has also very worst prediction rate which are 5% . This has occurred because class 2 carries some actions such as- conversation,shoving,hitting staring at each other etc these actions are common with class 1 for that reason class 2 is misclassifying with class

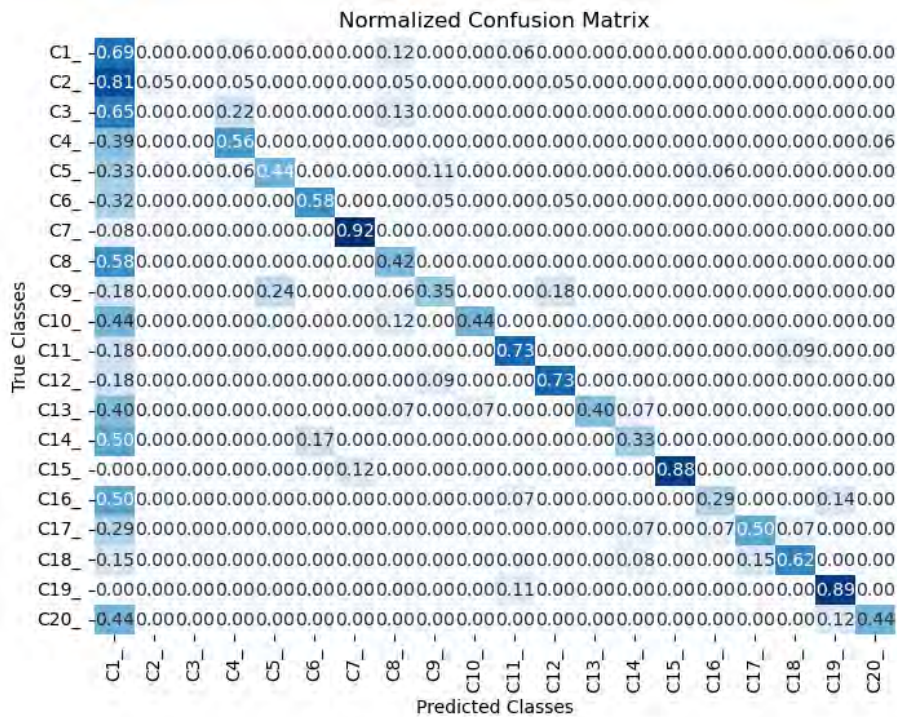


Figure 5.2: Confusion Matrix (ConvLSTM) for 20 classes

1. In addition, if we see the matrix all the classes slightly misclassifying with class 1 and the main reason is that class 1 has a common action which is conversation or debate and this action is common in all of the class. For that reason, the classes are misclassifying with class 1 by using this model.

From the classification report shown in Fig. 5.3 we can see that class 7 and class 19 has performed better comparing to other classes. We can see that class 7 obtained precision of 86%, recall of 92 % and f1-score of 89% whereas class 19 obtained precision of 76%, recall of 89 % and f1-score of 82% On the other hand we can see class 2 and class 3 obtained very worst scores. As we can see, class 3 obtained 0 % in precision, recall and f1-score and class 2 obaitend precision of 100%, recall of 5% and f1-score of 9%. So, we can conclude that these two classes, class 2 and 3 performed worst comparing to the other classes. In this report the accuracy is 59%, the macro average precision is 68%,macro average recall is 51% and macro average f1-score is 54%.

After finding major problems with classes 1, 2, and 3, we decided to remove these three classes and see how the model performs. If we look at the confusion matrix now, we can see that before, when all classes were included, all of them were getting misclassifying with class 1. But now, after taking out these three classes, the diagonal line in the matrix as illustrated by Fig. 5.4 looks better compared to the previous confusion matrix that had all classes. Even though a few classes are still getting mixed up with class 4, the model (conv-lstm) is doing better after getting rid of these three classes.

From the classification report shown in Fig. 5.5 we can see at the classification report with 17 class of conv-lstm, here class 15 and class 19 has performed better comparing to other classes. We can see that class 15 has 93% of precision, recall and f1-score and class 19 has 88% of precision, recall and f1-score . Moreover, we

20 Class Conv-Istm				
Class Name	Precision	Recall	F1-Score	Support
class1	0.09	0.69	0.16	16
class2	1	0.05	0.09	21
class3	0	0	0	23
class4	0.56	0.56	0.56	18
class5	0.67	0.44	0.53	18
class6	0.79	0.58	0.67	19
class7	0.86	0.92	0.89	13
class8	0.33	0.42	0.37	12
class9	0.6	0.35	0.44	17
class10	0.88	0.44	0.58	16
class11	0.67	0.73	0.7	11
class12	0.62	0.73	0.67	11
class13	1	0.4	0.57	15
class14	0.67	0.33	0.44	18
class15	1	0.88	0.93	16
class16	0.67	0.29	0.4	14
class17	0.78	0.5	0.61	14
class18	0.8	0.62	0.7	13
class19	0.76	0.89	0.82	18
class20	0.88	0.44	0.58	16
accuracy			0.59	319
macro avg	0.68	0.51	0.54	319
weighted avg	0.67	0.49	0.51	319

Figure 5.3: Classification Report (ConvLSTM) for 20 classes

can see that some classes got comparatively less score which are class 5 and class 17. We can notice that class 5 has less recall with 33% and class 17 has 39% recall which is comparatively lesser than other classes. In this report the accuracy is 66%, the macro average precision is 67%, macro average recall is 67% and macro average f1-score is 65%.

So, we can conclude that after removing the 3 classes, the model (conv-lstm) performs better than before.

Further more, our dataset is again sub grouped into two types of data: violent and non-violent. The first twelve classes are considered violent, and the remaining eight are labeled as non-violent. That's how we've organized our dataset into two categories. We can check out the confusion matrix below for the binary classification results.

As illustrated by Fig. 5.6 we can see the confusion matrix, the violent class has performed better with the prediction rate of 85% over true class where non-violent class has the prediction rate of 63% over true class.

From the classification report shown in Fig. 5.7 we can see that both violent and non violent class has performed moderately well where violent class has precision of 61%, recall of 85% and f1-score of 71% whereas non-violent class has precision of 86%, recall of 63% and f1-score of 73%. In this report the accuracy is 72%, the macro

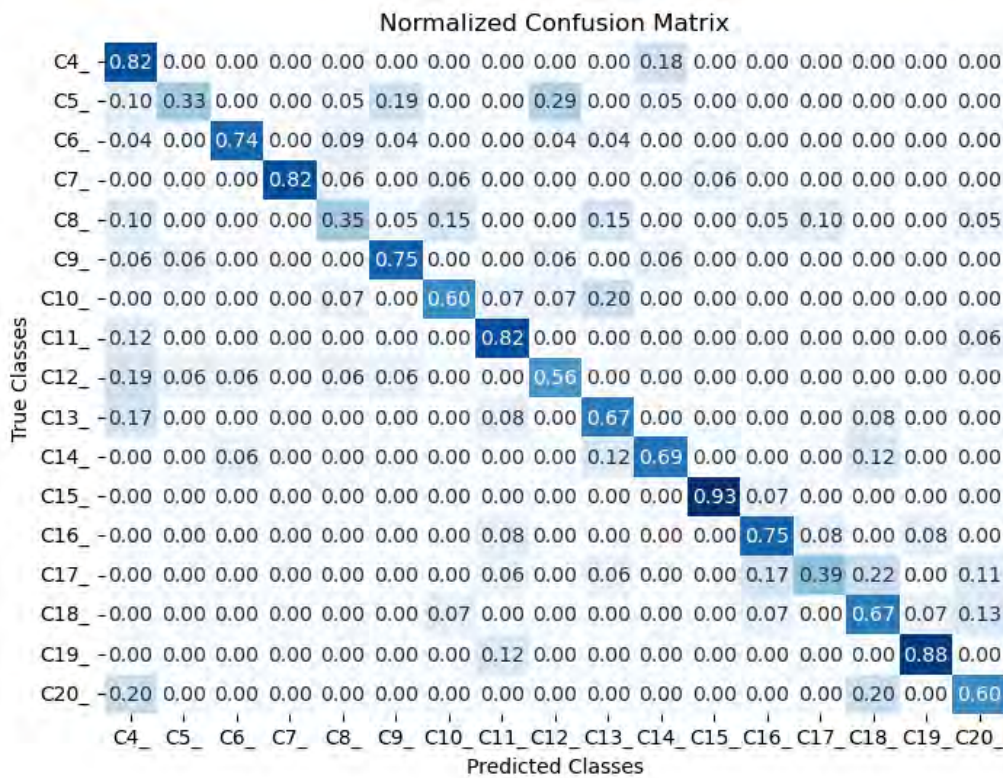


Figure 5.4: Confusion Matrix (ConvLSTM) for 17 classes

average precision is 73%, macro average recall is 74% and macro average f1-score is 72%.

5.2 LRCN

Using the model, we have worked in three ways. Firstly, we took all the classes and evaluated the accuracy of 62%. Secondly, as few classes had some issues, we discarded 3 classes and worked with 17 classes and after applying the model we obtained an evaluated accuracy of 63%. Lastly, we have performed the binary classification as the dataset has violent and non-violent action and here we have obtained the accuracy of 88%. For all of these, the dataset was split into training (seventy-five percent) and testing (twenty-five percent). Later, the training data was further divided into training (eighty percent) and validation (twenty percent). As depicted in Figure 5.8, a total of fourteen epochs were used during the model training for twenty classes.

The confusion matrix is of size twenty by twenty, representing the twenty different classes. Within these twenty classes, classes one to twelve are categorized as violent, while classes thirteen to twenty are non-violent. The rows of the matrix correspond to the true classes, and the columns represent the predicted classes. In this context, the diagonal entries signify genuine positives or classes predicted accurately.

It is evident that class sixteen performed exceptionally well compared to the other classes. According to the confusion matrix, class sixteen achieved a prediction rate of ninety-three percent, accurately predicting instances of the true class. Class sixteen includes activities such as sneezing once or more, retrieving a tissue, using it to

17 Class Conv-Istm				
Class Name	Precision	Recall	F1-Score	Support
class4	0.38	0.82	0.51	11
class5	0.78	0.33	0.47	21
class6	0.89	0.74	0.81	23
class7	1	0.82	0.9	17
class8	0.54	0.35	0.42	20
class9	0.63	0.75	0.69	16
class10	0.64	0.6	0.62	15
class11	0.7	0.82	0.76	17
class12	0.5	0.56	0.53	16
class13	0.44	0.67	0.53	12
class14	0.73	0.69	0.71	16
class15	0.93	0.93	0.93	15
class16	0.6	0.75	0.67	12
class17	0.7	0.39	0.5	18
class18	0.53	0.67	0.59	15
class19	0.88	0.88	0.88	17
class20	0.5	0.6	0.55	10
accuracy				
accuracy			0.66	271
macro avg	0.67	0.67	0.65	271
weighted avg	0.69	0.66	0.66	271

Figure 5.5: Classification Report (ConvLSTM) for 17 classes

wipe the nose, and reading a book, among others. These actions are unique in class 16 compared to other classes's actions. So due to that class 16 has extremely well prediction rate over other classes. . Secondly, class 7 also has a good prediction rate which is 92 % over the true class. Since class 7 includes debating ,retrieving a firearm from a purse/clothing/backpack, aiming a firearm,using a firearm to strike or attack etc. which made this distinct and unique from the other classes.

On the other hand, shockingly class 3 has the worst prediction rate which is 0%. The class 3 contains debating , Middle finger showing , kicking , choking etc actions. The class 3 is mostly predicting class 1, since class 3 and class 1 both class have almost same actions. Though class 3 has one unique action which is- middle finger showing but this specific action has less number of frames as a result, for less number of frames, unfortunately the model is not learning that particular action. In addition, class 3 is also misclassifying with class 13 as class 13 also has the same action which is debating or conversation. So, for these reasons class 3 is misclassifying with class 1 and class 13 and has the worst prediction rate of 0%. Also we can notice that, class 2, class 11 and class 14 has approximately less prediction rate which is 33%, 27% and 33% and these classes are misclassifying with class 1 mostly. According to the Confusio matrix of LRCN, we noticed that most of the classes are misclassifying with class 1 as class 1 has common actions- conversation , pushing , pulling etc these actions are almost common in every classes. For this reason, most of the classes are misclassifying with class 1 mostly.

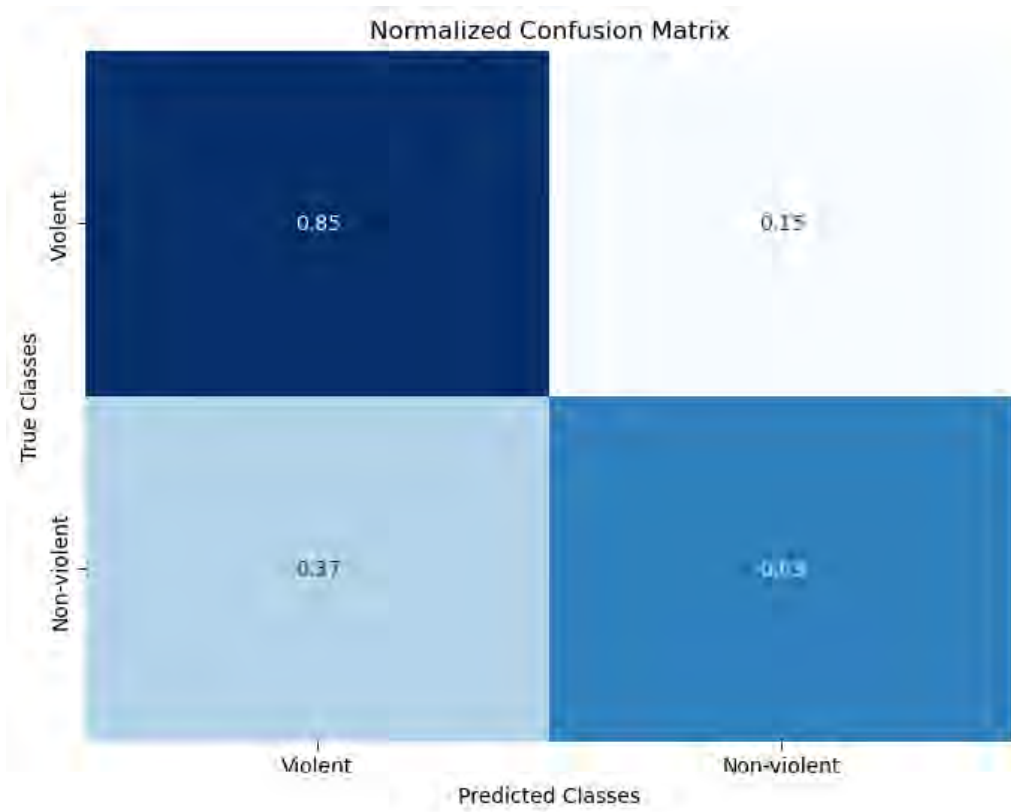


Figure 5.6: Confusion Matrix (ConvLSTM) for 2 classes

2 Class Conv-lstm				
Class Name	Precision	Recall	F1-Score	Support
Violent	0.61	0.85	0.71	159
Non-violent	0.86	0.63	0.73	160
accuracy			0.72	319
macro avg	0.73	0.74	0.72	319
weighted avg	0.76	0.72	0.72	319

Figure 5.7: Classification Report (ConvLSTM) for 2 classes

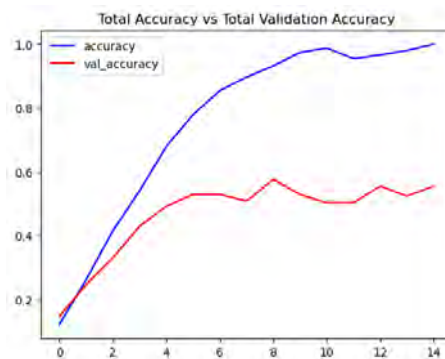


Figure 5.8: Training accuracy vs Validation accuracy (LRCN)

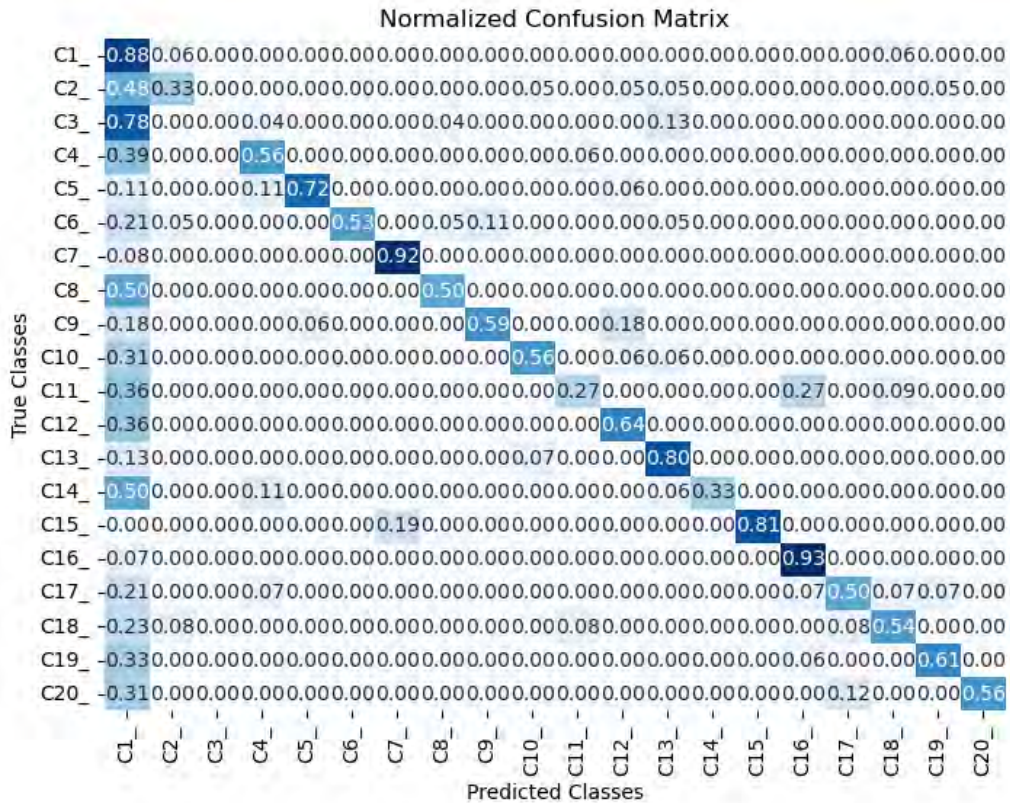


Figure 5.9: Confusion Matrix (LRCN) for 20 classes

According to the classification report, we can see that class 16 has performed extremely well compared to other classes. As we can see, class 16 has a precision of 72%, a recall of 93%, and an f1-score of 81%. We can also see that class 15 and class 7 have good performance in precision, recall, and f1-scores. Class 15 has a precision of 100%, a recall of 81%, and an f1-score of 90%. Moreover, class 7 has a precision of 80%, a recall of 92%, and an f1-score of 86%.

So, we observed that classes 16, 15, and 7 performed better compared to other classes. On the other hand, we can see that class 3 has performed the worst, having 0% precision, recall, and f1-score. This occurred due to some errors in class 3, as major frames of class 3 are corrupted, preventing the model from learning class 3. In addition, we can see that class 1 and class 2 also performed poorly compared to other classes. Class 1 has a low precision of 13% and an f1-score of 23%. Moreover, class 2 has a low recall score of 33%. Lastly, according to the report, the accuracy is 62%, the macro-average precision is 72%, the macro-average recall is 58%, and the macro-average f1-score is 60%.

As we have seen there is some major problem in class 1, class 2 and class 3 so we have discarded these 3 classes and checked how the model is performing. Now if we look at the confusion matrix we can notice that, previously with all classes, all of the classes were misclassifying with class 1 but now after discarding these three classes we can see that the diagonal line looks better comparing with previous confusion matrix with all classes. Eventhough few classes are misclassifying with class 4 but after removing those 3 classes the model (LRCN) performs better comparing to the all class. As we can see at the classification report with 17 class of LRCN, here class 14 and class 15 has performed better comparing to other classes. We can see that

20 Class LRCN				
Class Name	Precision	Recall	F1-Score	Support
class1	0.13	0.88	0.23	16
class2	0.7	0.33	0.45	21
class3	0	0	0	23
class4	0.62	0.56	0.59	18
class5	0.93	0.72	0.81	18
class6	1	0.53	0.69	19
class7	0.8	0.92	0.86	13
class8	0.75	0.5	0.6	12
class9	0.83	0.59	0.69	17
class10	0.82	0.56	0.67	16
class11	0.6	0.27	0.37	11
class12	0.54	0.64	0.58	11
class13	0.63	0.8	0.71	15
class14	1	0.33	0.5	18
class15	1	0.81	0.9	16
class16	0.72	0.93	0.81	14
class17	0.7	0.5	0.58	14
class18	0.7	0.54	0.61	13
class19	0.85	0.61	0.71	18
class20	1	0.56	0.72	16
accuracy			0.62	319
macro avg	0.72	0.58	0.6	319
weighted avg	0.71	0.56	0.59	319

Figure 5.10: Classification Report (LRCN) for 20 classes

these two classes has 93% of precision, 87% of recall and 90% of f1-score. Moreover, we can see that some classes got comparatively less score which are class 10, class 11 and class 12. In these classes we can see that these classes got less recall scores which is around 30% comparing to other classes. So, we can conclude that after removing the 3 classes, the model performs better than before. In this report the accuracy is 63%, the macro average precision is 75%, macro average recall is 64% and macro average f1-score is 66%.

In addition, as our dataset has two types of data one is violent and the other one is non-violent. The first twelve classes are violent and the remaining 8 are non-violent. That is how we have classified the dataset binary wise. We can see the confusion matrix below of the binary classification. As we can see the confusion matrix, the violent class has performed better with the prediction rate of 92% where non-violent class has the prediction rate of 84%. Here in the classification report we can see that both violent and non violent class has performed really well where violent class has precision of 80%, recall of 92% and f1-score of 86% whereas non-violent class has precision of 94%, recall of 84% and f1-score of 89%.

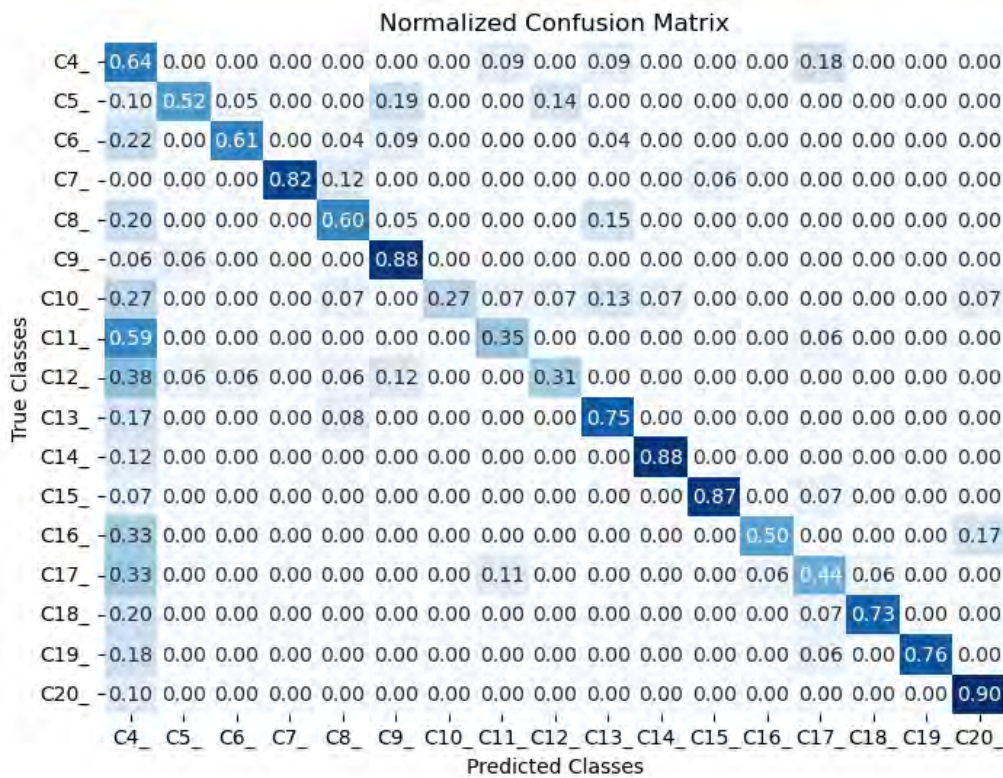


Figure 5.11: Confusion Matrix (LRCN) for 17 classes

5.3 Comparison

During our study, we used two separate models, namely Long Short-Term Recurrent Convolutional Networks (LRCN) and Convolutional Long Short-Term Memory (ConvLSTM). Now, we will conduct a comparison analysis of these two deep learning models, with a specific emphasis on their efficacy in effectively categorising diverse video footage.

Our assessment relies on various performance metrics, such as accuracy, recall, precision, and F1-score. In our analysis, it was discovered that LRCN achieved an accuracy of 0.62 for 20 classes, slightly outperforming ConvLSTM, which achieved an accuracy of 0.59. On the other hand, LRCN showcased a higher precision of 0.72 in comparison to ConvLSTM’s precision of 0.68. This indicates that LRCN is more appropriate when it comes to minimising false positives. In addition, LRCN showed a little bit higher score of F1 of 0.60 compared to ConvLSTM which had an F1-score of 0.54. Considering the exceptional performance of LRCN across all metrics, it is evident that LRCN is a superior model for classifying all 20 classes.

Nevertheless, when we evaluate the models on 17 classes, excluding the classes with similar data and classification challenges, we can observe that the ConvLSTM outperforms with an accuracy of 66%, while the LRCN achieves 63%. The recall trend is consistent with ConvLSTM achieving a score of 0.67, while LRCN achieves 0.64. However, in terms of precision, the LRCN models experience a significant increase of 75%, whereas ConvLSTM achieves a score of 0.67. Both the F1 scores show a similar value.

The LRCN demonstrates exceptional performance in accurately classifying violent and nonviolent superclasses, achieving high accuracy, recall, precision, and f1 score,

17 Class LRCN				
Class Name	Precision	Recall	F1-Score	Support
class4	0.11	0.64	0.19	11
class5	0.85	0.52	0.65	21
class6	0.88	0.61	0.72	23
class7	1	0.82	0.9	17
class8	0.67	0.6	0.63	20
class9	0.61	0.88	0.72	16
class10	1	0.27	0.42	15
class11	0.6	0.35	0.44	17
class12	0.56	0.31	0.4	16
class13	0.56	0.75	0.64	12
class14	0.93	0.88	0.9	16
class15	0.93	0.87	0.9	15
class16	0.86	0.5	0.63	12
class17	0.57	0.44	0.5	18
class18	0.92	0.73	0.81	15
class19	1	0.76	0.87	17
class20	0.75	0.9	0.82	10
accuracy			0.63	271
macro avg	0.75	0.64	0.66	271
weighted avg	0.77	0.63	0.66	271

Figure 5.12: Classification Report (LRCN) for 17 classes

all above 87%.

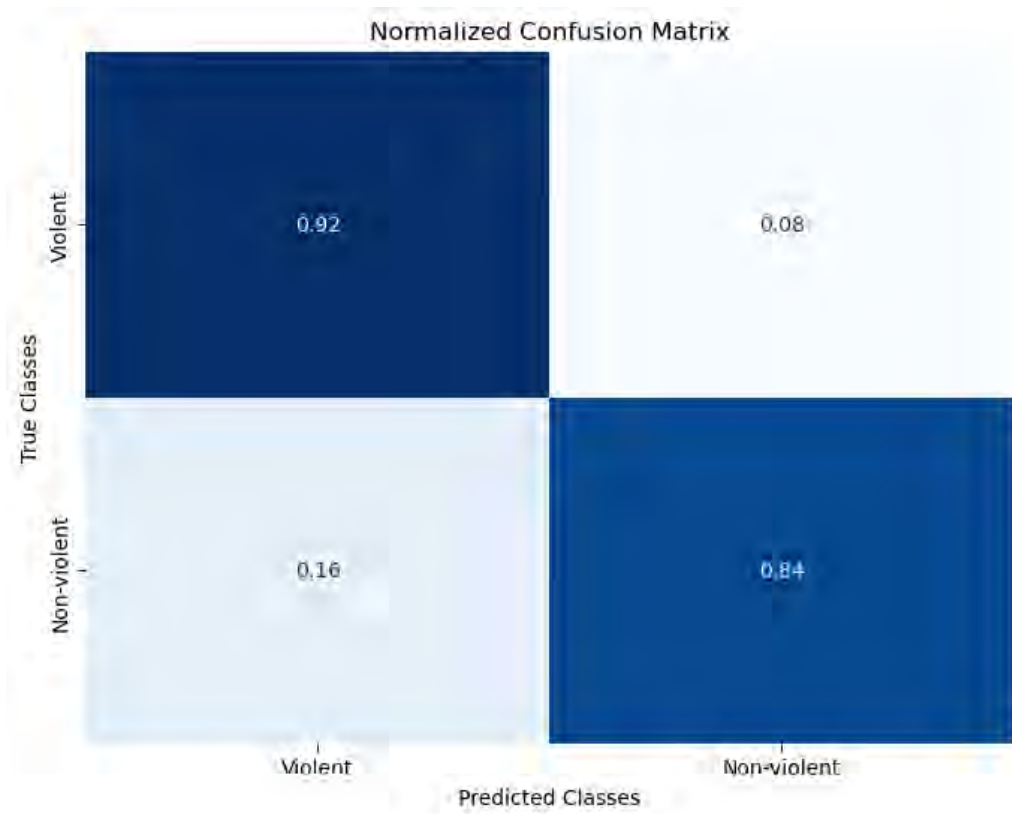


Figure 5.13: Confusion Matrix (LRCN) for 2 classes

2 Class LRCN				
Class Name	Precision	Recall	F1-Score	Support
Violent	0.8	0.92	0.86	159
Non-violent	0.94	0.84	0.89	160
accuracy			0.88	319
macro avg	0.87	0.88	0.87	319
weighted avg	0.88	0.88	0.88	319

Figure 5.14: Classification Report (LRCN) for 2 classes



Figure 5.15: Comparison for 20 classes



Figure 5.16: Comparison for 17 classes

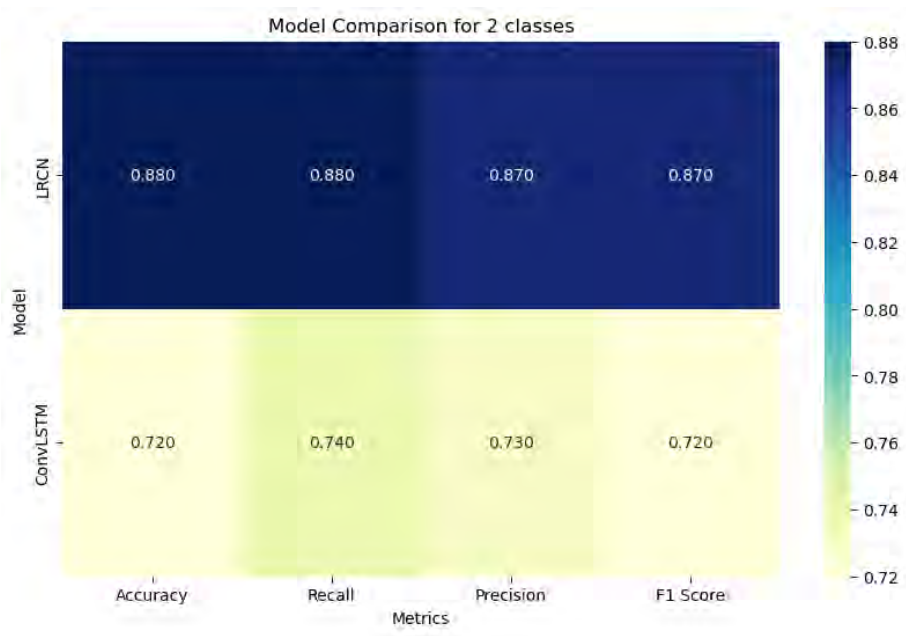


Figure 5.17: Comparison for 2 classes

Chapter 6

Conclusion

In summary, various sophisticated architectures have been created to control violent activities and to respond accordingly. However, completely eliminating the threat remains a substantial challenge. This paper seeks to enhance existing technologies for the efficient management and prevention of criminal activities. Unlike existing scholarly literature focusing on the technical aspects of systems, this study addresses the implementation of Human Activity Recognition (HAR), offering an application with significant implications for an important industry and the potential for transformative outcomes in modern society.

Bibliography

- [1] W. Ahmad, B. M. Kazmi, and H. Ali, "Human activity recognition using multi-head cnn followed by lstm," in *2019 15th international conference on emerging technologies (ICET)*, IEEE, 2019, pp. 1–6.
- [2] Z. Gao, H.-Z. Xuan, H. Zhang, S. Wan, and K.-K. R. Choo, "Adaptive fusion and category-level dictionary learning model for multiview human action recognition," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 9280–9293, 2019.
- [3] L. Wu, Z. Yang, J. He, *et al.*, "Ontology-based global and collective motion patterns for event classification in basketball videos," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 7, pp. 2178–2190, 2019.
- [4] C. Dai, X. Liu, and J. Lai, "Human action recognition using two-stream attention based lstm networks," *Applied soft computing*, vol. 86, p. 105 820, 2020.
- [5] N. Jaouedi, N. Boujnah, and M. S. Bouhleb, "A new hybrid deep learning model for human action recognition," *Journal of King Saud University-Computer and Information Sciences*, vol. 32, no. 4, pp. 447–453, 2020.
- [6] D. C. Luvizon, D. Picard, and H. Tabia, "Multi-task deep learning for real-time 3d human pose estimation and action recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 8, pp. 2752–2764, 2020.
- [7] C. Shiranthika, N. Premakumara, H.-L. Chiu, H. Samani, C. Shyalika, and C.-Y. Yang, "Human activity recognition using cnn & lstm," in *2020 5th International Conference on Information Technology Research (ICITR)*, IEEE, 2020, pp. 1–6.
- [8] J.-K. Tsai, C.-C. Hsu, W.-Y. Wang, and S.-K. Huang, "Deep learning-based real-time multiple-person action recognition system," *Sensors*, vol. 20, no. 17, p. 4758, 2020.
- [9] Z. Zhang, Z. Lv, C. Gan, and Q. Zhu, "Human action recognition using convolutional lstm and fully-connected lstm with different attentions," *Neurocomputing*, vol. 410, pp. 304–316, 2020.
- [10] S. Chakraborty, R. Mondal, P. K. Singh, R. Sarkar, and D. Bhattacharjee, "Transfer learning with fine tuning for human action recognition from still images," *Multimedia Tools and Applications*, vol. 80, pp. 20 547–20 578, 2021.
- [11] M. A. Khan, Y.-D. Zhang, S. A. Khan, M. Attique, A. Rehman, and S. Seo, "A resource conscious human action recognition framework using 26-layered deep convolutional neural network," *Multimedia Tools and Applications*, vol. 80, pp. 35 827–35 849, 2021.

- [12] B. S. Kumar, S. V. Raju, and H. V. Reddy, “Human action recognition using a novel deep learning approach,” in *IOP Conference Series: Materials Science and Engineering*, IOP Publishing, vol. 1042, 2021, p. 012 031.
- [13] C. Liu, J. Ying, H. Yang, X. Hu, and J. Liu, “Improved human action recognition approach based on two-stream convolutional neural network model,” *The visual computer*, vol. 37, pp. 1327–1341, 2021.
- [14] K. Muhammad, Mustaqeem, A. Ullah, *et al.*, “Human action recognition using attention based lstm network with dilated cnn features,” *Future Generation Computer Systems*, vol. 125, pp. 820–830, 2021, issn: 0167-739X. DOI: <https://doi.org/10.1016/j.future.2021.06.045>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0167739X21002405>.
- [15] K. Muhammad, A. Ullah, A. S. Imran, *et al.*, “Human action recognition using attention based lstm network with dilated cnn features,” *Future Generation Computer Systems*, vol. 125, pp. 820–830, 2021.
- [16] C. Plizzari, M. Cannici, and M. Matteucci, “Skeleton-based action recognition via spatial and temporal transformer networks,” *Computer Vision and Image Understanding*, vol. 208, p. 103 219, 2021.
- [17] H. Basak, R. Kundu, P. K. Singh, M. F. Ijaz, M. Woźniak, and R. Sarkar, “A union of deep learning and swarm-based optimization for 3d human action recognition,” *Scientific Reports*, vol. 12, no. 1, p. 5494, 2022.
- [18] J. Chen and C. M. Ho, “Mm-vit: Multi-modal video transformer for compressed video action recognition,” in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2022, pp. 1910–1921.
- [19] H. Duan, Y. Zhao, K. Chen, D. Lin, and B. Dai, “Revisiting skeleton-based action recognition,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2969–2978.
- [20] T. Guo, H. Liu, Z. Chen, M. Liu, T. Wang, and R. Ding, “Contrastive learning from extremely augmented skeleton sequences for self-supervised action recognition,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, 2022, pp. 762–770.
- [21] K. Hu, Y. Ding, J. Jin, L. Weng, and M. Xia, “Skeleton motion recognition based on multi-scale deep spatio-temporal features,” *Applied Sciences*, vol. 12, no. 3, p. 1028, 2022.
- [22] I. U. Khan, S. Afzal, and J. W. Lee, “Human activity recognition via hybrid deep learning based model,” *Sensors*, vol. 22, no. 1, p. 323, 2022.
- [23] V. Mazzia, S. Angarano, F. Salvetti, F. Angelini, and M. Chiaberge, “Action transformer: A self-attention model for short-time pose-based human action recognition,” *Pattern Recognition*, vol. 124, p. 108 487, 2022.
- [24] A. W. Muhamad and A. A. Mohammed, “A comparative study using improved lstm/gru for human action recognition,” 2022.
- [25] Y. Pang, Q. Ke, H. Rahmani, J. Bailey, and J. Liu, “Igformer: Interaction graph transformer for skeleton-based human interaction recognition,” in *Computer Vision—ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXV*, Springer, 2022, pp. 605–622.

- [26] Y.-F. Song, Z. Zhang, C. Shan, and L. Wang, “Constructing stronger and faster baselines for skeleton-based action recognition,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 45, no. 2, pp. 1474–1488, 2022.
- [27] Z. Sun, Q. Ke, H. Rahmani, M. Bennamoun, G. Wang, and J. Liu, “Human action recognition from various data modalities: A review,” *IEEE transactions on pattern analysis and machine intelligence*, 2022.
- [28] J. Xiao, L. Jing, L. Zhang, *et al.*, “Learning from temporal gradient for semi-supervised action recognition,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 3252–3262.
- [29] S. K. Yadav, K. Tiwari, H. M. Pandey, and S. A. Akbar, “Skeleton-based human activity recognition using convlstm and guided feature learning,” *Soft Computing*, pp. 1–14, 2022.
- [30] J. Zhang, G. Ye, Z. Tu, *et al.*, “A spatial attentive and temporal dilated (satd) gcnn for skeleton-based action recognition,” *CAAI Transactions on Intelligence Technology*, vol. 7, no. 1, pp. 46–55, 2022.
- [31] G. Jocher, A. Chaurasia, and J. Qiu, *YOLO by Ultralytics*, version 8.0.0, Jan. 2023. [Online]. Available: <https://github.com/ultralytics/ultralytics>.
- [32] A. Ray, M. H. Kolekar, R. Balasubramanian, and A. Hafiane, “Transfer learning enhanced vision-based human activity recognition: A decade-long analysis,” *International Journal of Information Management Data Insights*, vol. 3, no. 1, p. 100 142, 2023.