Bachelor of Science in Computer Science & Engineering

Traditional Bangladeshi Sports Video Classification Using Deep Neural Network

by

Moumita Sen Sarma

ID: 1504103

Department of Computer Science & Engineering

Chittagong University of Engineering & Technology (CUET)

Chattogram-4349, Bangladesh.

April, 2021

Traditional Bangladeshi Sports Video Classification Using Deep Neural Network

Submitted in partial fulfilment of the requirements for
Degree of Bachelor of Science
in Computer Science & Engineering

by

Moumita Sen Sarma

ID: 1504103

Supervised by

Dr. Kaushik Deb

Professor

Department of Computer Science & Engineering

Chittagong University of Engineering & Technology (CUET)

Chattogram-4349, Bangladesh.

The thesis titled '**Traditional Bangladeshi Sports Video Classification Using Deep Neural Network'** submitted by ID: 1504103, Session 2019-2020 has been accepted as satisfactory in fulfilment of the requirement for the degree of Bachelor of Science in Computer Science & Engineering to be awarded by the Chittagong University of Engineering & Technology (CUET).

Board of Examiners

Chairman (Supervisor)

Dr. Kaushik Deb
Professor
Department of Computer Science & Engineering
Chittagong University of Engineering & Technology (CUET)

Member (Ex-Officio)

Dr. Asaduzzaman
Professor & Head
Department of Computer Science & Engineering
Chittagong University of Engineering & Technology (CUET)

Member (External)

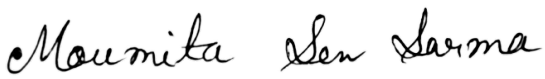
Dr. Md. Ibrahim Khan
Professor
Department of Computer Science & Engineering
Chittagong University of Engineering & Technology (CUET)

Declaration of Originality

This is to certify that I am the sole author of this thesis and that neither any part of this thesis nor the whole of the thesis has been submitted for a degree to any other institution.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. I am also aware that if any infringement of anyone's copyright is found, whether intentional or otherwise, I may be subject to legal and disciplinary action determined by Dept. of CSE, CUET.

I hereby assign every rights in the copyright of this thesis work to Dept. of CSE, CUET, who shall be the owner of the copyright of this work and any reproduction or use in any form or by any means whatsoever is prohibited without the consent of Dept. of CSE, CUET.



Signature of the candidate

Date: 19 April, 2021

Acknowledgements

The success and ultimate outcome of this thesis requires a lot of guidance and support from many individuals and I consider myself truly fortunate to have received this through the accomplishment of this thesis. It's been a worthwhile experience, both professionally and personally. Everything I've accomplished has been possible only because of such supervision and guidance. Above all, Thanks to almighty God for enabling me to complete this thesis successfully.

I would like to express my deep gratitude to my honorable thesis supervisor Dr. Kaushik Deb, Professor, Department of Computer Science & Engineering, Chittagong University of Engineering & Technology (CUET) for his guidance, encouragement and continuous support during my thesis work. I am thankful for his many critical questions, forcing me to see things from different perspectives, and his magnificent support throughout the entire time.

Finally, I want to thank my parents and brother for their unconditional love, support, encouragement, and contribution all throughout my life and academic career in every aspect along the years.

Abstract

Sports activities play a crucial role in preserving our health and mind. Due to the rapid growth of sports video repositories, automatized classification has become essential for easy access and retrieval, content-based recommendations, contextual advertising, etc. Traditional Bangladeshi sport is a genre of sports that bears the cultural significance of Bangladesh. Classification of this genre can act as a catalyst in reviving their lost dignity. In this paper, the Deep Learning method is utilized to classify traditional Bangladeshi sports videos by extracting both the spatial and temporal features from the videos. In this regard, a new Traditional Bangladeshi Sports Video (TBSV) dataset is constructed containing five classes: Boli Khela, Kabaddi, Lathi Khela, Kho Kho, and Nouka Baich. A key contribution of this work is to develop a scratch model by incorporating the two most prominent deep learning algorithms: convolutional neural network (CNN) and long short term memory (LSTM). Moreover, the transfer learning approach with the fine-tuned VGG19 and LSTM is used for TBSV classification. Furthermore, the proposed model is assessed over four challenging datasets: KTH, UCF-11, UCF-101, and UCF Sports. This model outperforms some recent works on these datasets while showing 99% average accuracy on the TBSV dataset.

Keywords: Traditional Bangladeshi Sports, Video Classification, Convolutional Neural Network (CNN), Long Short Term Memory (LSTM), Transfer Learning, Fine Tuning

Table of Contents

Acknowledgements **iii**

Abstract **iv**

List of Figures **viii**

List of Tables **ix**

1 Introduction **1**

1.1 Introduction 1
1.2 Bangladeshi Sports Video Classification Workflow 2
1.3 Challenges 3
1.4 Applications 4
1.5 Motivation 4
1.6 Contribution of the Thesis 5
1.7 Thesis Organization 6
1.8 Conclusion 6

2 Literature Review **7**

2.1 Introduction 7
2.2 Related Literature Review on Traditional Bangladeshi Sports Video Classification 7
2.2.1 Feature Extraction using Statistical Methods 7
2.2.2 Feature Extraction using Deep Learning Methods 9
2.3 Conclusion 11

3 Methodology **12**

3.1 Introduction 12
3.2 Steps of Proposed Traditional Bangladeshi Sports Video Classification Framework 12
3.3 Selection of Frames of Specific Length 13
3.4 Image/Frame Pre-processing 14
3.4.1 Resizing & Scaling 15
3.4.2 Augmentation Process 15
3.4.3 Normalization 16

3.5 Feature Extraction . . . 17
3.5.1 Spatial Feature Extraction . . . 18
3.5.2 Temporal Feature Extraction . . . 21
3.6 Classification . . . 25
3.7 Transfer Learning Approach with Pretrained Model . . . 27
3.7.1 Combination of Resnet152 v2 Network and Recurrent Neural Network . . . 28
3.7.2 Combination of Inception v3 Network and Recurrent Neural Network . . . 29
3.7.3 Combination of Xception Network and Recurrent Neural Network . . . 30
3.7.4 Combination of VGG16 Network and Recurrent Neural Network . . . 31
3.7.5 Combination of VGG19 Network and Recurrent Neural Network . . . 32
3.8 Fine Tuning Approach with VGG Models . . . 33
3.9 Implementation . . . 35
3.9.1 Hardware Requirements . . . 35
3.9.2 Software Requirements . . . 36
3.9.3 Deep Learning Optimizer . . . 36
3.9.4 Learning Rate . . . 36
3.9.5 Loss function . . . 37
3.9.6 Batch Size . . . 37
3.9.7 Epoch . . . 37
3.10 Conclusion . . . 38

4 Results and Discussions 39
4.1 Introduction . . . 39
4.2 Experimental Dataset Description . . . 39
4.2.1 Traditional Bangladeshi Sports Video (TBSV) Dataset . . . 39
4.2.2 KTH Dataset . . . 41
4.2.3 UCF-11 Dataset . . . 41
4.2.4 UCF-101 Dataset . . . 41
4.2.5 UCF Sports Dataset . . . 41
4.3 Impact Analysis . . . 42
4.3.1 Social and Environmental Impact . . . 42
4.3.2 Ethical Impact . . . 42
4.4 Evaluation Metrics . . . 42
4.4.1 Confusion Matrix . . . 43

4.4.2 Accuracy 44
4.4.3 Precision 44
4.4.4 Recall 44
4.4.5 F1 Score 45
4.5 Evaluation of Performance 45
4.5.1 Frame Length Selection 45
4.5.2 Effect of Normalization on Performance 46
4.5.3 Performance Evaluation of Scratch Model 46
4.5.4 Performance Evaluation of Pretrained Models 48
4.5.5 Performance Evaluation of Fine-tuned Pretrained Models . 49
4.5.6 Performance Evaluation of Proposed Model on KTH Sports, UCF-11, UCF Sports, and UCF-101 Dataset 53
4.6 Conclusion 55

5 Conclusion **56**
5.1 Conclusion 56
5.2 Future Work 57

List of Figures

1.1 Block Diagram of Traditional Bangladeshi Sports Video Classification Workflow. 3

3.1 Steps of Proposed Traditional Bangladeshi Sports Video Classification Workflow. 14
3.2 Overview of the Pre-processing Steps. 15
3.3 Image Resizing Process: (a) Original Image (b) Resized Image . . 15
3.4 Example of Augmented Images: (a) Original Image (b) Horizontally Flipped (c) Rotated(d) Width Shifted (e) Height Shifted. . . 17
3.5 Working Principle of Convolutional Layer. 19
3.6 Mechanism of Maxpool Layer. 20
3.7 ReLU Activation Function. 21
3.8 Overview of Convolutional Neural Network Architecture. 21
3.9 Basic Architecture of RNN. 22
3.10 Stretched-out Form of RNN Architecture. 22
3.11 Architecture of LSTM. 24
3.12 Overview of Dense Layer. 26
3.13 Overview of Network Architecture. 27
3.14 Residual Block with Skip Connection. 29
3.15 Complete architecture of VGG16 Network 31
3.16 Combined Architecture Based on VGG16 Network. 32
3.17 Complete Architecture of VGG19 Network. 33
3.18 Combined Architecture Based on VGG19 Network. 34

4.1 Sequential frames of (a) Boli Khela; (b) Kabaddi; (c) Kho Kho; (d) Lathi Khela; (e) Nouka Baich. 40
4.2 Frame Length vs. Accuracy Curve. 46
4.3 Accuracy and Loss Curve for Scratch Model. 47
4.4 Confusion Matrix for Scratch Model. 48
4.5 The feature Maps Produced by the Convolutional Part of Model-9 of (a) Boli Khela; (b) Kabaddi; (c) Kho Kho; (d) Lathi Khela; (e) Nouka Baich. 52
4.6 Performance of Model-9. 52
4.7 Confusion Matrix for Model-9. 53

List of Tables

3.1 Various Fine Tuning Network Configurations. 35

4.1 TBSV Dataset Details. 40
4.2 Effect of Normalization on Performance. 46
4.3 Class-wise Performance of the Scratch Model. 47
4.4 Performance Comparison of Some Pretrained Models on the TBSV Dataset. 48
4.5 Network Configuration and F1 Score of Fine-tuned VGG Models. 49
4.6 Class-wise Performance of Model-9. 53
4.7 Performance Comparison with Other Methods. 54

Chapter 1

Introduction

1.1 Introduction

Video data has been developed, distributed, and dispersed at a breakneck pace, becoming an indispensable component of today's big data. This stream has facilitated the research and development of new techniques for a wide variety of video classification applications that include an analysis of the video's contents. Classifying videos into different genres or categories is crucial because it allows effective cataloging and retrieval of vast video archives. The objective of video classification focuses on the automated tagging of video clips depending on their semantic contents like human actions or complex events.

In this context, sports video classification relates to the process of categorizing sports videos based on pertinent and complex sporting activity that includes spatial and motion characteristics. This process involves consideration of both series of actions and the surrounding aspects present in the video sequence. However, the core concept of this work is to explore and classify Traditional Bangladeshi Sports Video, which is quite a novel approach in this ground.

Recently, DNN or deep neural network-based models have been widely used and are quite effective in solving complex computer vision and signal processing tasks. Due to the advent of high-performance hardware, applications are not anymore confined to image recognition. Recently, deep neural network (DNN) models or recent deep neural nets have been shown to be particularly successful in tackling complicated signal and image processing problems. With the advent of high-performance computers, application possibilities have expanded beyond image recognition. However, video input based Deep Neural Networks are twice as

complex as their image counterparts. which demands larger memory and computational cost. Therefore, sports video classification is a fascinating and as well as a quite challenging task.

In this chapter, the overview of the proposed Traditional Bangladeshi Sports Video Classification system and also the challenges encountered in the process will be discussed. Motivation and applications of this specific thesis interest will also be stated in this chapter.

1.2 Bangladeshi Sports Video Classification Workflow

In the video, there exists a correlation between the subsequent frames in terms of their semantic contents. Therefore, if the temporal connections of the frames can be tracked, the classification system will be able to attain much promising result. So, sports video classification is an extended form of sports image classification in this context.

Convolutional Neural Network, a Deep Learning algorithm, has the efficiency in detecting local conjunctions of features from the previous layer and mapping their appearance to a feature map [1] . Although filters are hand-engineered in classical methods but with sufficient training, CNN has the potential to know these filters/characteristics by itself.

Another popular Deep Learning architecture, named as Recurrent Neural Network is extensively used for sequential or time-series data. It can extract temporal features from sequential data effortlessly which can be effectively used for video data to grab the temporal information as a video is a sequential collection of images.

However, at the very first of the approach, some pre-processing tasks need to be performed to scale and normalize the data.

So, to sum up, the block diagram of the Traditional Bangladeshi Sports Video Classification framework is shown in fig 3.1:

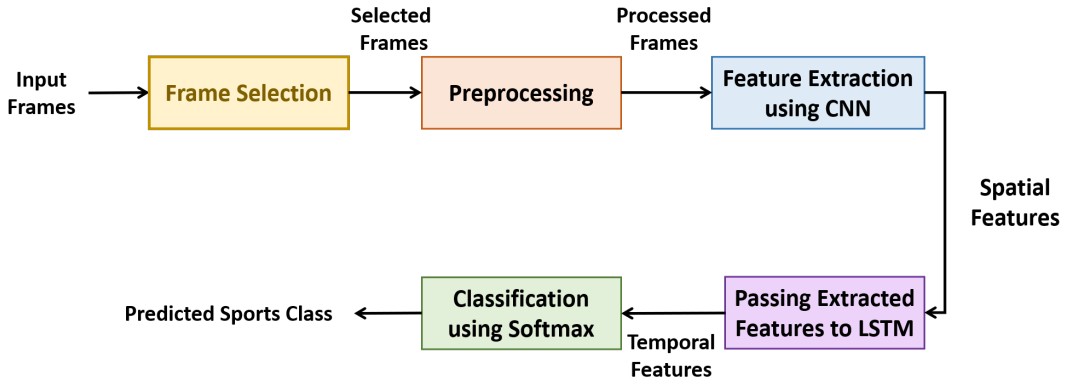


Figure 1.1: Block Diagram of Traditional Bangladeshi Sports Video Classification Workflow.

1.3 Challenges

Classification of Traditional Bangladeshi sports video poses some unique challenges to the Deep Learning models. Sports video classification task involves the identification of different actions and as well as capturing of spatial information from video clips. The major challenges in sports video classification are enlisted as follows:

- Capturing complex Spatio-Temporal features to detecting the sports activity
- Designing optimum model amidst various possible combinations of architectures and hyperparameters
- Huge computational cost due to working with 5d data shape of video i.e. (samples, frames, height, width, channels) integrated with Deep Neural Network
- Variable distance and uneven illumination condition
- Change in viewpoint
- Change in scale and orientation.
- Scarcity of Traditional Bangladeshi Sports Video dataset.

1.4 Applications

Sports video categorization is a broad research field that has recently grown in popularity, owing to commercial sports concerns. Enormous amounts of sports footage are filmed every day, for internal post-game review of results and strategy, for TV productions of major sports teams, or personal memories, often for sharing with friends. Likewise, Traditional Bangladeshi Sports Videos are also drawing attention to users due to cultural and social interest. The indexing of individual sports activity according to their category, is a vital task for further exploration. So, sports video classification has an extensive range of application areas, some of which are:

- Automatic organization of videos according to the categories
- Sports Video Searching
- Automatic Recommendation System
- Managing Archives of a huge collection of sports videos.
- Recommendation of Ads based on sports interest of the users.
- Match Analysis

1.5 Motivation

Computer Vision and Artificial Intelligence technologies are becoming trendy in the study of analyzing videos. However, the area of computer vision is currently evolving from statistical approaches to deep learning neural network methods due to their efficiency in extracting complex features without human intervention. Furthermore, in recent times, deep learning models integrated with a wide variety of configurations, have outperformed existing state-of-the-art approaches. So, there is a huge scope for commencing innovation and development in this evolving research arena.

Likewise, sports video classification is quite a novel research interest, and a couple of methodologies have been established for this task. However, there is still

room for improvement and exploration on this ground. Moreover, implementation of this task on Traditional Bangladeshi Sports Video dataset is totally a fresh conception which seemed quite fascinating to work with.
Therefore, the key motivations behind this thesis are stated below:

- Starting with a fresh new branch.
- Difficulty in finding a sports video file from a large video collection
- Automatically classifying for easy access and retrieval.
- Overcoming the limitations of existing works.
- Endeavoring to revive the traditional extinct Bangladeshi sports.

1.6 Contribution of the Thesis

The objective of a thesis or research work is to contribute to the development of human knowledge by upgrading existing accomplishments with some logic and facts. In this thesis work, the main concern is to work with an entirely new concept, i.e., implementing Deep Learning algorithms on a newly created dataset of traditional Bangladeshi sports videos and achieving a promising result in this task. The key contributions of this thesis are the following:

- Constructing a new dataset of 500 traditional Bangladeshi sports videos of five categories
- Extracting both Spatial and Temporal features through building a scratch network to detect the traditional Bangladeshi sports activity
- Implementation of transfer learning approach integrated with additional layers to successfully classify the traditional Bangladeshi sports videos
- Exploring fine tuning approach with the two best performing pretrained models, integrated with some custom layers.
- Rejuvenating the extinct traditional Bangladeshi sports as well as unleash a new section in the Deep Learning field.

1.7 Thesis Organization

The rest of this thesis report is organized as follows:

- Chapter 2 gives a brief summary of previous research works in the field of Traditional Bangladeshi Sports Video Classification
- Chapter 3 describes the proposed methodology for categorizing of Traditional Bangladeshi Sports Video
- Chapter 4 explains the details of working dataset and represents the analysis of the performance of the proposed framework
- Chapter 5 contains the overall summary of this thesis work and provides some future recommendations as well

1.8 Conclusion

In this chapter, an overview of Traditional Bangladeshi Sports Video Classification framework is provided. In addition to the challenges, the summary of the Traditional Bangladeshi Sports Video Classification framework is described in this chapter. The motivation behind this work is also stated here. In the next chapter, background and present state of the problem will be discussed.

Chapter 2

Literature Review

2.1 Introduction

As stated earlier, the categorization of sports video is quite a new branch in the evolving field of Deep Learning technologies. This concept has drawn the attention of some researches who got motivated by the novelty of this task. On the other hand, before the evolution of Deep Learning techniques, various statistical and classical methods were used by some researchers for categorizing sports videos. However, a few existing methodologies are found for classifying Traditional Bangladeshi Sports Video as the extensibility of this task is still lurking. With a brief summary of the previous study, this chapter discusses various feature representation and classification methods applied by different researches through statistical and Deep Learning approaches, and the performances of these researches on different datasets.

2.2 Related Literature Review on Traditional Bangladeshi Sports Video Classification

2.2.1 Feature Extraction using Statistical Methods

In the early stage, various statistical and classical methods were used to extract the features from inputs manually. The extracted features lead to the final classification of different categories of videos. Some statistical methods used in related works to the sports video classification task to extract features are illustrated here.

In [2], sports categories were classified from user-generated mobile videos by analyzing audio, visual and sensor data and exploring adaptive fusion coupled with multi-user and multimodal data, along with multiclass SVM classifier. However, significant confusion had been found among sports having similar semantic views. However, edge features of 500 sports videos obtained from Non-Subsampled Shearlet Transform (NSST) were analyzed by the authors in [3] and KNN was used as a classifier to categories five sports video types. Besides, authors in [4] proposed an automatic sports image classification method under ASSAVID system and used KNN as classifier; however, a few images were misclassified as the wrong sport. On the other hand, the authors in [5] categorized sports video types from signature heatmaps and used Fisherfaces algorithm as principal component analyzer and finally, Euclidean distance was used for final classification yet only sport played in match-like situations were concentrated on. In [6], the authors considered audio and visual features extracted from the thermal video for classifying 180 1-minute video sequences from three sports types where four motion features were extracted and integrated directly with MFCC audio features for classification by using K-Nearest Neighbor as the classifier.

However, authors in [7] fused two HMMs representing color and motion features and used the Baum-Welch algorithm to classify 220 minutes of sports video with four genre types. Likewise, in [8], a Hidden Markov Model (HMM) based classification approach integrated with Baum-Welch algorithm was proposed to categorize 3 genres of sports videos where speed of color changes was computed for each video frame. Authors in [9] explored Mel-Frequency Cepstral Coefficients, Global Camera Motion, and Dominant Color features for audio-visual analysis of three categories of sports videos and used the HMM based model for classifying them. In [10], four different events, i.e. goal-kick, placed-kick, shot-on-goal, and throw-in of soccer video dataset were classified through observing ball directions using Hidden Markov Model. Moreover, the time performance of their experiment on the soccer event classification module was also represented.

2.2.2 Feature Extraction using Deep Learning Methods

Sports video classification refers to classify sports according to their categories by extracting spatial and motion features present in the sequential frames. Recently, Computer Vision and Artificial Intelligence technologies are becoming trendy in the study of analyzing videos. As a significant part of these technologies, Deep Learning based methods have been proven to extract complex features from images automatically. In this context, several researchers have worked on the two most recent deep learning techniques Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) to extract spatial and temporal features from sports videos. Some deep learning methods used in related works to the sports video classification task to extract features are discussed here.

In [11], a combined architecture of dilated CNN and RNN was introduced for classifying five categories of sports from sequential frames. However, more high-level features could not be extracted due to the inability of implementing deep architecture with large enough input size because of hardware limitations. Moreover, in the extension of their research work, in [12], the authors proposed two types of scratch model combining CNN and RNN and a transfer learning approach with pre-trained VGG16 model incorporated with RNN on two datasets named SportsC10 and SportsC15 of 10 and 15 categories of sports respectively. In this case, the transfer learning approach integrated with RNN came up with the most promising performance.

Carrying out a comparative analysis of several feature extraction methods and classifiers in [13], the authors used Median Filter with various window lengths to filter classification results for frame sequences in continuous TV broadcasts where CNN was used as a feature extractor. Two types of supervised classification approach named SVM and Random Forest were used as a classifier in which SVM came out to be the best one, better performance was achieved only in offline. Furthermore, extensive research was conducted by the authors of [14] where they introduced a multiresolution, foveated architecture for speeding up the training and three types of fusion model were implemented on a dataset of 1M YouTube videos containing 487 sports classes.

However, in [15], a Deep Neural Network combining CNN and RNN was constructed for the detection of soccer video event where semantic features are captured from key frames using CNN by categorizing nine classes of soccer frame images as per their different semantic views and thus constructing a dataset called Soccer Semantic Image Dataset(SSID). Likewise, in [16],an action recognition framework was established on three benchmark datasets including HMDB51, UCF-101, and YouTube 11 Actions through extracting deep features from every sixth frame of the videos by utilizing CNN. Later the sequential information was mapped with Deep Bi-directional LSTM, which yielded improved results over state-of-the-art action recognition approaches.

On the other hand, an AlexNet-based architecture was used in [17] for four types of shot classification categorized as close-up, long, crowd/out-of-the-field shots, and medium in field sports videos of cricket and soccer collected from Youtube. The comparative analysis was conducted with state-of-the-art frameworks which proved the outperformance of proposed architecture. Additionally, the authors of [18] constructed a hybrid model by combining GMM and Kalman Filters with GRU in classification tasks over KTH, UCF 101, and UCF Sports datasets. This model produced promising results relative to some existing approaches on these datasets. By analyzing the temporal coherence, Ge et al. [19] an attention mechanism based GoogleNet-LSTM model was proposed that showed impressive performance to recognize actions in UCF-11, HMDB-51, and UCF-101 datasets. Their method was capable of effectively grabbing the salient areas of actions in videos. However, in [20] spatio-temporal features learned from CNN based principal component analysis network (PCANet) was used with bag-of-features and encoding schemes in KTH and UCF Sports action recognition dataset and attained promising results. Likewise, the authors of [21] introduced a novel recurrent attention CNN for focusing the crucial regions in videos to detect actions in UCF-11, HMDB-51, and UCF-101 datasets.

However, a very minimal amount of significant research has been carried out in the field of recognizing traditional Bengali sports. Whereas, in [1], transfer learning approach was used through retraining the last layer of renowned Inception

V3 architecture developed by Google for classifying five traditional Bangladeshi sports categories i.e. 'Danguli', 'Kabadi', 'Kanamachi', 'Latthi Khela' and 'Nouka Baich' from augmented images of a dataset having in total 3600 images yielding decent result in this ground.

2.3 Conclusion

This chapter includes a comprehensive overview of the literature review. The discussion is divided into two parts according to the mechanism of extraction of features for convenience. Different feature extraction techniques and classifiers used by the researchers are described here. The next chapter contains an explanation of the methodology of the traditional Bangladeshi sports video classification process in detail.

Chapter 3

Methodology

3.1 Introduction

Videos are a collection of sequential images, and sports videos are not an exception to it. There remains temporal connectivity among the sequential frames of the videos. For classifying different sports categories, it is essential to capture both the spatial and temporal relationships among the successive frames of the videos to recognize the action present in the sports video. Recently, Convolution Neural Network has been proven to extract complex spatial features among the input images by capturing the local patterns in the images. Furthermore, the extracted spatial features can be sequentially handled by Recurrent Neural Network, a powerful and robust type of neural network that can form a much deeper understanding of a sequence and its context compared to other algorithms.
In this research work, a combined network consisting of Convolution Neural Network and Recurrent Neural Network incorporated with multiclass classifier is proposed for the classification of traditional Bangladeshi sports videos.

3.2 Steps of Proposed Traditional Bangladeshi Sports Video Classification Framework

Figure 3.1 shows the basic steps of the proposed methodology for Traditional Bangladeshi Sports Video Classification framework. As states earlier, video is a collection of sequential frames, and there exists a temporal relationship among the frames. So, a specified number of sequential frames are needed to be selected for analyzing the video data. In this regard, initially, the required number of sequential RGB color frames are selected from the video input data. Next, the

selected frame i.e. images are preprocessed through resize and rescale, augmentation and normalization operations for further analysis. Then, spatial features are extracted simultaneously from the frames of a video through the Convolutional Neural Network. After that, extracted spatial characteristics of the frames are sequentially analyzed by LSTM to extract the temporal relations among the frames. A dense layer or fully connected layer is added after that. Finally, Softmax activation is used in the output layer to classify the video into five categories based on the extracted spatial and temporal features.

In the training part of the classification stage, the feature vector set, calculated by LSTM is fed to a multi-class Softmax classifier along with the label for each sample. The testing part is executed using this trained network. In this part, the label for each of the testing samples is predicted using the trained network.

3.3 Selection of Frames of Specific Length

The dataset that has been constructed for this research work consists of a total of 500 traditional Bangladeshi sports videos collected from Youtube. In this dataset, five classes have been selected: Boli Khela, Kabaddi, Kho Kho, Lathi Khela and Nouka Baich. For each class, 100 videos are taken. Each video has 150 sequential frames having frame rate of 30 per second and is 5 secs in length. The resolution of the frames is 1280x720 pixels.

However, we carried out an experiment using the scratch network to pick the right frame length, with 100 video samples in each class. The performance comparison according to frames is briefly explained in this thesis work. From the experiment, it has been observed that the network exhibits the best result for the case of frame length 20. Therefore, 20 sequential frames have been picked from each video for the whole experiment. Moreover, the frame length 20 is the optimum one which enhances the performance and reduces computation cost and time. So, after the frame selection step, we've come up with 500 videos each containing 20 frames, i.e. in total 10,000 images/frames.

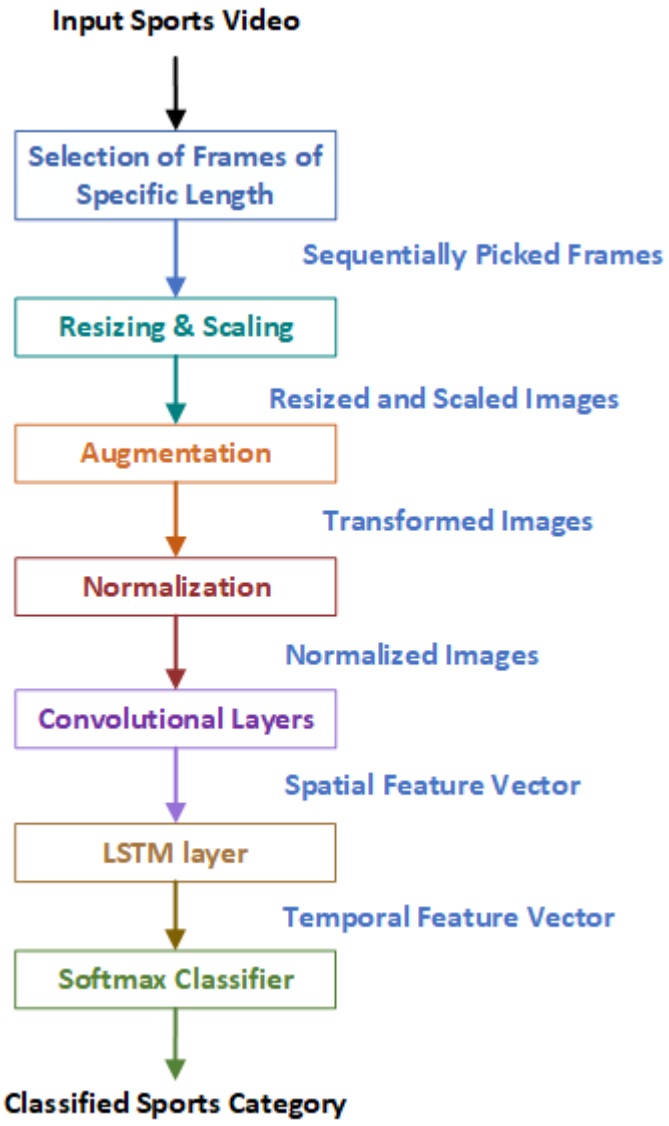


Figure 3.1: Steps of Proposed Traditional Bangladeshi Sports Video Classification Workflow.

3.4 Image/Frame Pre-processing

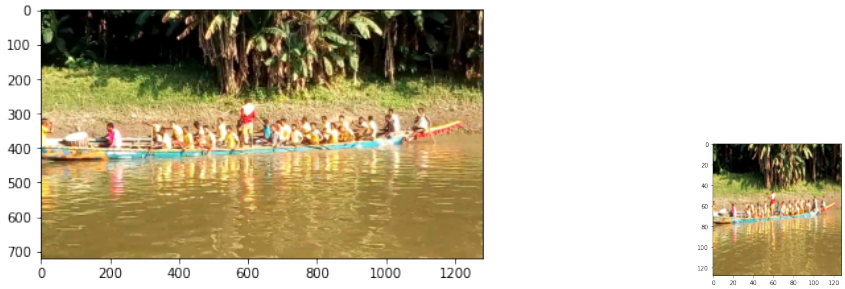
Image processing refers to improving image data (properties) by removing unnecessary artefacts and/or enhancing certain essential image features so that the deep learning methods can take advantage of this enhanced data. It has been proven that pre-processed images are capable of bringing about a radical change on the deep neural network's output. In this context, pre-processing is the first step of the Traditional Bangladeshi Sports Video Classification framework. However, the pre-processing stage is comprised of three parts, i.e. resizing and scaling,

augmenting and normalizing the images/frames. These steps are depicted in figure 3.2:

Figure 3.2: Overview of the Pre-processing Steps.

3.4.1 Resizing & Scaling

Image resizing is required for reducing computational cost and time. For boosting the performance and also for the sake of reducing cost and time, each RGB frame/image has been resized to 128x128 pixels which were originally in size of 720x1280 pixels. Then we've scaled down the RGB color images by the factor of 255 for the sake of treating all images equally. Moreover, the original images consist of RGB coefficients in the 0-255, but these values will be too high for the deep learning model to be processed.So, we've converted the pixel values to the [0, 1] interval by scaling with a 1/255.0 factor. Figure 3.3 shows the output of the image resizing process.



(a) (b)

Figure 3.3: Image Resizing Process: (a) Original Image (b) Resized Image

3.4.2 Augmentation Process

Augmentation is a technique of producing more training samples from the existing ones via some random variations through some specified transformations. More data means more results for deep neural networks. Training of deep learning models on more data can result in more efficient models, and the augmentation

techniques can generate variations of the images that greatly boost the generalization ability of the models.

Moreover, modern deep learning algorithms, such as the Convolutional Neural Network (CNN) can learn translational invariant features. Nevertheless, augmentation can further aid in this translational invariant approach to learning and can assist the model in capturing features that are also invariant to transforms. So, the aim is to add new, plausible examples to the training dataset.

After resizing and scaling the frames, the augmentation technique has been applied to those processed frames. In this work, online, i.e. real-time augmentation has been used, which creates transformed images at each epoch of training. However, we've considered four different methods for transformation: horizontal flip, width shift, height shift, rotation. This technique contributes to the better generalization of the network. Figure 3.4 shows examples of augmented images.

3.4.3 Normalization

Normalization is added to standardize raw input pixels which will transform input pixels by subtracting the batch mean, and then dividing by the batch's standard deviation which makes mean 0 and standard deviation of 1. If data is not normalized, we may have some numerical data points in our data set that might be very high, and others that might be very low. In unnormalized data, thus naturally large values become dominant according to relatively small values during training as the importance of each input is not equally distributed. When we normalize our inputs, however, we put all of our data on the same scale, which makes convergence faster while training the network.

For unnormalized data, as gradient descent might need a lot of steps and longer time to converge, a very small learning rate has to be used. Whereas for normalized data, a more spherical curve is obtained in which wherever it is started gradient descent can go straight to the minimum in much less time and oscillation. Therefore, when we have normalized our training data, however, we have put all of our data on the same scale, which makes convergence faster while training the network. Effect of normalization on the performance of the model is briefly explained in this thesis work.

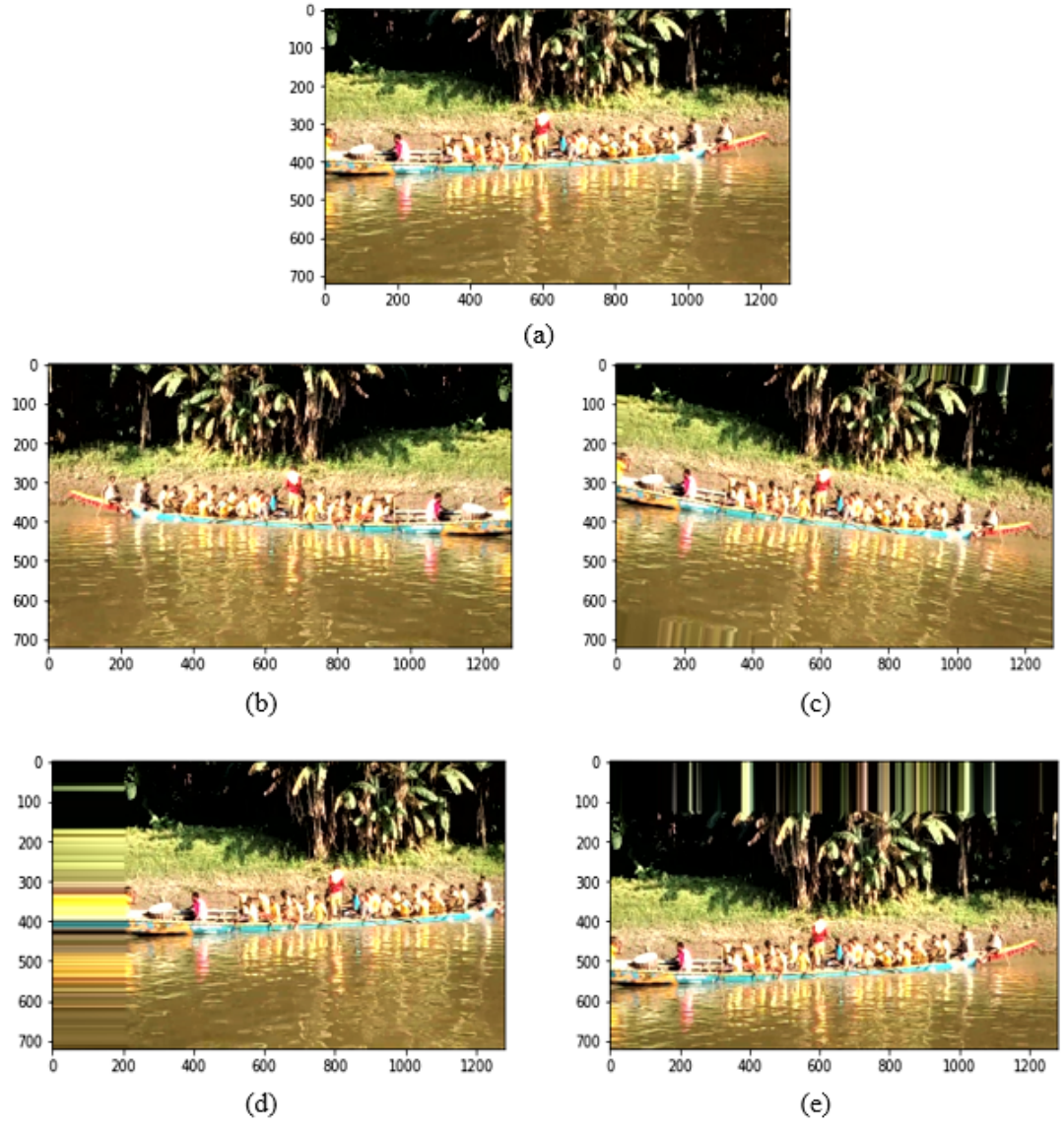


Figure 3.4: Example of Augmented Images: (a) Original Image (b) Horizontally Flipped (c) Rotated(d) Width Shifted (e) Height Shifted.

3.5 Feature Extraction

Feature extraction is the process of transforming input data into a set of features. Features are unique characteristics of input data that is useful to categorize among various categories of inputs. This process is also termed as a dimensional reduction mechanism where the original collection of raw input variables is compressed to more adaptable groups or features. This process results in more informative and non-redundant data that helps in subsequent learning and generalization steps.

However, feature extraction is the third step in this proposed framework. There

remains a spatial and temporal connection among the subsequent frames in the sports video. So, both spatial and temporal feature extraction are equally vital in categorizing the videos with a decent result. So, the main focus of this thesis work is to construct a combined model comprised of both semantic and time-domain feature extraction techniques.

3.5.1 Spatial Feature Extraction

Spatial features can be specified as the information regarding surrounding or context of the scenes in videos. In the sports video, spatial features can be the surrounding aspects of the playground, characteristics of the players and everything about the view of the events. Recently, Convolutional Neural Networks have been recognized as one of the most effective architectures for extracting complex spatial features of the images. Convolutional layers can learn local and translation invariant patterns from the images which are quite useful for the classification task.

The integral component of the convolutional neural network is the convolutional layer that takes after the name of this network. This layer conducts a “Convolution” process. Convolution is a linear mathematical operation that performs matrix multiplication between the filter of a specific dimension and the portion of the image on which the filter hovers. The filter is a 2-D array of weights that facilitates the extraction of spatial features. Initially, the weights or values of the filters are chosen randomly. However, as the training process continues, the weights get updated as per the loss function’s result. The filter moves to the right by the value of stride, i.e. the number of pixels the filter shifts over the input matrix, until it covers the total width of the image. The outcome of a convolutional layer is a 2-D output array that is termed as "Feature Map". After the creation of the feature map, each value in the feature map is transferred to an activation function that introduces nonlinearity in the output result. Without an activation function, the Convolutional Neural Network will be merely a linear regression.

Another essential component of the convolutional layer is "Padding" that refers to add layers of zeros to the input images that reduces the problem of shrank

outputs and loss of information on corners of the image. However, there are two types of padding: valid padding and the same padding. The valid padding implies no padding operation at all. On the other hand, the same padding operation adds padding layers in a way that the output image has the same dimension as the input image. The mathematical equations of the dimension of the output image of the convolutional layer with same and valid padding are specified in equation 4.4:

$$Output\_Width = \frac{W - F_w + 2P}{S_w} + 1$$
$$Output\_Height = \frac{H - F_h + 2P}{S_h} + 1 \tag{3.1}$$

In equation 4.4, W and H refer to the width and height of the input image. However, F_w, F_h, S_w, S_h represent the width and height of the filter and stride respectively. P is the value of padding which equals to 1 if it is same padding otherwise it is 0.

So, it can be said that a convolutional layer consists of three key parts: filter, stride and padding. Figure 3.5 depicts how a convolutional layer works on the input image.

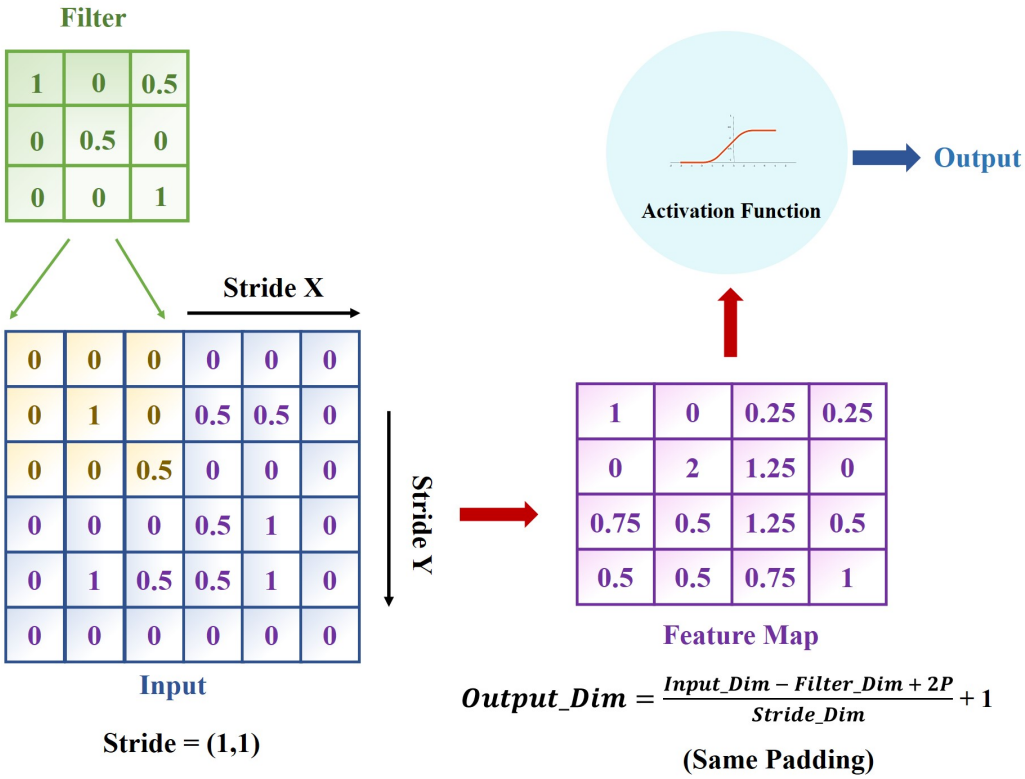


Figure 3.5: Working Principle of Convolutional Layer.

Max Pooling layer is another vital element of convolutional neural network that has come up with down sampling strategy. This layer works by calculating the maximum value from the region of the feature map covered by the filter. The key purpose of this layer is to accumulate the most activated presence of a feature. Its purpose is to gradually decrease the spatial size of the representation to minimize the number of parameters and the computation cost of the network. Max pooling operates by applying a max filter to the non-overlapping subregions of the initial representation. Suppose, there is a 4x4 matrix representing the initial input and a 2x2 filter that will be run over the input. A stride of 2 has been picked. For each of the regions covered by the filter, the max of that region is considered. Thus, a new, output matrix is created where each element is the max of the covered area in the original input. The working mechanism of the max-pooling layer is illustrated in figure 3.6.

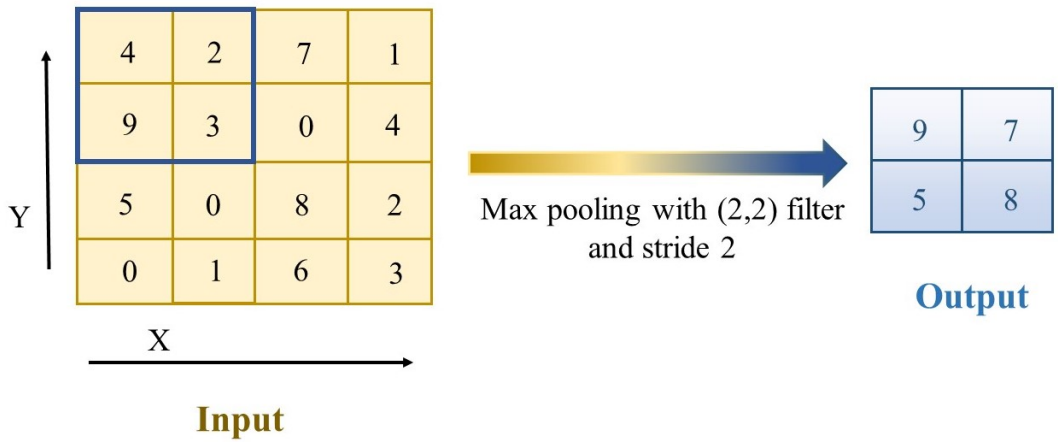


Figure 3.6: Mechanism of Maxpool Layer.

However, for extracting the spatial features from traditional Bangladeshi sports videos, a convolutional neural network has been constructed, which is comprised of 8 layers. In this network, the same padding is used, and the stride dimension is (1,1). For faster convergence during training, ReLu is used as an activation function which provides output to$[0, \infty]$ interval. Fig 3.7 plots the outcome originated from the calculation of ReLU function on each input of a series of integers from -20 to 20.

In this work, as a spatial feature extractor, a TimeDistributed CNN architecture has been constructed. The input dimension of a TimeDistributed CNN layer is (samples, frames per video, height, width, channels), where 'samples' refers to

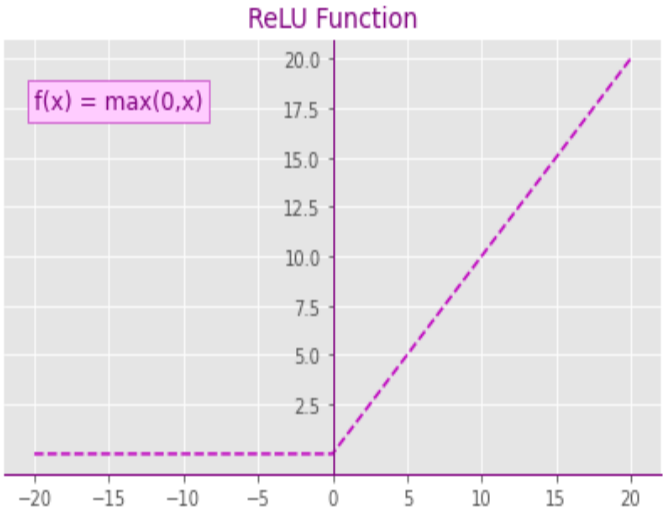


Figure 3.7: ReLU Activation Function.

the no. of samples in each batch. This architecture employs the same layer of this architecture to all the picked frames of each video one by one through sharing same weights and thus finally produces feature maps for the frames. The whole convolutional neural network architecture built for this research work is depicted in figure 3.8.

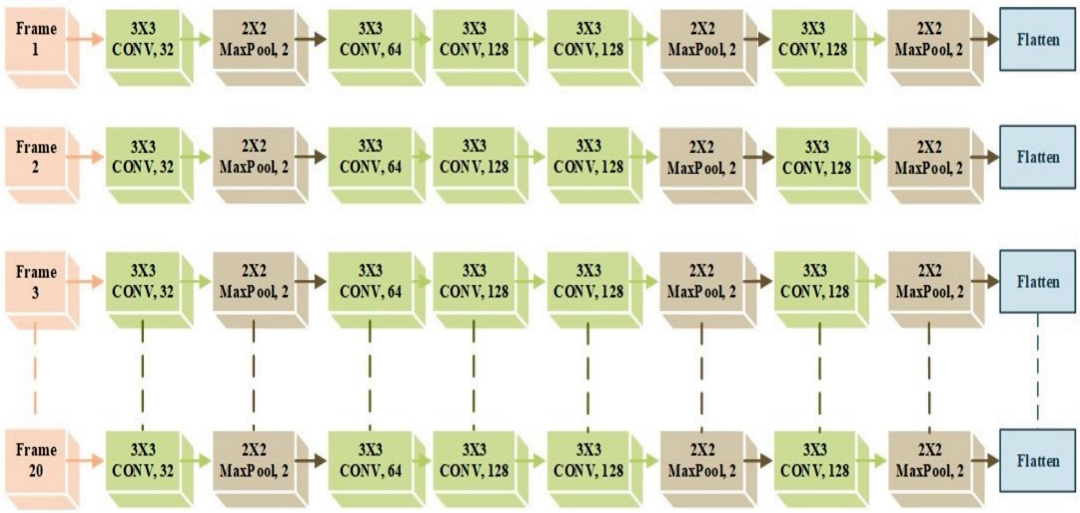


Figure 3.8: Overview of Convolutional Neural Network Architecture.

3.5.2 Temporal Feature Extraction

In a sports video, there remains temporal connectivity among the sequential frames. In this regard, the Recurrent Neural Network can play a vital role. In Recurrent Neural Network, not only the current input but also the previously

received inputs are taken under consideration for grabbing the changes of motion present in the successive frames. This network can be considered as an extension of Feedforward network that allows to process variable length of sequences.

The basic architecture of RNN depicted in figure 3.9 is quite identical to that of Feedforward network except for the loop. Through iterating over the sequence elements, RNN processes those sequences and maintains a state consist of the relevant information that it has seen so far.

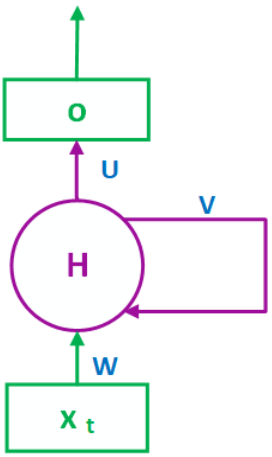


Figure 3.9: Basic Architecture of RNN.

If we stretch out the basic recurrent neural network, it can be seen that it's a network of nodes, alike to neurons arranged in consecutive layers. Fig 3.10 shows the stretched out form of RNN architecture. There are three types of node

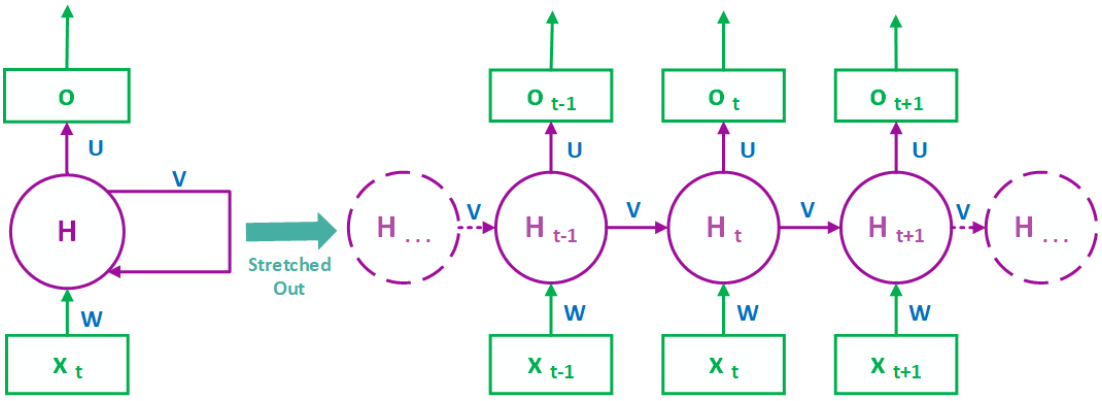


Figure 3.10: Stretched-out Form of RNN Architecture.

present in RNN: input nodes, hidden nodes and output nodes. All the nodes are connected with one another through a directed connection that has a real valued weight. Weights are assigned to input to hidden node connection (W), hidden to

hidden node connection (V) and hidden to output node connection (U). Initially, it gets the x_0 from the sequence of input. After that, the output h_0 along with x_1 is the input for the next step. So, the h_0 and x_1 are considered as the inputs for the next step. Likewise, h_1 from the next with x_2 works as input for the next step and so forth. However, RNN works by following set of equations:

$$Hidden\ State,\ h_t = f(\ b + Vh_{t-1} + Wx_t\ ) \tag{3.2}$$

$$Output,\ o_t = c + Uh_t \tag{3.3}$$

$$Probabilistic\ Output,\ \hat{y}_t = softmax(\ o_t\ ) \tag{3.4}$$

In equation 3.2, *Hidden State* is considered as memory to the RNN cell, f is a non-linear transformation function and b refers to bias value. Likewise, in equation 3.3, c also represents a bias value. In equation 3.4, $softmax$ is a function that provides us with a vector $\hat{y}$ of normalized probabilities over the output.
However, traditional RNN suffers from short term memory, i.e. it is incompetent in retaining information for longer periods. In this regard, LSTM, i.e. Long Short Term Memory was developed as a remedy to short term memory problem. LSTM is a Recurrent Neural Network which was introduced to capture long term dependencies in sequential data. Moreover, LSTM has come up with more controlling power on information flow in the network. It also reduces the vanishing gradient problem to an extent. Two core parts of LSTM are cell state and gate. The cell state holds salient information while the gates decide which related information is required to remember and which to forget. LSTM has three types of gates to control the flow of information to the cell: forget gate, input gate and output gate. The architecture of LSTM is illustrated in figure 3.11.
The cell state works almost like a conveyor belt that runs through the entire chain with just a few linear operations. The gates of LSTM are capable of adding or erasing information to the cell state.
Initially, forget gate of LSTM decides which information are going to be eliminated from the cell state through a sigmoid function. Forget gate works through the following equation:

$$f_t = \sigma(W_f.[H_{t-1}, X_t] + b_f) \tag{3.5}$$

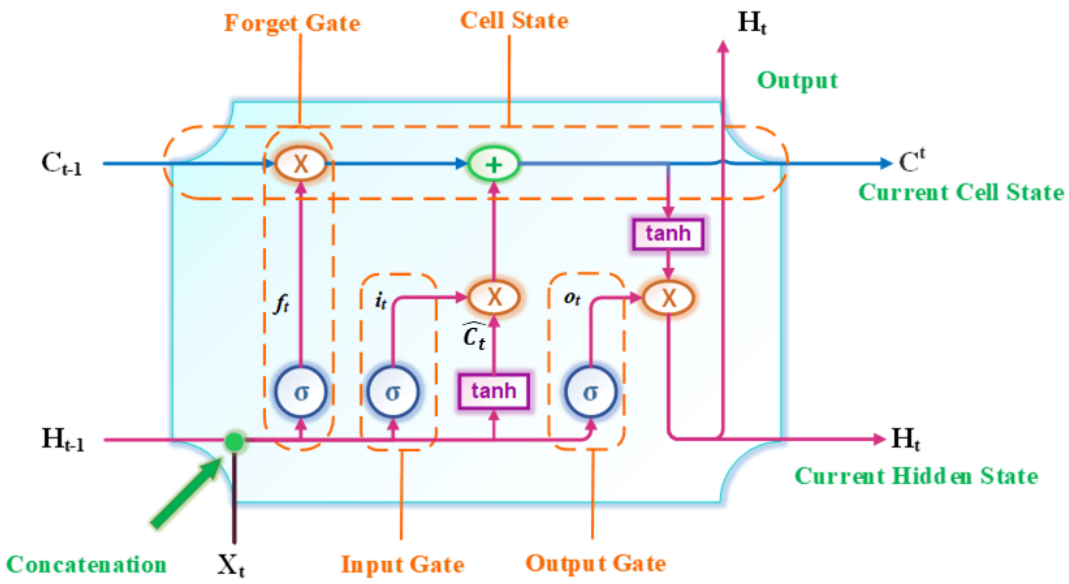


Figure 3.11: Architecture of LSTM.

In this equation, f_t refers to the output of forget gate. The sigmoid function is represented by σ symbol, which produces output between [0, 1] interval where 0 represent the elimination of information and 1 denotes the incorporation of information.W_f is the weight vector for the forget gate which comes to be in dot product with the concatenation of hidden state of the previous step, i.e. H_{t-1} and current input X_t. However, bf refers to the bias vector for forget gate.

After that, the input gate decides which new information are going to be stored in cell state. Input gate works through the following equations:

$$i_t = \sigma(W_i.[H_{t-1}, X_t] + b_i) \tag{3.6}$$

$$\hat{C}_t = tanh(W_c.[H_{t-1}, X_t] + b_c) \tag{3.7}$$

In equation 3.6, i_t refers to the output of the input gate. W_i and b_i represent the weight vector and bias value respectively. In equation 3.7, a *tanh* function is applied to create $\hat{C}_t$ that represents the vector of new information that could be added to the state. A The *tanh* activation is used to help control values that flow across the network by maintaining fixed interval i.e. [-1, 1] of values. W_c is the weight vector which comes to be in dot product with the concatenation of the hidden state of the previous step i.e. H_{t-1} and current input X_t. However, b_c refers to the bias vector for this operation.

Then, the cell state for the current step is calculated using the following equation:

$$C_t = f_t * C_{t-1} + i_t * \hat{C}_t \tag{3.8}$$

Here, f_t and i_t represent the output of the forget gate and input gate, respectively. C_{t-1} is the cell state of the previous step, and C_t contains the information to be added to the cell.

Finally, the output gate decides the outcome of the hidden state of current LSTM cell by the equations provided below:

$$o_t = \sigma(W_o.[H_{t-1}, X_t] + b_o) \tag{3.9}$$

$$H_t = o_t * tanh(C_t) \tag{3.10}$$

Here, in equation 3.9, the sigmoid function is applied to get o_t that refers to the result of the output gate. W_o is the weight vector which comes to be in dot product with the concatenation of hidden state of previous step i.e. ht-1 and current input X_t. However, b_o refers to the bias vector for this operation. In equation 3.10, the hidden state, H_t is measured through the multiplication of output gate result and $tanh$ of cell state C_t. Thus, an LSTM cell provides two outputs: hidden state (H_t) and cell state (C_t) which are passed to next hidden unit.

So, in this way, LSTM processes data and flows the information as it propagates forward.

In this research work, the extracted spatial features of sequential frames of videos are fed to an LSTM layer for analyzing them in order. The number of hidden units used here is 128.

3.6 Classification

For classification, the Softmax activation function is used in the output layer. However, it is a good practice to use a Dense layer before the final output layer. A dense layer works like a layer of neurons where each neuron takes input from all neurons of the previous layer in the network. This layer comprised of a weight matrix, a bias vector and an activation function. This layer is also called a fully

connected layer which does a linear operation of the inputs and weights in the forward propagation, which is represented in the following equation:

$$Z = f(W.X + b) \tag{3.11}$$

In equation 3.11, Z denotes the output of dense layer and f represents an activation function which is applied on the bias vector, b along with the dot product of weight matrix, W and input, X. In backward propagation, the weights are fine-tuned through measuring the error rate calculated by the distance of predicted output and actual output. The principle of the dense layer is depicted in figure 3.12. In this research work, a dense layer of 128 neurons is added after the LSTM layer and is followed by the output layer. Dropout is used in this layer with 0.2 rate for reducing the overfitting problem.

In the output layer, the softmax function yields the probability distribution of

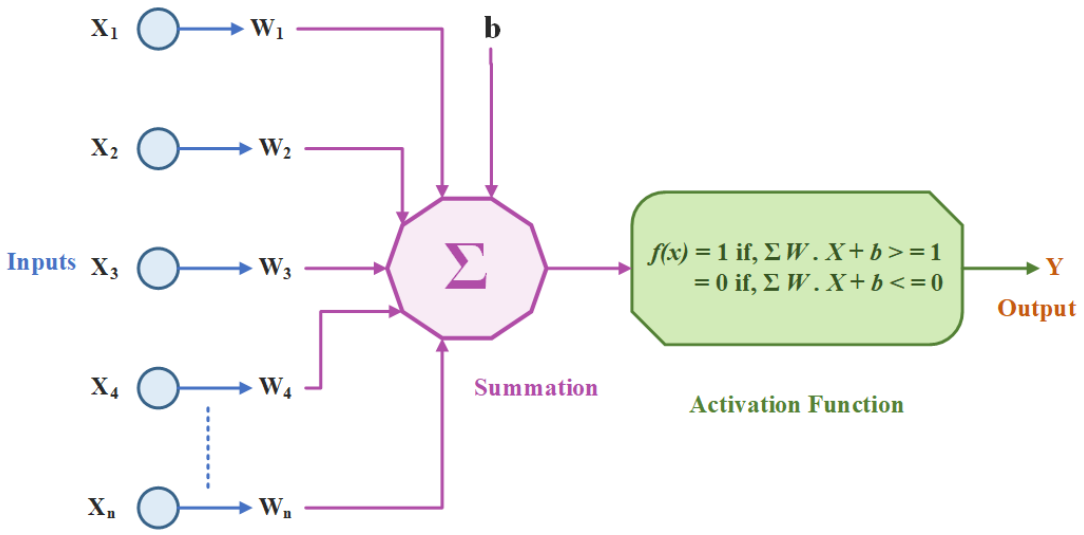


Figure 3.12: Overview of Dense Layer.

the five classes based on the output of the previous layer in the network. This function produces output ranging between 0 to 1 and is produced in such a way that all output values sum to 1. This function is widely used for multiclass classification in deep learning. The number of neurons in the output layer is equal to the number of classes in the dataset.

For getting the probability distribution Softmax function uses equation 3.12, which is given below:

$$Softmax(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{3.12}$$

In this equation, $\vec{z}$ refers to the input vector of this function, all the z_i values represent the elements of the input vector, and K is the number of classes.
The whole network built for this research work is comprised of CNN layers, LSTM layer, Dense layer and output layer. This network is depicted in figure 3.13.

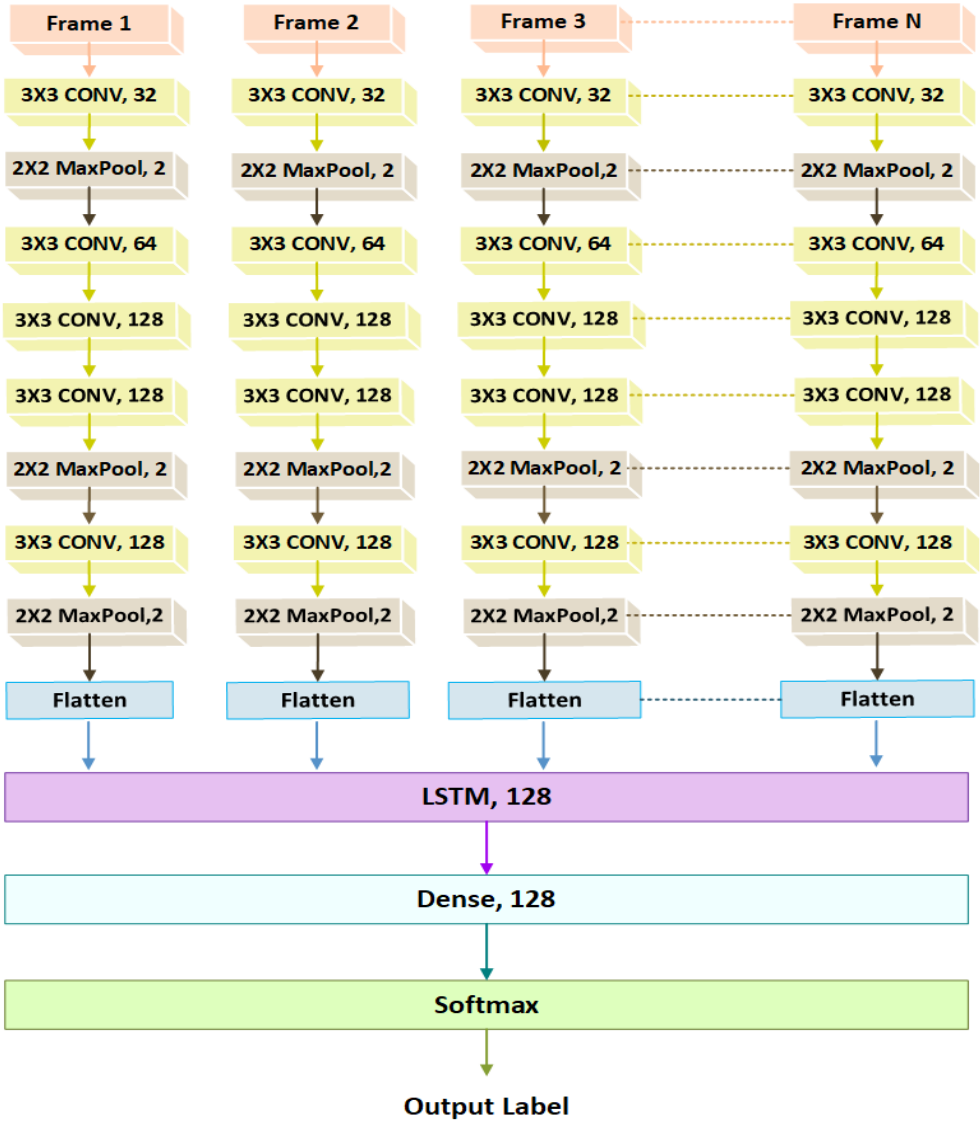


Figure 3.13: Overview of Network Architecture.

3.7 Transfer Learning Approach with Pretrained Model

Deep neural networks require a huge amount of training data for learning the features. In this context, using a pretrained network is quite an efficient approach

for the case of working with a relatively small dataset in deep learning-based networks which is called Transfer Learning approach. The transfer learning approach is accustomed to making proper utilization of models trained on one problem as a preface on a related issue. Moreover, transfer learning is a versatile technique which invokes the use of pre-trained models as a feature extraction method that can be incorporated into completely new models. Transfer learning can extensively reduce the training time required for deep learning model and produce low generalization error.

A pretrained network is one that has been already trained with some other dataset with a different number of classes. There are several numbers of top-notch pretrained networks that were established for computer vision tasks. Likewise, Resnet, Inception, Xception, VGG are pretrained Convolutional Neural Networks which are trained on one of the benchmark datasets, i.e. ImageNet dataset and showed impressive performance on ImageNet classification challenge.

In this research work, Resnet, Inception, Xception, and two variations of the VGG network have been used for spatial feature extraction and compared their performance briefly.

3.7.1 Combination of Resnet152 v2 Network and Recurrent Neural Network

ResNet, an abbreviation for Residual Network, is a type of neural network that was first introduced in 2015. Resnets are comprised of Residual Blocks that use a technique called 'skip connection' to avoid the issue of vanishing gradients. It's termed as the foundation of residual blocks. Fig 3.14 illustrates a residual block.

The skip connection is essentially an identity mapping in which the input from a prior layer is inserted straightway to the output of some other layer. The principle behind incorporating more layers in deep learning is that these layers can gradually learn more advanced features. However, it has been discovered that the standard CNN model has a maximum depth threshold. This implies that as more layers are added to a network, its output diminishes due to the vanishing gradient issue. The skip connections resolve the issue of vanishing gradients by enabling the gradient to pass via an alternate shortcut route. These relations

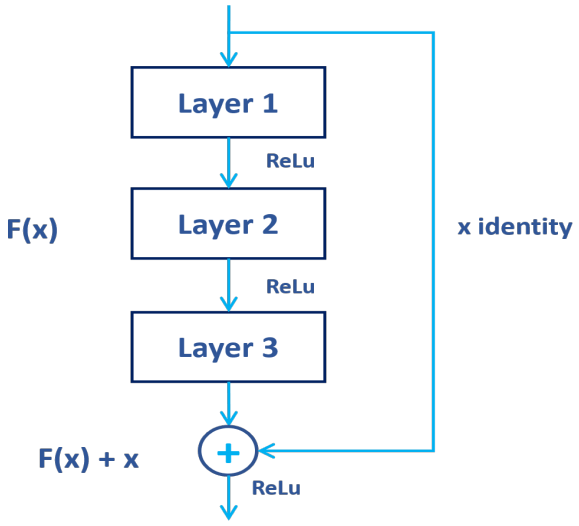


Figure 3.14: Residual Block with Skip Connection.

often aid the model by enabling it to learn the identity functions, ensuring that the higher layer performs at least as well as the lower layer, if not worse.
In this work, we have implemented a recent and advanced variation of ResNet, ResNet152 v2 that consists of 152 layers which uses batch normalization after activation of each weight layer. Later this network is integrated with LSTM and dense layer. The ResNet152 v2 network extracts the spatial features and LSTM grabs the temporal connectivity. The performance of this network has been described briefly in this work.

3.7.2 Combination of Inception v3 Network and Recurrent Neural Network

The third variant of Inception network developed by Google is Inception v3 which is a convolutional neural network that aids in computer vision tasks. It was first used as part of the ImageNet Recognition Challenge and this network showed splendid performance in this challenge. Via dimensionality reduction with stacked 1×1 convolutions, Inception Modules allow for more efficient computation in deeper networks. It is the first variation of Google's Inception Network, where batch normalization was used. Moreover, in this network $n \times n$ convolutions are factorised into asymmetric convolutions: $1 \times n$ and $n \times 1$ convolutions. In this context, 5×5 convolution is factorized to two 3×3 convolution operations.

Furthermore, 3×3 convolutions are used in the space of 7×7 to enhance the performance. This also reduces computational time and thus boosts computational speed because a 7×7 convolution is more expensive than 3×3 convolution. This dimensionality reduction technique immensely helps to get better performance with reduced computational cost.

In this work, we have implemented Inception v3 of 42 layers as spatial feature extractor and integrated it with LSTM and dense layer. Here, LSTM works as temporal feature extractor. The performance of this network has been described briefly in this work.

3.7.3 Combination of Xception Network and Recurrent Neural Network

Being an extension of Inception network, Xception uses depthwise separable convolutions in place of the regular Inception modules. Xception network outperforms Inception network on Imagenet Dataset though they hold nearly same number of parameters. Depthwise separable convolutions are computationally more robust alternatives to classical convolutions. It is divided into two steps: pointwise convolution and depthwise convolution. However, in Xception network, the pointwise convolution is followed by depthwise convolution. The architecture of Xception consists of 3 core parts: entry flow, middle flow, and exit flow. The data initially passes via the entrance flow, then eight times via the middle flow, and ultimately via the exit flow. In this context, the whole architecture is comprised of depthwise separable convolution blocks and maxpooling both of which are connected with shortcuts in the same way like ResNet implementations. Here, batch normalization is applied on both convolution and separable convolution layers. Moreover, by depthwise Separable Convolutions, the number of operations is reduced by a factor proportional to 1/N. So, the architecture engineering of Xception network can be credited to its superior performance with almost the same number of parameters.

In this work, we have implemented Xception network of 71 layers as spatial feature extractor and integrated it with LSTM and dense layer. Here, LSTM works as

temporal feature extractor. The performance of this network has been described briefly in this work.

3.7.4 Combination of VGG16 Network and Recurrent Neural Network

VGG16 is a pretrained Convolutional Neural Network that contains 16 weighted layers comprised of convolution layers and fully connected layers followed by output layer for classification purpose. Moreover, there are five convolutional blocks, and after each block, there is a max-pool layer in this network. The input size used in this network is 224x224. The model attained 92.7% top-5 test accuracy in ImageNet, a dataset comprised of over 14 million images of 1000 classes. The overall architecture of VGG16 network is shown in figure 3.15.

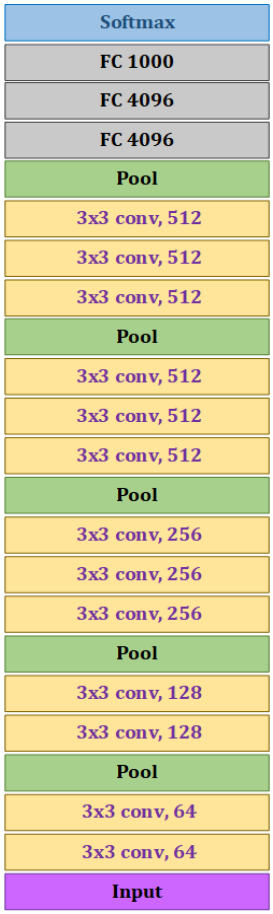


Figure 3.15: Complete architecture of VGG16 Network

In this study, we've utilized the 13 weighted layers, which were designed for feature extraction from the input frames of videos, and the input size is fixed

to 128x128. Later, the part of VGG16 network used for feature extraction is flattened and integrated with an LSTM layer of 128 hidden units. A dense layer with 128 neurons is followed by the output layer is used to final classification of output label. The integration of pretrained VGG16 network and custom layers is depicted in figure 3.16.

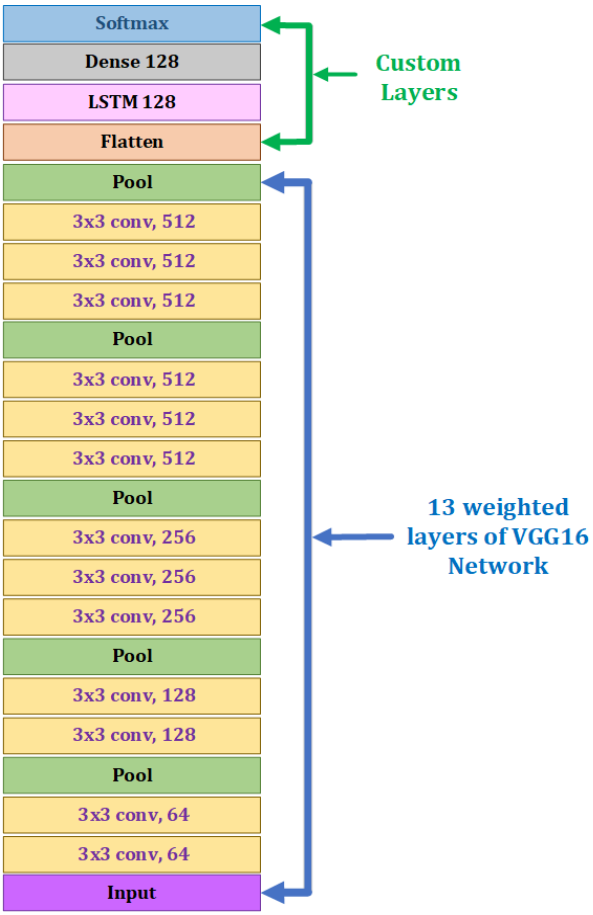


Figure 3.16: Combined Architecture Based on VGG16 Network.

3.7.5 Combination of VGG19 Network and Recurrent Neural Network

VGG19 is another variation of the VGG network, which is almost similar to VGG16 in architecture. Likewise, there are 5 convolutional blocks, and after each block, there is a max-pool layer in this network. However, each of last two convolutional blocks holds one extra convolutional layer than VGG16 network. The input size used in this network is 224x224. However, the number of weighted layers is 19 in case of VGG19, which means it has more parameters than VGG16.

The model attained 92% top-5 test accuracy in ImageNet, a dataset comprised of over 14 million images of 1000 classes. Figure 3.17 shows the complete architecture of VGG19.

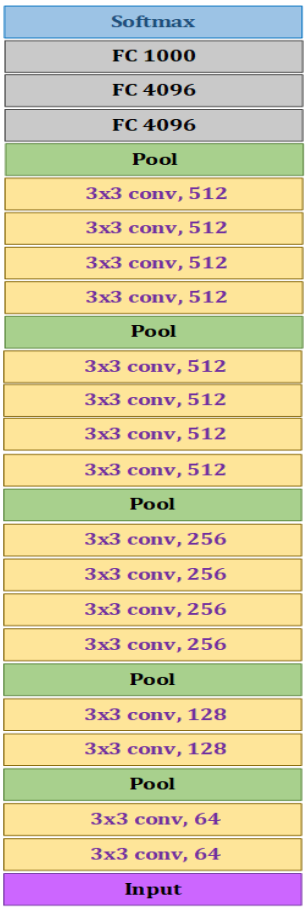


Figure 3.17: Complete Architecture of VGG19 Network.

In this research work, initially, 16 weighted layers of VGG19 is exploited to get spatial features from the sequential images of training videos. Later the adopted part of VGG19 architecture is flattened and consolidated with some custom layers: LSTM layer of 128 hidden units and Dense layer of 128 neurons. Finally, the softmax output layer is used for final classification of output label. The integration of pretrained VGG16 network and custom layers is depicted in figure 3.18.

3.8 Fine Tuning Approach with VGG Models

A widely utilized tactic for model reuse is fine-tuning that aims to make the pretrained model more relevant for the new dataset. In general, fine-tuning can

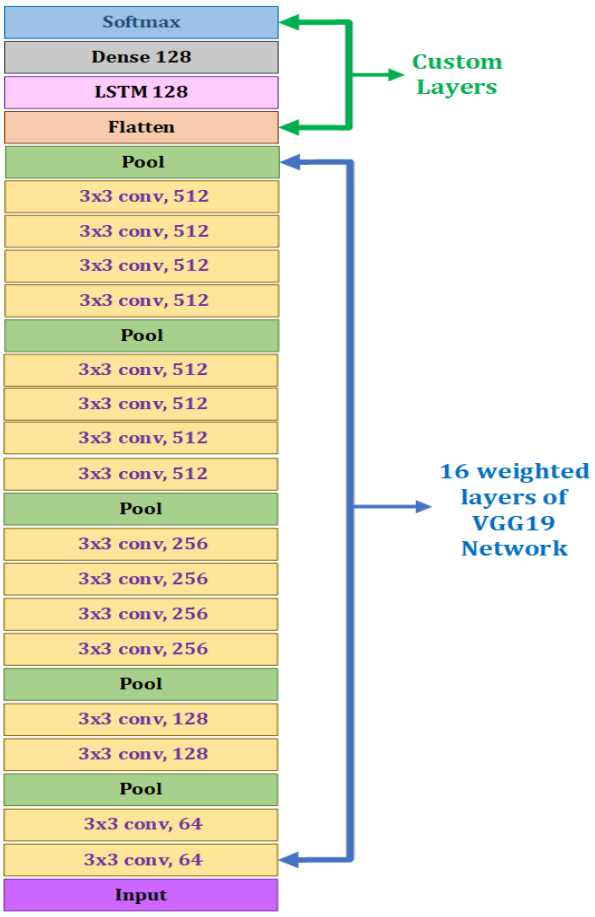


Figure 3.18: Combined Architecture Based on VGG19 Network.

be used if the dataset is not significantly different in context from the dataset on which the pre-trained model was trained. In its early layers, a pre-trained network on a broad and complex dataset extracts universal features such as curves and edges, which are important and useful to most classification task.

So, fine-tuning can be performed through unfreezing some of the top layers of a frozen model, adding some custom layers over that model, and then jointly training both the unfreezed and custom layers. Unfreezing a layer refers that some modifications are performed on the certain layers and update the weights for these layers during training on the specific dataset. This results in better adjustment of the model on new dataset.

In this work, we have performed fine tuning of the two best performing pretrained models, VGG16 and VGG19, integrated with some custom layers of LSTM and dense layer. As stated earlier, VGG16 and VGG19 models are convolutional neural networks consist of 16 and 19 layers. For better adaptation of these model on our dataset, we have unfreezed some of the top layers of these models and after

that integrated with LSTM. Various network configurations experimented with by the fine-tuning task are illustrated in table 3.1. Performance comparison of the inspected various configurations of fine-tuned VGG16 and VGG19 are briefly illustrated in this paper.

Model	**VGG Network Version**	**Trainable Layers of VGG (Excluding fc layers)**	**Hidden Units in LSTM Layer**	**Neurons in Dense Layer (0.2 dropout)**	**Total Trainable Parameters**
Model-1	VGG-16	None	128	128	4,277,515
Model-2	VGG-16	All			18,992,203
Model-3	VGG-16	Last 4 layers			11,356,939
Model-4	VGG-16	Last 8 layers			17,256,715
Model-5	VGG-16	Last 12 layers			18,732,043
Model-6	VGG-19	None			4,277,515
Model-7	VGG-19	All			24,301,899
Model-8	VGG-19	Last 4 layers			11,356,939
Model-9	VGG-19	Last 8 layers			18,436,363
Model-10	VGG-19	Last 12 layers			22,566,411

Table 3.1: Various Fine Tuning Network Configurations.

3.9 Implementation

The implementation requirements of traditional Bangladeshi sports video classification are demonstrated here.

3.9.1 Hardware Requirements

The overall process of this work propagates the following hardwares from input to output.

- Tesla K80 GPU
- GPU RAM 12 GB
- Single core hyper threaded Xeon Processor CPU

- Physical Memory 68 GB

3.9.2 Software Requirements

Necessary software requirements for the classification of traditional Bangladeshi sports video are:

- Tensorflow == 2.4.1
- Keras == 2.4.3
- Python == 3.7.10
- Numpy == 1.19.5
- Markdown ==2.6.10
- Jupyter Notebook
- Operating System : ubuntu 18.04, windows 10

However, each of these software aren't requisite at the same time. These software were used to incorporate various parts of our framework. In this regard, to evaluate our method, all that is needed is an IDE on which the system can be run.

3.9.3 Deep Learning Optimizer

The optimizer defines exactly how the loss is being used to update parameters. The basic trick in deep learning is to use the loss score as an input signal to change the weight values slightly in the direction of lowering the current example's loss score. The optimizer is responsible for this adjustment. Various types of optimizers are: Stochastic Gradient Descent, Adam, RMSProp, Adagrad, etc. In this work, Adam is used as optimizer during training of the models as Adam shows better convergence than others [22].

3.9.4 Learning Rate

The learning rate decides the step size when progressing towards to the minimum of the loss value. It is difficult to choose the learning rate since a too high learning

rate will make the model converge to a sub-optimal solution very rapidly, whereas a too low learning rate will make the process stick together. In this work, learning rate of 0.0001 worked as optimal for the classification of TBSV dataset.

3.9.5 Loss function

The objective function is the difference between the actual output and the model's predicted output of a sample data. In order to keep the algorithm working optimally, this measurement is used as a feedback.As classification of TBSV is a multiclass classification task, categorical cross-entropy has been used as loss function. Categorical cross-entropy is a type of loss function where the target values for multiclass classification are represented in a one-hot vector. The multinomial probability distribution provided by the Softmax activation function in the output layer is used by the categorical cross-entropy loss function for measuring the prediction error/loss of the model. This function computes the loss in the following manner:

$$CCL = -logp(k) \tag{3.13}$$

In the above equation, CCL refers to the categorical cross-entropy loss, and $p(k)$ denotes the probabilistic value of the class k that is fired-up in the one-hot vector.

3.9.6 Batch Size

The batch size hyperparameter specifies the amount of samples to process before modifying the model's internal parameters. One or more batches may be generated from a training dataset. In this work, we have considered batch size of 64.

3.9.7 Epoch

The number of epochs specifies how many times the learning algorithm can pass the training dataset. In our work, the number epoch is considered 65 for training scratch model and 30 for pretrained model.

3.10 Conclusion

A methodology for Traditional Bangladeshi Sports Video Classification has been discussed in this chapter. The proposed method has been experimented with a scratch deep neural network combining CNN with LSTM. Moreover, some pretrained models have also been explored. Furthermore, fine tuning has been applied on the best performing pretrained models integrated with LSTM. In all the experimented networks, CNN acts as the spatial features extractor and LSTM works as temporal feature extractor for the input frames. The upcoming chapter will dive into the experimental results of methodology.

Chapter 4

Results and Discussions

4.1 Introduction

In the previous chapter, a detailed description of the methodology for classifying Traditional Bangladeshi Sports Video was discussed. This chapter will give a closer look at the results of proposed method.

However, unfortunately, there is no standard dataset for traditional Bangladeshi sports video. Hence, for this work, Bangladeshi sports video data has been collected from Youtube, under which the integrity of the proposed method can be examined.

4.2 Experimental Dataset Description

4.2.1 Traditional Bangladeshi Sports Video (TBSV) Dataset

As mentioned earlier, a few attempts can be found to classify traditional Bangladeshi sports videos. However, unfortunately, there is no standard dataset for this task. Therefore, one of the vital contributions of this work is to develop the novel TBSV dataset. This dataset consists of 500 traditional Bangladeshi sports videos belonging to five classes: Kabaddi, Boli Khela, Kho Kho, Lathi Khela, and Nouka Baich, collected from Youtube. The details of this dataset are illustrated in table 4.1. Samples of some input frames of video of each class are showen in figure 4.1.

For experimenting with the performance of the models, the traditional Bangladeshi sports video dataset has been split into train, test and validation data. The split ratio is 70 : 15 : 15 (i.e. 70% of the video data is selected for training, 15%

Traditional Bangladeshi Sports Video (TBSV) Dataset	
Total Classes	5
Number of videos per class	100
Frames per second	30
Video Length	5 s
Resolution	720 x 1280

Table 4.1: TBSV Dataset Details.

for validating and 15% for testing the models randomly). However, for the sake of consistency, this randomization is kept fixed throughout the experiment.

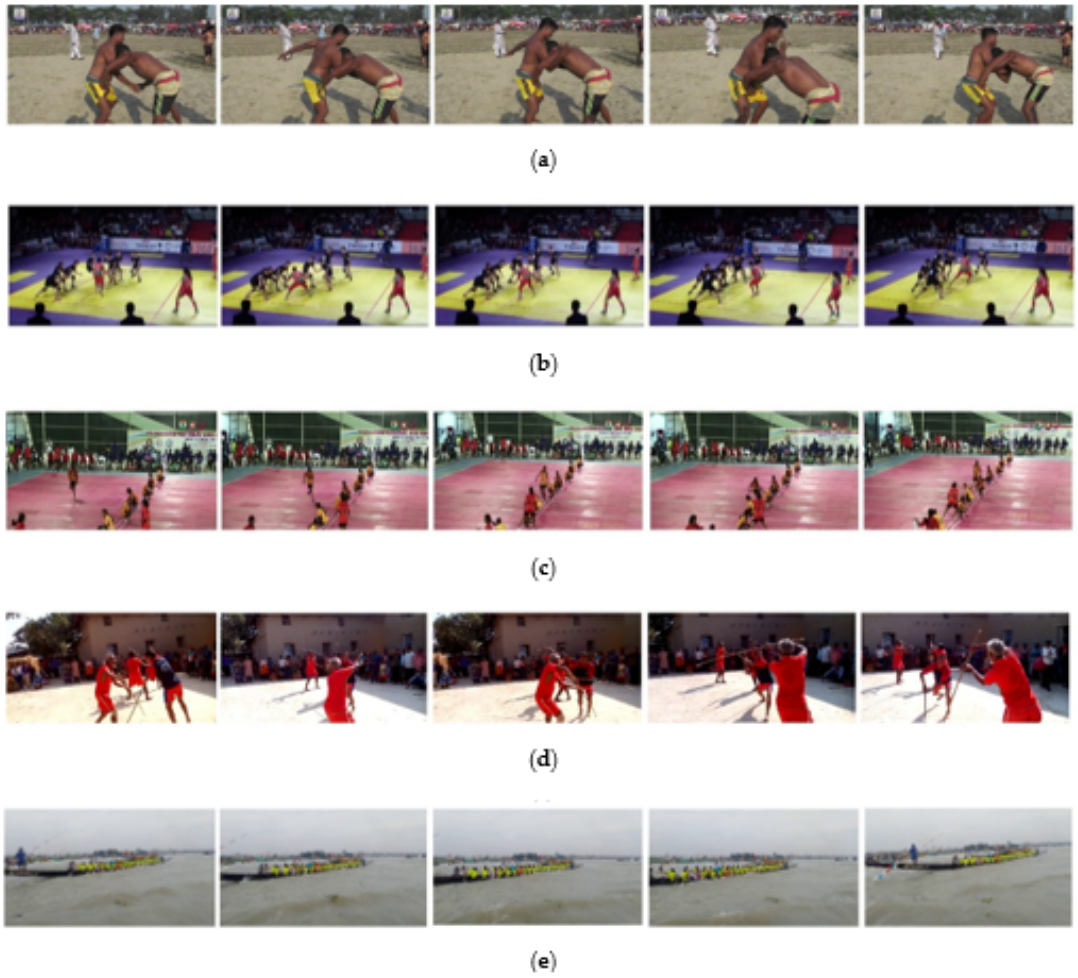


Figure 4.1: Sequential frames of (a) Boli Khela; (b) Kabaddi; (c) Kho Kho; (d) Lathi Khela; (e) Nouka Baich.

4.2.2 KTH Dataset

KTH [23] is among the most extensive datasets, widely used in human action recognition tasks. It includes six types of human actions: walking, running, boxing, jogging, hand-waving, and hand-clapping. This dataset contains 600 videos, 100 videos per class with a resolution of 160 x 120. The videos are an average of 4 s in length and have a frame rate of 25 fps.

4.2.3 UCF-11 Dataset

UCF-11 [24] is one of the most remarkable benchmark datasets of action recognition. It's also named as UCF YouTube Action dataset. This dataset includes 11 action classes: biking, basketball shooting, volleyball spiking, diving, horse back riding, soccer juggling, swinging, golf swinging, tennis swinging, trampoline jumping, and walking with a dog. It contains 1600 videos with the frame rate of 29.97 fps.

4.2.4 UCF-101 Dataset

UCF-101 [25] is the greatest and one of the benchmark datasets that include 101 action classes of five categories: Sports, Body-Motion Only, Human-Object Interaction, Playing Musical Instruments, and Human-Human Interaction. It contains 13,320 videos of 320 x 240 resolution, and the maximum number of frames is 150 per video. In this work, due to GPU memory constraint, we have conducted our experiment over 66 classes belonging to two categories of this dataset: body motion only and sports.

4.2.5 UCF Sports Dataset

UCF Sports [26, 27] is one of the leading datasets in the application of action recognition and localization tasks. It includes 150 sequences of 10 classes: Walking, Running, Lifting, Diving, Kicking, Golf-Swing, Swing-Bench, Riding-Horse, Swing-Side, Skate-Boarding, collected from YouTube. The resolution of the videos is 720 x 480, with a variable number of videos in each class. The frame

rate of videos is 10 fps. However, the length of the videos is not fixed, whereas the min video length is 2.20 s, and the max video length is 14.4 s.

4.3 Impact Analysis

Traditional Bangladeshi sports video classification has the potential to have a significant effect on the social and environmental spheres, and also on ethics. These impacts are illustrated briefly.

4.3.1 Social and Environmental Impact

The history of Bangladesh encompasses its cultural diversity and literary heritage. Furthermore, significant parts of its profoundly ingrained heritage are reflected in its traditional sports activities. Therefore, sport in Bangladesh is a popular medium of amusement as well as an integral part of Bangladeshi culture. However, due to the lack of nurture and retention, these traditional sports activities are losing their dignity day by day. Classification of traditional Bangladeshi sports videos perhaps can revive the glory and pride of these sports. By this work, these tribal sports can be reintroduced the people of our country as well as all over the world.

4.3.2 Ethical Impact

The classification of traditional Bangladeshi sports videos can arise some potential ethical issues if anyone tries to conduct further research on this using personal repositories without copyright.

4.4 Evaluation Metrics

In order to assess the integrity of the mathematical or machine learning models, evaluation metrics are widely used. For any framework, it is quite essential to evaluate the machine learning models or algorithms. To test a model, there are various types of evaluation metrics available. Some of significant evaluation metrics are:

- Accuracy
- Precision
- Recall
- F1 Score
- Confusion Matrix

These metrics impact how we determine the significance of various features in the result of algorithms/models and our final choice of which algorithm / model-version to use. Moreover, it is very crucial to use multiple evaluation metrics in order to access a model's efficiency from several perspectives which helps to prove the worthiness of a model to a great extent. However, these metrics are illustrated briefly in this section.

4.4.1 Confusion Matrix

A confusion matrix is the N x N matrix used to test the effectiveness of a classification model, where N is the number of classes in the dataset. As there are five classes in traditional Bangladeshi sports video dataset: Boli Khela, Kabaddi, Kho Kho, Lathi Khela, Nouka Baich, in our case of multiclass classification problem, the confusion matrix will be a 5 x 5 matrix containing a total of 25 values. Each value in the confusion matrix indicates the number of predictions made by the model through which it can be determined whether the classes are correctly classified or not. There are four kinds of assessing values in a confusion matrix, which are illustrated below:

- **True Positive:** It denotes the number of predictions in which the positive class is predicted as positive by the model.
- **False Positive:** It represents the number of predictions where the negative class is misclassified as positive by the model.
- **True Negative:** It denotes the number of predictions where the negative class is accurately predicted as negative by the model.

- **False Negative:** It represents the number of predictions where the positive class is misclassified as negative by the model.

4.4.2 Accuracy

One way to find out how much the algorithm/model recognizes a data point correctly is termed as the accuracy of a model/algorithm. In fact, accuracy denotes the number of accurately estimated data points from all of the data points. Accuracy is one of the most vital evaluation metrics. It's pretty easy to grasp and easily suited for both a binary and a multi-class classification problem. Accuracy is measured through the following equation:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{4.1}$$

4.4.3 Precision

Precision is also termed as a positive predictive value. It denotes the ratio of accurate positive predictions to the overall predicted positives. Precision works with the following equation:

$$Precision = \frac{TP}{TP + FP} \tag{4.2}$$

Precision turns 1 only when $TP = TP + FP$, which means FP is zero. The more the value of FP, the less will be the value of precision.

4.4.4 Recall

Recall is also named as positive sensitivity or true positive rate. It represents the ratio of accurate positive predictions to the total positive labels. Recall works with the following equation:

$$Recall = \frac{TP}{TP + FN} \tag{4.3}$$

4.4.5 F1 Score

It is also termed as the F Score or the F Measure. F1-score is a metric that takes both precision and recall into consideration and is defined as follows:

$$F1\ Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{4.4}$$

F1 Score is the combination of precision and recall that makes it a better evaluation metric than accuracy. If we need to strike a balance between precision and recall and if there is an unequal class distribution, the F1 Score could be a more preferable metric to use.

4.5 Evaluation of Performance

For experimenting with the performance of the models, the TBSV dataset has been split into train, test and validation data. The split ratio is 70:15:15 (i.e. 70% of the video data is selected for training, 15% for validating and 15% for testing the models randomly). However, for the sake of consistency, this randomization is kept fixed throughout the experiment. Performance evaluation of scratch model and transfer learning based models will be illustrated briefly in this section.

4.5.1 Frame Length Selection

For implementing the scratch model and the fine-tuned pretrained models in classifying the sports videos, a specific number of frames is needed to be picked from each video to commence the task of classification. In this regard, we carried out an experiment using the scratch model to determine the right frame length, with 100 video samples in each class of TBSV dataset. In Figure 4, validation and test accuracy for frame length 10–25 are shown. From this figure, it can be observed that the model exhibits the best result, i.e., near about 93% validation accuracy and 91% test accuracy for the case of frame length 20. Hence, the frame length of 20 was deemed for further analysis and exploration in this ground.

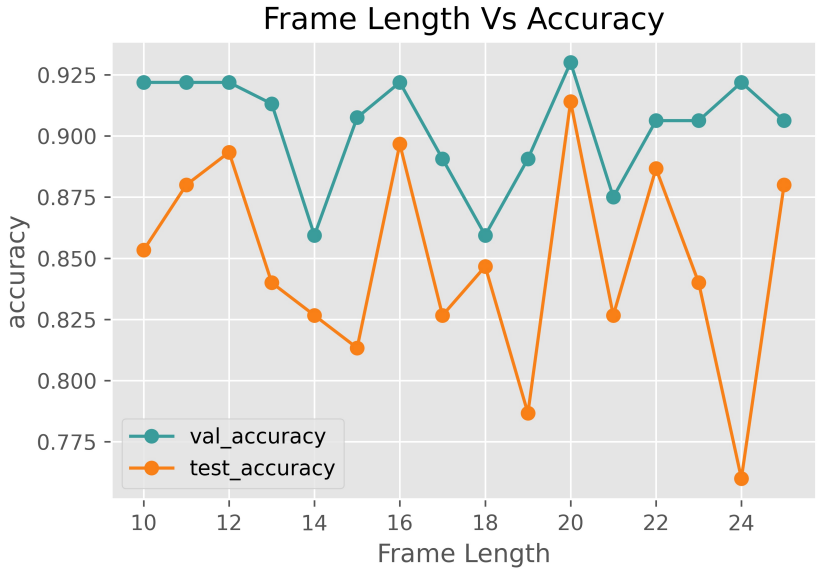


Figure 4.2: Frame Length vs. Accuracy Curve.

4.5.2 Effect of Normalization on Performance

As stated previously, we have used normalization for faster convergence of the model during training. The performance comparison of the scratch model on normalized and unnormalized data are represented in table 4.2. The table shows that, with normalized training data, better performance has been achieved in less epoch relative to the unnormalized training data.

	Training Accuracy	**Validation Accuracy**	**Test Accuracy**	**Total Epochs**
Normalized Data	99.30%	93%	91%	65
Unnormalized Data	96%	90%	88%	70

Table 4.2: Effect of Normalization on Performance.

4.5.3 Performance Evaluation of Scratch Model

As stated earlier, the scratch model built for classifying 5 types of traditional Bangladeshi sports video, is comprised of 5 convolutional layers, 3 maxpool layers, a flatten layer, an LSTM layer and a dense layer followed by output layer. So, in total, the scratch model consists of 11 layers excluding output layer. The model is trained with training data and validated with validation data. This scratch model has been compiled through Adam optimizer with 0.0001 learning

rate and categorical cross-entropy loss function. However, the model is trained till 65 epochs with batch size of 64. The training and validation accuracy and loss curves of scratch model are depicted in figure 4.3.

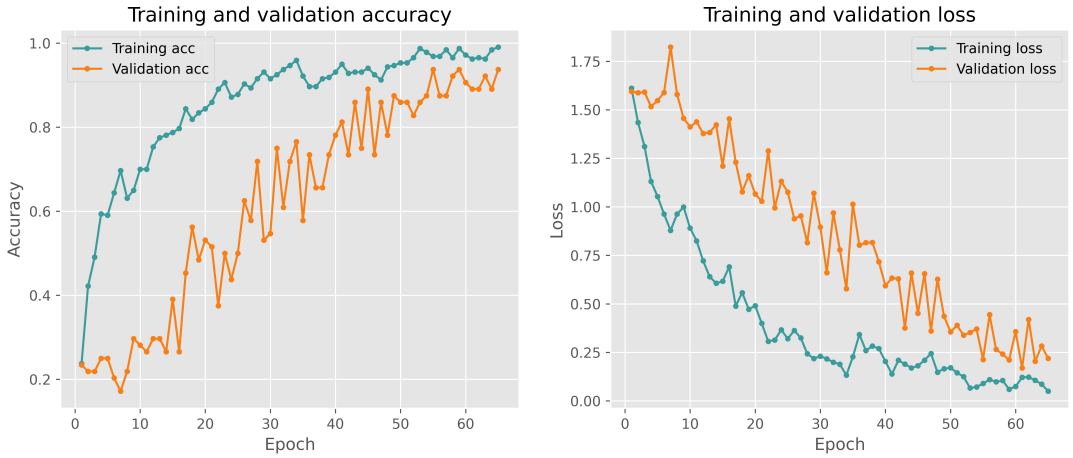


Figure 4.3: Accuracy and Loss Curve for Scratch Model.

This figure illustrates that after the 54th epoch, the training accuracy stops increasing, whereas, at this epoch, we got the maximum validation accuracy, i.e., 93%. This observation refers that at 54th epoch, the model performs the best, and after this epoch, the model ceases learning. Moreover, figure 4.4 depicts the confusion matrix of this model applied to test data from which it can be observed that a tiny amount of misclassification has been found in some classes due to having some semantic similarity with the misclassified classes. Class-wise performance based on different evaluation metrics applied to test data for this model is summarized in table 4.3.

Classes	**Mean Accuracy (%)**	**Precision (%)**	**Sensitivity (%)**	**Specificity (%)**	**F1 score (%)**
Boli Khela	91	93	87	98	90
Kabaddi		94	100	98	97
Kho Kho		93	93	98	93
Lathi Khela		76	87	93	81
Nouka Baich		100	87	100	93

Table 4.3: Class-wise Performance of the Scratch Model.

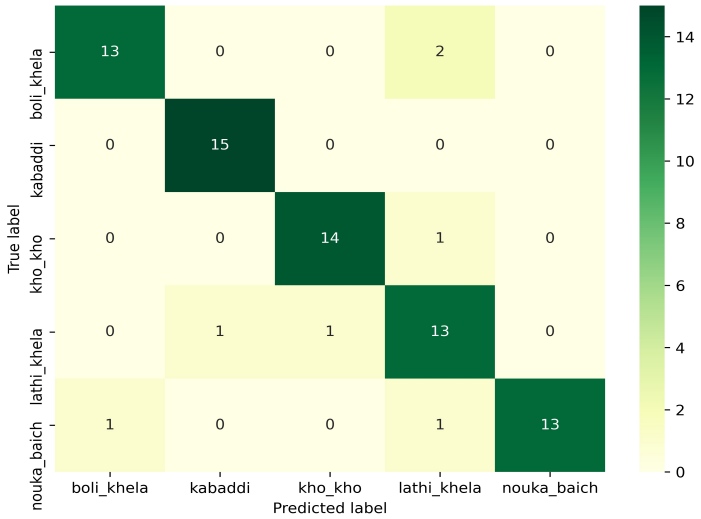


Figure 4.4: Confusion Matrix for Scratch Model.

4.5.4 Performance Evaluation of Pretrained Models

We have conducted an experiment with some of the leading pretrained models: ResNet152 v2, Inception v3, Xception, VGG16, VGG19, combined with LSTM. All of these models are compiled through Adam optimizer with 0.0001 learning rate and categorical cross-entropy loss function. The performance comparison of these models over the TBSV dataset is rendered in table 4.4. From this table, it can be observed that the VGG16 and VGG19 models outperform the others in classifying TBSV classes.

Pretrained Models	**Hidden Units in LSTM Layer**	**Neurons in Dense Layer (0.2 dropout)**	**Test Accuracy (%)**
ResNet152 v2	128	128	85
Inception v3			86
Xception			88
VGG16			90
VGG19			92

Table 4.4: Performance Comparison of Some Pretrained Models on the TBSV Dataset.

4.5.5 Performance Evaluation of Fine-tuned Pretrained Models

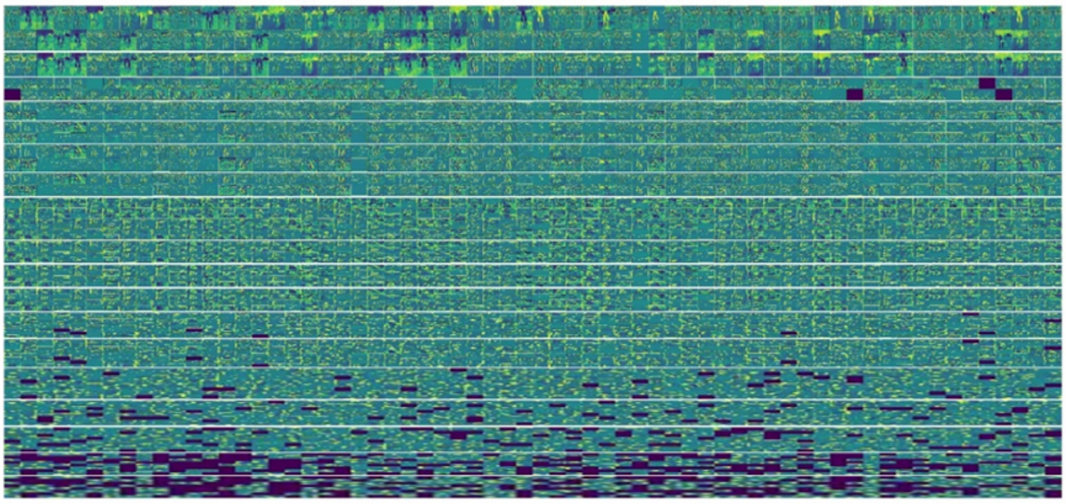
Driven by the impressive performance of VGG16 and VGG19, we dived into we exploration of fine-tuned VGG16 and VGG19 networks incorporated with LSTM. Various network configurations experimented with by the fine-tuning task are illustrated in table 4.5. All the models are compiled using Adam optimizer with 0.0001 learning rate and categorical cross-entropy loss function, trained until 30 epochs with the batch size of 64. However, GPU memory constraint is a crucial issue in training deep learning models.

Model	**VGG Network Version**	**Trainable Layers (Excluding fc layers)**	**Hidden Units in LSTM Layer**	**Neurons in Dense Layer (0.2 dropout)**	**Total Trainable Parameters**	**Training Accuracy (%)**	**Average F1 score (%)**
Model-1	VGG-16	None	128	128	4,277,515	99	90
Model-2	VGG-16	All			18,992,203	97	91
Model-3	VGG-16	Last 4 layers			11,356,939	100	97
Model-4	VGG-16	Last 8 layers			17,256,715	99	98
Model-5	VGG-16	Last 12 layers			18,732,043	97	89
Model-6	VGG-19	None			4,277,515	98	92
Model-7	VGG-19	All			24,301,899	97	88
Model-8	VGG-19	Last 4 layers			11,356,939	99	92
Model-9	**VGG-19**	**Last 8 layers**			**18,436,363**	**100**	**99**
Model-10	VGG-19	Last 12 layers			22,566,411	97	88

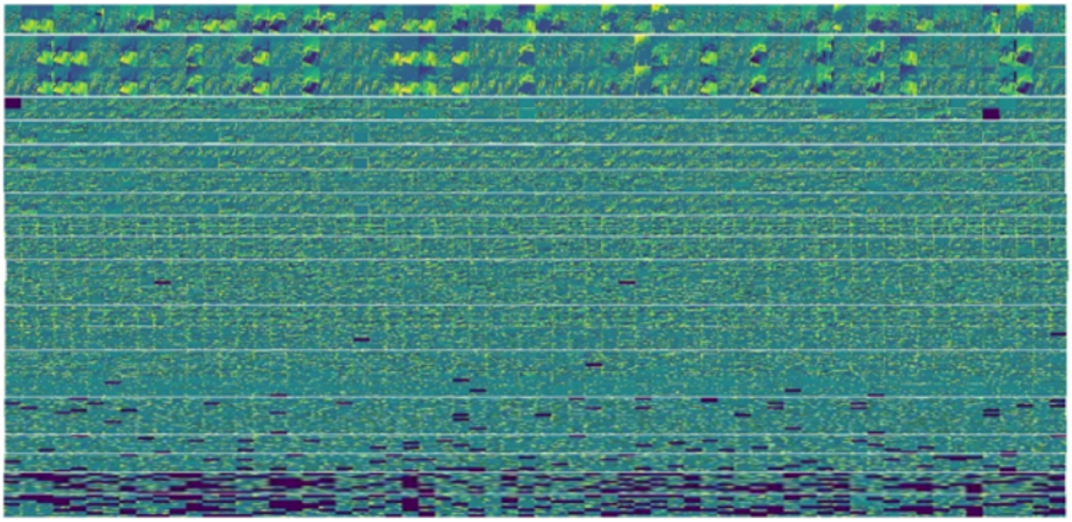
Table 4.5: Network Configuration and F1 Score of Fine-tuned VGG Models.

However, from this table, it can be noticed that the proposed model, i.e., Model-9 with retraining last 8 layers of VGG19 integrated with LSTM and Fully Connected layer, achieved the maximum F1 score of 99%, which is the ultimate evaluation metric considered in this paper. Here, dropout rate of 0.2 is applied to the fully connected layer to reduce overfitting of the model. The rate 0.2 means 20 neurons out of each 100 from this layer are ignored while training. Moreover, our proposed model contains 18,436,363 parameters that is 5,865,536 fewer than the fine tuned all layers of VGG19 as in Model-7 (24,301,899 parameters). The feature maps

produced by the convolutional part of Model-9 from a frame of videos of each class are depicted in figure 4.5.



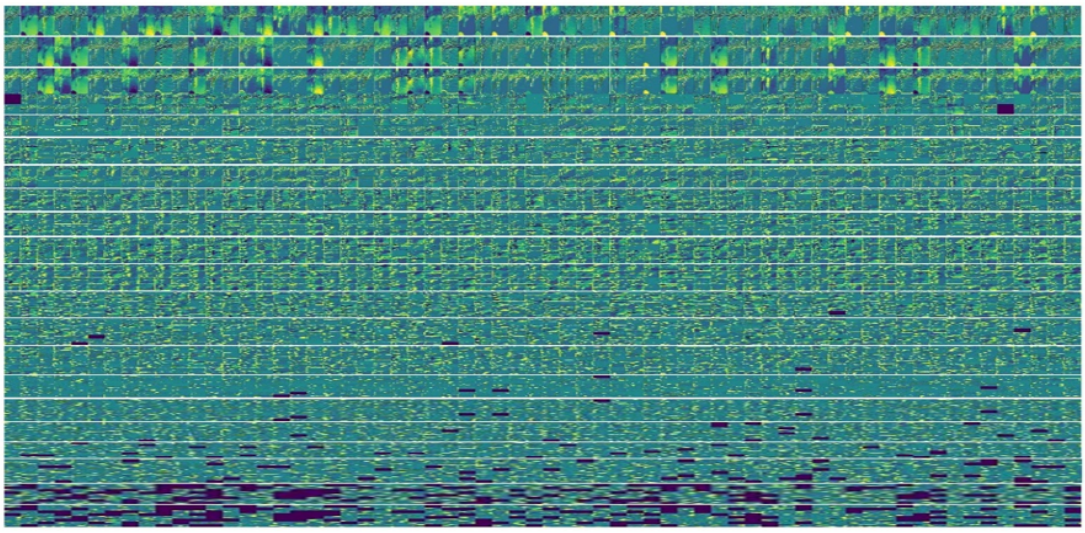
(**a**)



(**b**)

The feature map provides a glimpse of how the input data is dissected into various filters of the model. However, Deep Neural Networks work like a black-box in extracting features from the input. Higher-level layers produce feature maps that are more abstract and opaquer to humans than lower-level layers in the model. They encode higher-level features that are difficult to extract and present in a human-readable way [28]. So, the deep learning models extract higher-level features that are mystical and more immense in amount than the features considered by mere humans.

There are certain key distinguishable features in each sports class. Boli Khela is

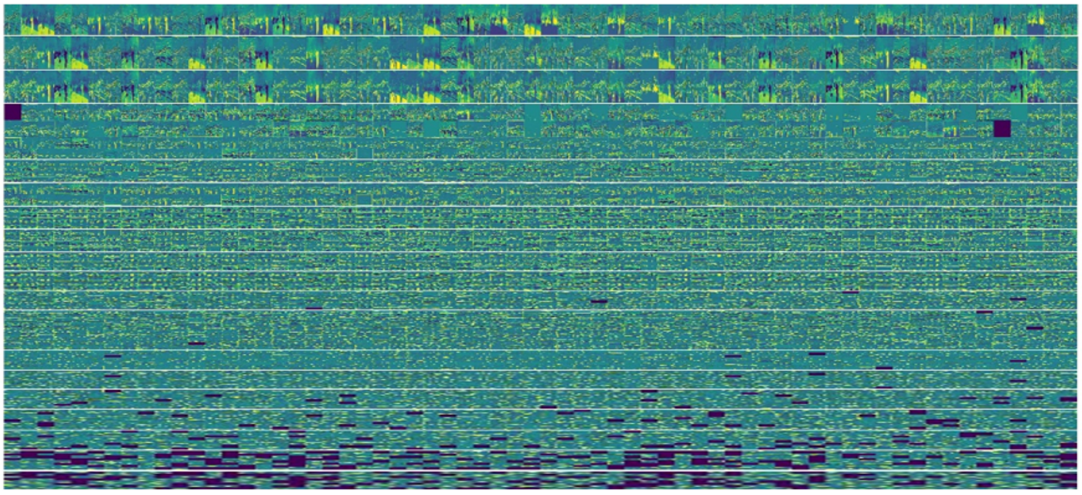


(**c**)

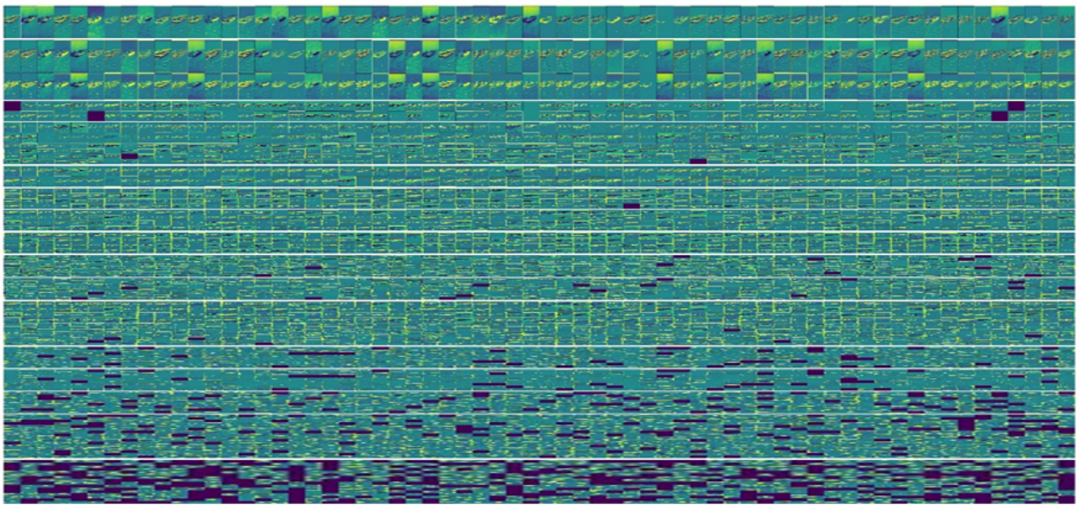
mostly played between two competitors in a sandy wrestling playground. Whereas the Kabaddi playground is divided into two halves occupied by two opposing teams. However, the chasing team in Kho Kho assembles in eight squares in the rectangle play-ground's central lane. Besides, the Lathi Khela is a stick fighting sport played between rival groups. Moreover, the Nouka Baich is a traditional boat rowing sport mostly played during the rainy season on water surfaces. In this paper, the feature maps for each class represented in figure 4.5 are organized according to feature maps of the lower level to higher-level layers of the convolutional part of Model-9. From the feature maps of lower-level layers, it can be observed that our convolutional architecture primarily grabs the surrounding of the playground, key sports equipment, and characteristics of players of each sport as features. As the layers go deeper, the extracted features become more encoded and lack human readability.

The training and validation accuracy, along with the loss curves of the proposed model, is represented in figure 4.6. It can be noticed from this figure, the maximum training accuracy of the model was attained at the 27th epoch, where validation accuracy was the maximum, i.e., 99%.

The confusion matrix of this proposed model, i.e., Model-9, is portrayed in figure 4.7. The figure shows that, due to having some similar semantic characteristics, only a test sample in 'Kabaddi' class was misclassified as 'Boli Khela', which



(d)



(e)

Figure 4.5: The feature Maps Produced by the Convolutional Part of Model-9 of (a) Boli Khela; (b) Kabaddi; (c) Kho Kho; (d) Lathi Khela; (e) Nouka Baich.

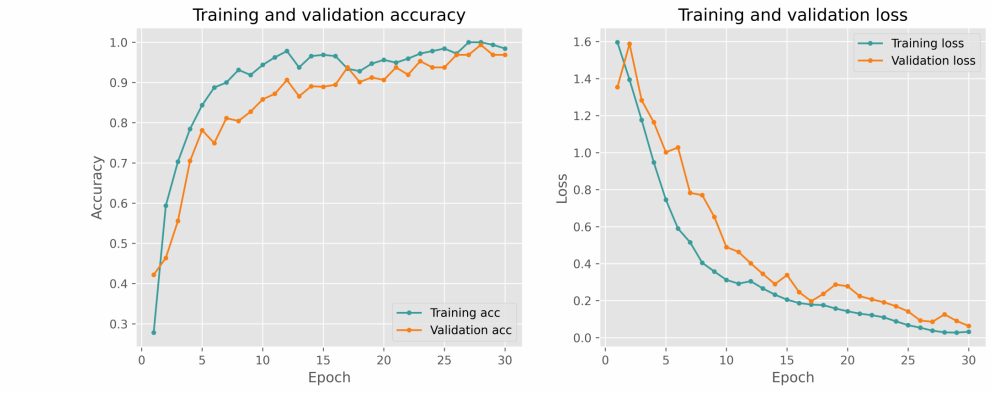


Figure 4.6: Performance of Model-9.

proves this model's efficiency to a great extent. To get more insight into Model-9's performance, class-wise values of evaluation metrics are demystified in table 4.6.

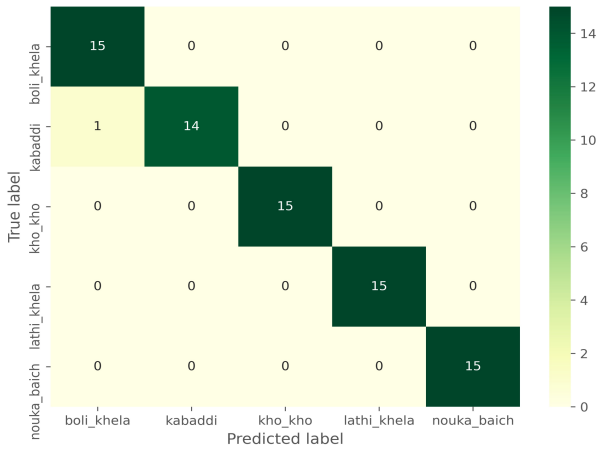


Figure 4.7: Confusion Matrix for Model-9.

Classes	**Mean Accuracy** (%)	**Precision** (%)	**Sensitivity** (%)	**Specificity** (%)	**F1 Score** (%)
Boli Khela	99	94	100	98	97
Kabaddi		100	93	100	97
Kho Kho		100	100	100	100
Lathi Khela		100	100	100	100
Nouka Baich		100	100	100	100

Table 4.6: Class-wise Performance of Model-9.

4.5.6 Performance Evaluation of Proposed Model on KTH Sports, UCF-11, UCF Sports, and UCF-101 Dataset

Driven by the impressive performance of Model 9 over the TBSV dataset, we extended our research work by employing Model 9 in the classification task on the four most prominent datasets: KTH, UCF-11, UCF-101, and UCF Sports. For all the datasets, the split ratio is considered as 80:20 (i.e., 80% of the video data is selected for training, and 20% for testing the models randomly). For consistency,

10 frames per video are considered in the experiment. Table 7 represents the performance comparison of our proposed model over KTH, UCF-11, and UCF Sports datasets with some recent works.

KTH		**UCF-11**		**UCF Sports**	
Method	**Accuracy**	**Method**	**Accuracy**	**Method**	**Accuracy**
Latah [29]	90.34%	Meng et al. [30]	89.7%	Zare et al. [31]	82.4%
Abdelbaky and Aly [20]	93.33%	Yang et al. [21]	91.2%	Jaouedi et al. [18]	89.01%
Xu et al. [32]	95.80%	Ullah et al. [16]	92.84%	Abdelbaky and Aly [20]	90%
Jaouedi et al.[18]	96.30%	Ge et al. [19]	94.12%	Cheng et al. [33]	90%
Our Proposed Model	97%	Our Proposed Model	95.6%	Our Proposed Model	94%

Table 4.7: Performance Comparison with Other Methods.

On the KTH dataset, our model achieved better accuracy, i.e., 97%, than the approach of [18], which was based on a hybrid deep learning.

In the case of the UCF-11 dataset, the proposed model achieved 95.6% accuracy, which surpasses the performance of the method of [19], which was based on attention mechanism-based CNN-LSTM network.

Additionally, after training the proposed model over the UCF Sports dataset, 94% test accuracy has been acquired, proving this model's worthiness relative to the DCNN-LSTM based method of [33] and deep convolution network-based PCANet with encoding method of [20].

Moreover, our proposed model achieved 96% accuracy on UCF-101 dataset and out-performs [18], which also recognizes 66 classes of this dataset by hybrid deep learning model and achieved 89.30% accuracy. Due to limitations of hardware and memory configuration, we have considered only 66 classes of two categories (sports and body-motion only) in UCF-101 dataset.

4.6 Conclusion

The outcomes of the classification for conventional Bangladeshi sports videos are represented in this chapter. The performances of the scratch networks and the pretrained models are also discussed here. Moreover, various network configurations has been established using fine-tuning of the best performing pretrained models. As seen by the analysis, Model-9 appeared to be the proposed model, which is comprised of fine-tuned last 8 layers of VGG19 integrated with LSTM. This model outperforms the other explored models in this ground. Furthermore, the proposed model has been assessed over three other human action recognition datasets: KTH, UCF-11, UCF Sports, and UCF-101. From the comparision of performance with some recent related works on these datasets, it can be observed that the proposed model surpass those works to a great extent. The conclusion to this thesis work is drawn in the next chapter.

Chapter 5

Conclusion

5.1 Conclusion

Sports activities play a vital role in maintaining our well-being. It is necessary to classify the sports videos for various purposes like match summarization, video retrieval, the formation of the new strategy for coaches, etc. Along with these necessities, the classification of Traditional Bangladeshi Sports (TBS) videos can unveil the cultural significance and thus reduce their extinction. However, this task is quite challenging as it requires to capture both spatial and temporal features and demands huge computational cost. However, the main barrier to this task is the scarcity of Traditional Bangladeshi Sports Video dataset. For this reason, we've built a new traditional Bangladeshi sports video dataset containing 500 sports videos of 5 categories: Boli Khela, Kabaddi, Lathi Khela, Kho Kho, Nouka Baich, which is one of the key contributions of this thesis work.

In this research work, a scratch network is developed by incorporating CNN with LSTM. After some preprocessing tasks, the processed frames of video are passed to the CNN network, used as a spatial feature extractor. Afterward, LSTM, a modified Recurrent Neural Network (RNN), is employed to extract the temporal features and reduce the vanishing gradient problem. A dense layer is added later, followed by the output layer, where the softmax activation function is used to classify five individual traditional Bangladeshi sports video classes. This network showed a decent performance of 91% accuracy over test data.

On the other hand, an experiment has been conducted with the transfer learning approach through some pretrained models: Inception v3, ResNet152 v2, Xception, VGG16, and VGG19. These networks are Convolutional Neural Networks (CNN), pretrained on the Imagenet dataset, and showed impressive performance

on that dataset. These architectures are integrated with LSTM for capturing temporal features, which is followed by a dense layer. Finally, the softmax activation function is used to classify five individual traditional Bangladeshi sports video classes. Experimental results over the test data highlight that VGG16 and VGG19 based models outperform other models.

Furthermore, driven by the splendid performance, an experiment was conducted with the transfer learning approach through fine-tuning VGG16 and VGG19 networks. However, the proposed model is comprised of the fine-tuned VGG19 network integrated with LSTM, which shows promising achievement on the TBSV dataset and some benchmark datasets of human activity: KTH, UCF-11, UCF-101, and UCF Sports. Additionally, the experimental results testify that our proposed model can be used in sports and human activity recognition tasks to a great extent.

5.2 Future Work

The classification of sports video is quite a novel research concept, and for this task, a few methodologies have been established. Besides, the implementation of this job on Conventional Bangladeshi Sports Video Dataset is a new approach that seemed very interesting to deal with. Our proposed model showed the promising result to successfully classify the sports videos. However, there is still space for improvement in this field.

In the future, we aim to work with a larger dataset as Deep Learning networks produce a better result with a larger amount of dataset. Furthermore, some more augmentation techniques are intended to use for better generalization. Additionally, we aspire to explore the fine tuning of some other pretrained models like Resnet, Inception, Xception, etc., and analyze their potency in this ground. Besides, some ensemble models are aimed to implement in future. To enhance the extensiveness of this work, some more traditional Bangladeshi sports classes can be considered.

References

[1] M. S. Islam, F. A. Foysal, N. Neehal, E. Karim and S. A. Hossain, 'Inceptb: A cnn based classification approach for recognizing traditional bengali games,' *Procedia computer science*, vol. 143, pp. 595–602, 2018 (cit. on pp. 2, 10).

[2] F. Cricri, M. J. Roininen, J. Leppänen, S. Mate, I. Curcio, S. Uhlmann and M. Gabbouj, 'Sport type classification of mobile videos,' *IEEE Transactions on Multimedia*, vol. 16, no. 4, pp. 917–932, 2014 (cit. on p. 8).

[3] S. U. Maheswari and R. Ramakrishnan, 'Sports video classification using multi scale framework and nearest neighbor classifier,' *Indian Journal of Science and Technology*, vol. 8, no. 6, p. 529, 2015 (cit. on p. 8).

[4] K. Messer, W. Christmas and J. Kittler, 'Automatic sports classification,' in *Object recognition supported by user interaction for service robots*, IEEE, vol. 2, 2002, pp. 1005–1008 (cit. on p. 8).

[5] R. Gade and T. Moeslund, 'Sports type classification using signature heatmaps,' in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2013, pp. 999–1004 (cit. on p. 8).

[6] R. Gade, M. Abou-Zleikha, M. Graesboll Christensen and T. B. Moeslund, 'Audio-visual classification of sports types,' in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 51–56 (cit. on p. 8).

[7] X. Gibert, H. Li and D. Doermann, 'Sports video classification using hmms,' in *2003 International Conference on Multimedia and Expo. ICME'03. Proceedings (Cat. No. 03TH8698)*, IEEE, vol. 2, 2003, pp. II–345 (cit. on p. 8).

[8] J. Hanna, F. Patlar, A. Akbulut, E. Mendi and C. Bayrak, 'Hmm based classification of sports videos using color feature,' in *2012 6th IEEE International Conference Intelligent Systems*, IEEE, 2012, pp. 388–390 (cit. on p. 8).

[9] J. Wang, C. Xu and E. Chng, 'Automatic sports video genre classification using pseudo-2d-hmm,' in *18th International Conference on Pattern Recognition (ICPR'06)*, IEEE, vol. 4, 2006, pp. 778–781 (cit. on p. 8).

[10] V. Ellappan and R. Rajasekaran, 'Event recognition and classification in sports video,' in *2017 Second International Conference on Recent Trends and Challenges in Computational Models (ICRTCCM)*, IEEE, 2017, pp. 182–187 (cit. on p. 8).

[11] M. A. Russo, A. Filonenko and K.-H. Jo, 'Sports classification in sequential frames using cnn and rnn,' in *2018 International Conference on Information and Communication Technology Robotics (ICT-ROBOT)*, IEEE, 2018, pp. 1–3 (cit. on p. 9).

[12] M. A. Russo, L. Kurnianggoro and K.-H. Jo, 'Classification of sports videos with combination of deep learning models and transfer learning,' in *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, IEEE, 2019, pp. 1–5 (cit. on p. 9).

[13] P. Campr, M. Herbig, J. Vaněk and J. Psutka, 'Sports video classification in continuous tv broadcasts,' in *2014 12th International Conference on Signal Processing (ICSP)*, IEEE, 2014, pp. 648–652 (cit. on p. 9).

[14] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar and L. Fei-Fei, 'Large-scale video classification with convolutional neural networks,' in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1725–1732 (cit. on p. 9).

[15] H. Jiang, Y. Lu and J. Xue, 'Automatic soccer video event detection based on a deep neural network combined cnn and rnn,' in *2016 IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI)*, IEEE, 2016, pp. 490–494 (cit. on p. 10).

[16] A. Ullah, J. Ahmad, K. Muhammad, M. Sajjad and S. W. Baik, 'Action recognition in video sequences using deep bi-directional lstm with cnn features,' *IEEE Access*, vol. 6, pp. 1155–1166, 2017 (cit. on pp. 10, 54).

[17] R. A. Minhas, A. Javed, A. Irtaza, M. T. Mahmood and Y. B. Joo, 'Shot classification of field sports videos using alexnet convolutional neural network,' *Applied Sciences*, vol. 9, no. 3, p. 483, 2019 (cit. on p. 10).

[18] N. Jaouedi, N. Boujnah and M. S. Bouhlel, '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 (cit. on pp. 10, 54).

[19] H. Ge, Z. Yan, W. Yu and L. Sun, 'An attention mechanism based convolutional lstm network for video action recognition,' *Multimedia Tools and Applications*, vol. 78, no. 14, pp. 20 533–20 556, 2019 (cit. on pp. 10, 54).

[20] A. Abdelbaky and S. Aly, 'Two-stream spatiotemporal feature fusion for human action recognition,' *The Visual Computer*, pp. 1–15, 2020 (cit. on pp. 10, 54).

[21] H. Yang, J. Zhang, S. Li, J. Lei and S. Chen, 'Attend it again: Recurrent attention convolutional neural network for action recognition,' *Applied Sciences*, vol. 8, no. 3, p. 383, 2018 (cit. on pp. 10, 54).

[22] D. P. Kingma and J. Ba, 'Adam: A method for stochastic optimization,' *arXiv preprint arXiv:1412.6980*, 2014 (cit. on p. 36).

[23] C. Schuldt, I. Laptev and B. Caputo, 'Recognizing human actions: A local svm approach,' in *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, IEEE, vol. 3, 2004, pp. 32–36 (cit. on p. 41).

[24] J. Liu, J. Luo and M. Shah, 'Recognizing realistic actions from videos "in the wild",' in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, 2009, pp. 1996–2003 (cit. on p. 41).

[25] K. Soomro, A. R. Zamir and M. Shah, 'A dataset of 101 human action classes from videos in the wild,' *Center for Research in Computer Vision*, vol. 2, no. 11, 2012 (cit. on p. 41).

[26] M. D. Rodriguez, J. Ahmed and M. Shah, 'Action mach a spatio-temporal maximum average correlation height filter for action recognition,' in *2008 IEEE conference on computer vision and pattern recognition*, IEEE, 2008, pp. 1–8 (cit. on p. 41).

[27] K. Soomro and A. R. Zamir, 'Action recognition in realistic sports videos,' in *Computer vision in sports*, Springer, 2014, pp. 181–208 (cit. on p. 41).

[28] F. Chollet *et al.*, *Deep learning with Python*, 1st ed. Manning Publications Co.: USA, 2017, pp. 160–166 (cit. on p. 50).

[29] M. Latah, 'Human action recognition using support vector machines and 3d convolutional neural networks,' *International Journal of Advances in Intelligent Informatics*, vol. 3, no. 1, pp. 47–55, 2017 (cit. on p. 54).

[30] B. Meng, X. Liu and X. Wang, 'Human action recognition based on quaternion spatial-temporal convolutional neural network and lstm in rgb videos,' *Multimedia Tools and Applications*, vol. 77, no. 20, pp. 26 901–26 918, 2018 (cit. on p. 54).

[31] A. Zare, H. A. Moghaddam and A. Sharifi, 'Video spatiotemporal mapping for human action recognition by convolutional neural network,' *Pattern Analysis and Applications*, vol. 23, no. 1, pp. 265–279, 2020 (cit. on p. 54).

[32] K. Xu, X. Jiang and T. Sun, 'Two-stream dictionary learning architecture for action recognition,' *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 3, pp. 567–576, 2017 (cit. on p. 54).

[33] K. Cheng, E. K. Lubamba and Q. Liu, 'Action prediction based on partial video observation via context and temporal sequential network with deformable convolution,' *IEEE Access*, vol. 8, pp. 133 527–133 540, 2020 (cit. on p. 54).