VIDEO SUMMARIZATION FOR CRICKET HIGHLIGHT GENERATION

Anees Ur Rehman
01-243212-001
Dr Sumaira Kausar

A thesis submitted in fulfilment of the requirements for the award of degree of Masters of Science (Computer Science)

Department of Computer Science

BAHRIA UNIVERSITY ISLAMABAD

OCTOBER 2023

Approval of Examination

Scholar Name: **Anees Ur Rehman**
Registration Number: **75906**
Enrollment: **01-243212-001**
Program of Study: **MS Computer Science**
Thesis Title: **VIDEO SUMMARIZATION FOR CRICKET HIGHLIGHT GENERATION**

It is to certify that the above scholar's thesis has been completed to my satisfaction and that its standard is appropriate for submission for examination. I have also conducted a plagiarism test for this thesis using HEC-prescribed software and found a similarity index 6%. that is within the permissible limit set by the HEC for the MS/M. Phil degree thesis. I have also found the thesis in a format recognized by the BU for the MS/M.Phil thesis.

Principal Supervisor Name: **Dr. Sumaira Kausar**

Principal Supervisor Signature:

Date:

Author's Declaration

I, **Anees Ur Rehman** hereby state that my MS/M.Phil thesis titled **"Video Summarization For Cricket Highlight Generation"** is my own work and has not been submitted previously by me for taking any degree from Bahria university or anywhere else in the country/world. At any time if my statement is found to be incorrect even after my graduation, the University has the right to withdraw/cancel my MS/M.Phil degree.

Name of Scholar: **Anees Ur Rehman**

Date: **12-09-2023**

Plagiarism Undertaking

I, solemnly declare that research work presented in the thesis titled **"Video Summarization For Cricket Highlight Generation"** is solely my research work with no significant contribution from any other person. Small contribution / help wherever taken has been duly acknowledged and that complete thesis has been written by me. I understand the zero tolerance policy of the HEC and Bahria University towards plagiarism. Therefore I as an Author of the above titled thesis declare that no portion of my thesis has been plagiarized and any material used as reference is properly referred / cited.

I undertake that if I am found guilty of any formal plagiarism in the above titled thesis even after award of MS/M.Phil degree, the university reserves the right to withdraw / revoke my MS/M.Phil degree and that HEC and the University has the right to publish my name on the HEC / University website on which names of scholars are placed who submitted plagiarized thesis.

Name of Scholar: **Anees Ur Rehman**

Date: **12-09-2023**

Dedication

All thanks to Allah Almighty, the sole helper, especially gracious, indeed most holy. This thesis is a tribute to my family especially my cherished mother and father, as well as my esteemed instructors and friends, who have always believed in me.

Acknowledgements

In preparing this thesis, I was in contact with many people, researchers, academicians, and practitioners. They have contributed to my understanding and thoughts. In particular, I wish to express my sincere appreciation to my main thesis supervisor **Professor Dr. Sumaira Kausar** for their encouragement, guidance, advice, and motivation. Without their continued support and interest, this thesis would not have been the same as presented here.

Librarians at Bahria University also deserve special thanks for their assistance in supplying the relevant literature. My fellow postgraduate students should also be recognized for their support. My sincere appreciation also extends to all my colleagues and others who have provided assistance on various occasions. Their views and tips are useful indeed. Unfortunately, it is not possible to list all of them in this limited space. I am grateful to all my family members.

Abstract

In today's digital age, where we are surrounded by nonstop video content, our study takes the forefront in the field of video summary, with a special focus on generating cricket highlights. Video summary is highly significant in today's world, particularly in the context of cricket highlights. It is essential for shortening lengthy videos, allowing viewers to save time while quickly engaging themselves in the most exciting moments in cricket matches. This method provides a quick and entertaining means to stay updated on cricket highlights in a context centered around video content. Creating automated cricket match highlights has considerable challenges, such as detecting players, umpire signals, and other critical happenings. In this study to address the above-mentioned problems, we used a two-pronged strategy to address the problem of identifying umpire gestures and cricket video frames. First, we created a customized CNN model specifically designed for binary classification, which allows us to detect cricket frame activity. Additionally, pre-trained CNN variants like MobileNetV2 and Visual Geometry Group (VGG16) as well as tried pre-train vision transformers take advantage of the power of transfer learning. We freeze the top layers of these models in this step and add a distinct classification layer designed exclusively to classify umpire gestures, with a focus on recognizing boundaries (four and six) and wickets. Following that, we identify which frames have activity generate video clips of those frames, and concatenate them to produce a cricket summary video. This strategy improved our overall accuracy, precision, and performance. The experiment results demonstrated that the VGG16 model came out on top with an incredible total accuracy of 88%. Using our own Umpire Gesture Image Dataset (UGID), we also investigated the performance of MobileNetV2, Custom CNN (CNN), and Vision Transformer models, which achieved appropriate accuracy rates of 86%, 81%, and 79%, respectively. These results validate our suggested architecture's effectiveness, demonstrating how it stands out in terms of accuracy when compared with engaging methodologies.

TABLE OF CONTENTS

LIST OF TABLE

LIST OF FIGURE

LIST OF SYMBOLS

UGID	–	Umpire Gesture Image Dataset
ML	–	Machine Learning
DL	–	Deep Learning
VGG	–	Visual Geometry Group
GPU	–	Graphics processing unit
CNN	–	Convolutional Neural Network
DCSN	–	Deep Cricket Summarization Network
MOS	–	Mean Opinion Score
LSTM	–	Long short term memory
MAAN	–	limited average attentional network
SVM	–	Support Vector Machine
OCR	–	Optical character recognition
RoNI	–	Region of No Interest
BI-LSTM	–	Bi-directional long short-term memory
BCE	–	Binary Cross Entropy
RGB	–	Red blue green

CHAPTER 1

INTRODUCTION

1.1 Introduction

Videos are one of the best and most important ways to get informational resources [1]. As the amount of digital content available on online platforms and on media grows, people require help in watching digital videos. Many videos are uploaded to web platforms every single minute. In the modern era, video data is widely used. The raw videos are often extremely lengthy and contain unnecessary material. Video summary, which extracts the most attractive and relevant parts of a video, is an effective method for getting an overview of a large collection of videos. It's the ideal technique for minimizing long data, giving users the essentials of the content without requiring an excessive amount of time. This method not only reduces the amount of data to be saved or interacted with, but it also improves resource use. Video summary is used in a variety of scenarios to assist in condensing lengthy content and eliminate extraneous details[2]. There is a lot of interesting information in the field of sports, such as cricket events. Sports analysis via video is growing in popularity these days. Cricket is an exciting sport for fans to watch. Cricket is getting increasingly popular, but due to the length and complexity of the game, automated cricket match analysis is challenging [3]. Due to the increased frequency and global coverage of sports events, collections of sports videos are increasing tremendously today. This is a significant obstacle to the research community's efficient management of sports video content. It takes a lot of time and effort to process, store, and broadcast the available collection of sports video content. Meanwhile, numerous sports are performed internationally now, some have earned a little more recognition and popularity than others [4]. Analyzing multimedia content is hard work for both people and machines. The primary goal of video summary or highlight extraction is to identify the game's most important moments and extract them, allowing viewers to see the highlights that are most interesting to them. Keyframe extraction and video

skimming are the two primary types of video summarization [5]. Key frames lack motion information although they are important for indexing videos. In this study, try to develop a novel method to automatically create a shorter version of a cricket match's video because doing so manually requires a high level of expertise and professional tools and it is a time-consuming task as well. To identify objects in videos and photos, GPU-constrained hardware and computer vision are employed. Deep learning algorithms both produce accurate results and can predict unknowable future data [6]. Visual recognition systems that have made significant contributions to research include segmentation and image classification. The key objective of this project is to create a powerful deep neural network capable of generating engaging highlights from diverse cricket match footage automatically and reliably. These videos are from a custom dataset "Umpire Gesture Image Dataset" (UGID) that includes both high and low-quality images. Umpire gestures are important in cricket because they serve as a vital source of interaction between players, officials, and spectators. These motions serve as quick visual clues that express the umpire's critical decisions and rulings. These actions are extremely important, from marking boundaries, whether fours or sixes, to signifying a wicket with a raised finger or the iconic "out" gesture. Umpire gestures effectively reflect the match's emerging dynamics, changing the game's path and leading to moments of thrill or disappointment. Using umpire movements in the video summary process is a unique technique for capturing significant events in a cricket match. By discovering and classifying these movements within the video material, it is feasible to automatically curate a shorter version that highlights the match's essential events. The goal is to capture the most thrilling and visually engaging moments from cricket games. Another portion of the research is examining the results of categorizing crucial match elements using a powerful and extensible deep learning method. This research presents a novel way for automatically making cricket highlights. This is accomplished by collecting key frames from videos and using video summarizing techniques to provide succinct summaries of occurrences. It excels at creating summaries. This approach can be used in the context of cricket to detect noteworthy occurrences throughout a match, such as umpire signals, boundaries (including fours and sixes), and wickets. Highlights are then created in accordance with these events. Attempts have been made throughout countless cricket matches and video highlights to construct a thorough list of events that are universally considered significant. There are two main modules in architecture. System output comes after input in the first. For example, when the umpire raises his finger to signal an out, it signifies a critical moment, perhaps resulting in

a wicket or a vital decision. Incorporating these gesture-triggered chunks into the summary video would effectively maintain the spirit of the contest, including its memorable moments. Umpire gestures function as natural breakpoints within the game, providing them the perfect indicators for composing a brief but accurate review that highlights the game's key moments. A Convolutional Neural Network (CNN) is a better approach to perform binary classification [7] utilized on a customized image dataset of umpire gestures. The goal is to determine whether a given frame in the video footage represents an activity, with the activity being the specific umpire gesture. CNNs are a proficient deep learning model built to automatically extract features from images, and their success in image classification applications is generally known [8]. This study examines the various events that occur throughout a cricket match. an attempt to classify various umpire gestures using computer vision along with deep learning. As a result of CNN's capacity to distinguish umpire actions, the summary video may effectively reflect the main event of the match, emphasizing crucial moments marked by these motions. Using a convolutional neural network, a custom dataset of cricket images of umpire signals, and videos has been obtained to generate the highlights of a cricket match.

1.2 Problem Description and objectives

In this study, I worked on video summarization for cricket highlight generation, with specific attention to umpire activities. The main difficulty is to automate the process of summarizing cricket videos by recognizing and collecting crucial events, such as umpire signals and key moments while removing replicated and less significant information. The primary goal of this research is first, to create an umpire gesture image dataset developed for cricket highlight generation, second, to extract frames from video and use these frames to pass the model to predict whether activity exists or not and select some activity frames from all frames, and finally, to use these selected frames to create concise and informative cricket highlights. Therefore our task is to figure out how to automatically select these interesting moments and generate highlight videos without requiring someone to do it manually. By achieving these goals, the study hopes to change the efficiency and user experience when producing cricket match summaries, allowing for a more simplified and automated process for video summarization.

1.3 Research Contribution

This study provides significant advances to the field of cricket video summary, with a particular emphasis on umpire gestures. The most important accomplishments include three primary domains. First, a unique dataset dedicated to cricket umpire gestures is carefully collected, and customized to the specific requirement of generating cricket match highlights. This dataset is extremely useful for training and assessing gesture recognition models. Second, identify and extract crucial events from cricket match videos. Those events include crucial events such as umpire signals and significant game situations. Finally, a solid framework is developed to produce a brief and informative cricket summary based on the extracted significant events. These contributions, taken together, move the field of cricket video summary forward by overcoming challenges related to umpire gestures.

1.4 Thesis Organization

The thesis is set forward as follows. The second chapter covers the related work which has been completed for title generation. The third chapter introduces the current study's data set preparations and proposed technique. In Chapter 4, the results are extensively addressed. The fifth chapter presents the conclusion and observations.

CHAPTER 2

RELATED WORK

The needs of fans and experts for many types of information may be fulfilled through the automatic analysis of sports videos [9]. Sports video analysis includes a wide range of applications, including player locations, ball trajectory extraction, content extraction and indexing, summarization and highlight identification, and many more. Due to features such as the same appearances, complicated challenges, dynamic backgrounds, unpredictable actions camera fluctuation, low-textured areas, transmission video editing, the inadequate resolution of distant players, and blurred movements, identifying and following players is extremely difficult [10].

2.1 Supervised Approaches

Bhalla et al [11] study proposes a novel method for automatically recognizing and summarizing crucial cricket match events. Critical events such as boundaries, wickets, and other playfield events have been extracted from a cricket match using techniques such as the recognition of optical characters, audio detection, and replay detection. The full sequence of events is then put together to make the highlights video for the cricket game. Researchers conducted several qualitative and quantitative investigations to put the model to the test. Apply the technique in the following order, shot detection, voice detection, scorecard identification, and lastly highlight production. N. Harikrishna et al [12] used a histogram to identify cuts in cricket video and a moving frames window approach to detect progressive transitions. Using this technique, shot-over segmentation is possible.Dange, B et al [13] suggest a method that generates highlights using sports-related footage automatically, leveraging video analysis techniques to generate brief summaries. This method uses audio cues, scoreboards, image understanding, and players' celebrations as signals to identify crucial cricket match moments based on event and excitement elements. A CNN-based system extracts various information from video to

provide classified highlights by combining audio, scoreboard, scene, and players modules. Users will receive a categorized and ordered summary video. To maintain event continuity, CNN is trained to extract scorecard text, and initial frames are detected. Baranwal, R et al [14] purpose of this research is to identify exciting occurrences in videos by combining data from the audio and video domains. First, a technique for separating the audio and video components is proposed. After that, the threshold is determined by measuring the" level of excitement" utilizing parameters like amplitude and spectrum center of gravity retrieved from the commentator's speech's amplitude. The research utilizing genuine cricket videos shows that these characteristics are closely related to how humans rate activity. To create highlights of cricket, audio/video information is finally combined with time-ordered scenes. Nandyal, S et al [15] proposed method, the Convolutional Neural Networks (CNN) model is utilized to extract characteristics and classify identified frames into Umpire positions of six event classes. These six cases were created using an entirely novel dataset of umpire action images. After extracting the image's features from each training image of an umpire in action, the model initially creates an identified feature knowledge testing. If the image in the frame was classified as "SIX, NO BALL, WIDE, OUT, LEG BYE, or FOUR," the acquired frame was allocated to one of the six categories and utilized to construct the Cricket Sports Event video summary. Nasir et al [16] proposed a non-learning technique that uses textual information to recognize three significant events in cricket videos are boundary (4), a six (6), and a wicket. Score captions were obtained from the input video using picture averaging, and changes in the scoreboard and wickets counter were then examined. To extract the text from the score captions, the input video frame was separated using the mean and standard deviation. Morphological variables are used to remove noise and outliers. A set of video frames was produced against each keyframe to generate the summary video. Rafiq et al [17] research focuses supervised approach using cricket as a benchmark to define five unique scene types that are batting, bowling, boundaries, viewers, and closeup scenes. The study makes use of a pre-trained Convolutional Neural Network approach, specifically the AlexNet architecture, for efficient scene classification. Three new fully connected layers are added to the already existing AlexNet CNN architecture using transfer learning and its pre-established weights. The model is then trained effectively using a selected sports dataset that has suitable labels. Gaikwad et al [18] proposed an efficient shot classification technique for sports-related videos based on AlexNet Convolutional Neural Networks (AlexNet CNN) in this paper. Using an eight-layered network comprised of 5 convolutional layers along with

three fully connected layers, the proposed method categorizes the shots by length, medium, close-up, and out of field shots. Response normalizing and dropout layers on feature maps optimized total training and validation efficiency throughout a diverse cricket dataset. The Alexnet CNN architecture of deep learning was used to classify sports video shots in the field. The network has five convolutional layers, commencing with three maximum pooling layers, which puts it deeper than standard CNN. Solayman et al [19] proposed Research in the field of video summarizing, which is the process of extracting significant moments from lengthy videos. The Deep Cricket Summarization Network (DCSN) is utilized to automatically extract crucial shots and generate a summary of that input video. The CricSum dataset addresses the issue of restricted data availability. The method employs a sequential structure, with extensive videos performing as inputs and brief summaries serving as outputs, and is measured by the Mean Opinion Score (MOS) method. The DCSN uses a decision-making process to pick important frames according to frame level information. A system based on Reinforcement Learning ensures a diversified summary while at the same time using semantic content via frame-level annotations.

2.2 Unsupervised Approaches

Shingrakhia et al [20] proposed a hybrid machine-learning strategy for highlighting cricket videos is presented in the paper. It identifies crucial times by utilizing enthusiasm, object, and event aspects. Using a dynamic threshold, speech-to-text system, and SGRNN-AM, the method analyzes audio to extract appealing clips. HRF-DBN is used to classify scenes inside these clips. Scorecard frames are used to extract player and action information, and SGRNN-AM detects events such as borders. The addition of an attention module improves accuracy. The approach works well on a variety of cricket video datasets. The [21] technique for recognizing shot boundaries in cricket footage was based on a YUV picture histogram. Both local and global histograms were used to improve segmentation results. To characterize shot boundary cut or fade kinds, the K-Nearest Neighbor (KNN) classification approach was utilized. By integrating local histograms matching each image block with a global histogram, better shot segmentation results were obtained. This method cannot determine the shot border between shots with comparable backgrounds. The authors of [22] provide a broad structure for video summarizing using attention modeling. By applying computation attention models despite fully understanding the semantic content of a specific video, the framework minimizes the requirement for elaborate heuristic strategies in video summarizing tasks. The authors of

[23] proposes an unsupervised method for video summarizing. The research effectively analyzes video frames using standard vision-based techniques using a deep learning-based feature acquisition method. The next application of several clustering approaches seeks to discover keyframes that successfully capture the core of the video. Deep learning based features are better than traditional features on the SumMe dataset. Junyu Gao et al [24] present an unsupervised video summarizing method that takes advantage of clip-clip interactions to enable contextually aware clip selection. This method obtains precise challenging tasks by treating clips simply as graph nodes and exploiting node feature magnitudes. The proposed approach beats benchmarks using a multi-task framework that includes reconstruction and distinct constraints, offering advances in unsupervised video summaries. Ye Yuan et al [25] work presents an unsupervised video summarizing method based on reinforcement learning and shot-level semantics. The method employs an encoder-decoder architecture, including a convolutional neural network over feature extraction including a bidirectional LSTM for keyframe selection. The approach's efficiency and positive results are shown through the evaluation of the benchmark datasets. The 3DST-UNet-RL model for video summarization is introduced in this paper [26]. It generates video footage using a three-dimensional spatio temporal U-Net and selects frames using reinforcement learning. The method can work simultaneously in unsupervised and supervised scenarios. The usefulness of the system in recognizing relevant information and decreasing storage costs is demonstrated by testing it against a standard video summarizing benchmark and a healthcare video summarization.

Although the method is not comprehensive, extra preprocessing procedures will be required before producing a video summary. As such, video exploration is time-consuming and inefficient in practice. This led to the development of a from-beginning-to-end deep learning approach. Regarding the fact that some prior papers depict the video summarizing job to be an unsupervised problem and offer solutions to it, an unsupervised strategy is a supervised approach overall.

2.3 Weakly Supervised Approaches

Gygli et al [27] used supervised learning and inference to connect several attributes in video frames, including aesthetics, item occurrences, and motion. Weakly supervised algorithms required a small number of annotations yet produced excellent results. Weakly supervised algorithms required a small number of annotations yet produced excellent results. This [28] study presents a novel weakly supervised strategy for video summarizing that only requires

a little human-crafted training material. It employs a versatile deep 3D CNN architecture to analyze video relevance having only video-level annotations and no manual data. It finds critical areas within a video by automatically learning from videos in the same genre. This method beats existing approaches, demonstrating its potential to better video summaries without extensive annotation. W-TALC is a novel framework [29] for Weakly-supervised Temporal Activities Localization and Classification. W-TALC prevails over state-of-the-art approaches in detecting activities with fine precision, illustrating its ability to reduce the requirement for difficult frame-wise annotations in activity localization. Sijia Cai et al [30] addresses the problem of video summarization, which is subjective and frequently requires lengthy annotations. A generative model named VESD was developed for using web-crawled videos. It incorporates a variational autoencoder to study video semantics and relevancy and a summary-generating encoder-attention-decoder. VESD beats previous algorithms in experiments on the CoSum and TVSum datasets, which makes it a valuable tool enabling web-based video summarizing. This study [31] focuses on action localization in videos having a weakly supervised approach, with the goal of reducing dependency on thorough frame-level annotations. Introduced Action Selection Learning (ASL), a novel method designed mainly to overcome class bias problems in frame-level classification. This work [32] improves video highlight detection by taking into consideration both visual and audio aspects. On two benchmark datasets, the approach is employ, which includes a bimodal attention mechanism and a noise-filtering methodology, superior to existing techniques. Yuan et al [33] proposed a technique in which the limited average attentional network (MAAN) stands out in the field of weakly-supervised action localization. It effectively addresses the challenge of precisely identifying full action regions by reducing the influence of excessively prominent locations. MAAN's novel MAA module efficiently balances the importance of different regions, which results in improved action localization accuracy. Extensive tests on large video datasets show MAAN's efficacy.

Table 2.1: Litrature Analytical Review

Reference	Years	Techniques	Dataset	Evaluation	Limitations
[11]	2019	GIST + OCR and CNN + OCR.	Cricket match videos dataset.	accuracy of 89.45%	Despite the fact because it misses Some of the red-circled fours and sixes.
[12]	2011	GSP algorithm, SVM and SMO algorithm.	Cricket match videos dataset.	accuracy of 87.8%	Performs poorly in the instance of SIX. This is because of the designed technique's Inability to differentiate between a six and a four.
[13]	2022	CNN + SVM and Gray scale + CNN + SVM.	Sports-related footage dataset.	accuracy of 84.7%	The results produced were superior to audio-based highlights generators, although more work is needed to reach higher accuracy.
[15]	2022	CNN classifier.	SNWOLF dataset.	accuracy 98.20%	Limited dataset.
[16]	2018	Morphological operators and OCR.	Thirty cricket video dataset.	precision of 94.8%	Additional preprocessing required.
[17]	2020	Pre-trained AlexNet Convolutional Neural Network (CNN).	Sports video dataset.	99.26& accuracy	Limited dataset.
[20]	2022	Hybrid machine-learning strategy.	Cricket video dataset.	96.82& precision and 96.32& accuracy	The precision of the video the summary process is also affected by learning and non-learning processes.
[22]	2016	CBIR system and GRFMN.	Sports images dataset.	average accuracy 80%	Limited dataset.
[23]	2020	Vision-based techniques, deep learning-based feature acquisition, clustering.	SumMe dataset.	accuracy of 81%	Limited dataset.
[25]	2023	Reinforcement learning, encoder-decoder architecture, bidirectional LSTM.	SumMe, TVSum, CoSum, and VTW dataset.	accuracy of 83%	Regarding better findings, the model needs to be improved and extended.

CHAPTER 3

METHODOLOGY

This chapter explains how we make interacting cricket match highlights. This section goes over the proposed technique in depth. We employed a methodology to solve this research problem that is shown in figure 3.1. In the sections that follow, we will go through all the specifics of our custom-created dataset. Then we discussed each component of the suggested approach in detail.

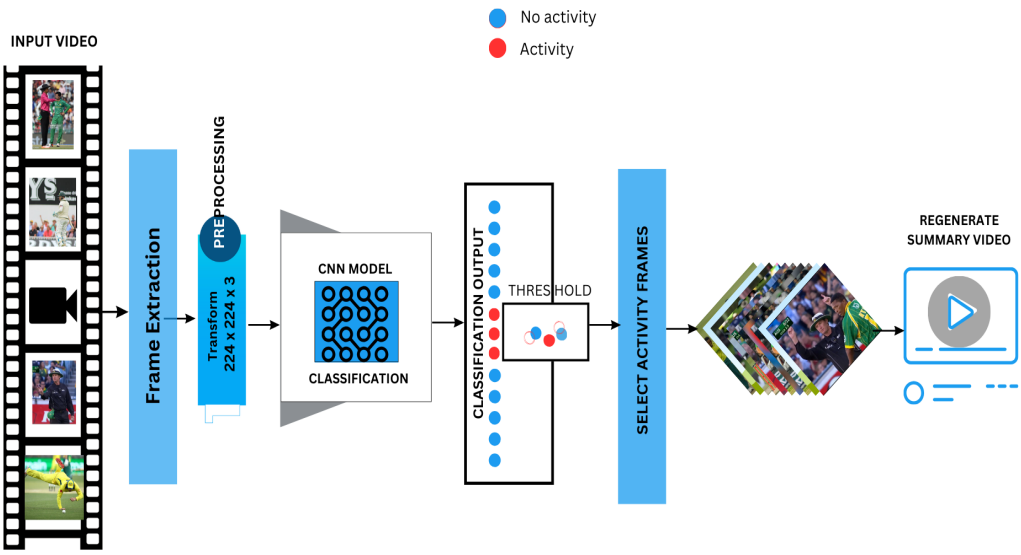


Figure 3.1: Proposed Methodology

3.1 Data Acquisition and Division

This section contains a detailed explanation of the dataset that was used in this study to train and evaluate the proposed method. The "Umpire Gesture Image Dataset" (UGID) was developed to help in the development and testing of a model capable of recognizing various umpire gestures during a game of cricket to create video summary. The dataset contains an extensive variety of photos taken from reliable websites such as Getty images, iStock, and Alamy. further I create a custom video dataset for video summarization.

3.1.1 Cricket Video dataset

A unique and specialized dataset was thoroughly constructed in order to further video summarizing techniques. Full-length cricket match videos were acquired from popular T20 cricket matches on YouTube, and the entire match video was divided into parts. This unique dataset comprises 15 cricket match videos, each consisting of four over and varying in length from 15 to 20 minutes. To go along with these full-length videos, brief summary videos ranging in duration from 2 to 3 minutes were painstakingly created. These summaries concentrated on condensing critical occasions, such as boundaries and wickets, through the use of Python library, including MoviePy, and additional refinement with CapCut. The generation of this dataset was motivated by two factors.First and foremost, it provided as a starting place for experimentation with end-to-end video summary techniques. Second, it provided the route for future initiatives, there are no standardized cricket video datasets available we try to developed specific datasets for video summarizing tasks in the video analysis. Looking into the future, we have ambitious intentions to considerably extend this dataset. Our primary goal is to collect a significantly larger library of cricket videos, which will be critical for training and refining pre-trained video summarization transformer models. This endeavor is motivated by a goal to improve cricket video summaries using cutting-edge methodologies. We hope that we can speed up the video summarizing end-to-end techniques by leveraging the capabilities of vision transformers. This innovative project fills a gap in the field by addressing the shortage of specialized cricket video datasets, providing a foundation for extraordinary breakthroughs in video summarizing approaches.

3.1.2 Dataset Description

The UGID dataset consists of images of umpire gestures along with their surrounding area. Images have been collected from many kinds of online re-

sources known for their extensive collections of cricket-related images. These platforms were chosen to provide a diverse and inclusive representation of cricketing activities and then dataset was divided into two categories: "activity" and "no activity." The "activity" class holds instances of umpire gestures, particularly those indicating a "Four," "Six," or "wicket." The "no activity" class, on the other hand, includes random images of cricket fields, players, umpires, and their surroundings with no significant umpire gestures that is shown in figure 3.2.

Figure 3.2: Dataset samples

3.1.3 Train-Test Split

The UGID dataset was further split into subsets for training and testing for use in the development and testing of the proposed model:

Training Data

The training subset represents 80 percent of the UGID dataset that is used to train the CNN model. It is made up of an appropriate balance of "activity" and "no activity" images.

Testing Data

The testing subset is used to evaluate the trained model's generalization and efficiency. It contains 20 percent of images that are different from those in the training dataset, providing unbiased evaluation. The testing subset, like the training subset, has a balanced distribution of "activity" and "no activity" pictures.

The train-test split was devised to reduce data leaking and overfitting while allowing an accurate evaluation of the model's capacity to recognize umpire gestures in a variety of cricket match circumstances.

3.1.4 Dataset Detailed Analysis

We give a detailed study regarding the Umpire Gesture Image Dataset (UGID) in this section. The analysis gives information about the dataset's composition, such as the proportion of classes, the quantity of images are shown in table 3.1.

Table 3.1: Dataset Detailed Analysis

Class	**Training Set**	**Testing Set**	**Total**
Activity	600	150	750
No activity	415	105	520

3.1.5 Data Preprocessing

The collection consisted of images of cricket umpire gestures labeled "activity" or "no activity." It was divided into training as well as testing. Preprocessing required resizing images to 224x224 pixels and segmenting data into training and validation subsets. The Image Data-Generator in TensorFlow simplifies data management. Class distribution and batch grouping increased efficiency. Specific preprocessing improved data quality and the performance of CNN models.

3.1.6 Data Augmentation

The approach began with the use of augmentation techniques, which introduced controlled alterations to images through transformations such as rescale, zoom range, horizontal flip, rotation range, width shift range, and brightness range. This broadened the dataset, Augmentation techniques can

be applied to three images taken from the dataset shown in figure 3.3 to increase the size of the dataset, UGID, and the value of model training and its subsequent results. Augmentation is the process of applying multiple modifications to images in order to create new training samples with significant variations. This augmentation technique not only increases the quantity of the dataset but also helps to improve the model's reliability and generalization. Changing the dataset with augmented images has the potential to increase model performance while minimizing the risk of overfitting, leading to more dependable and accurate outcomes in CNN model.

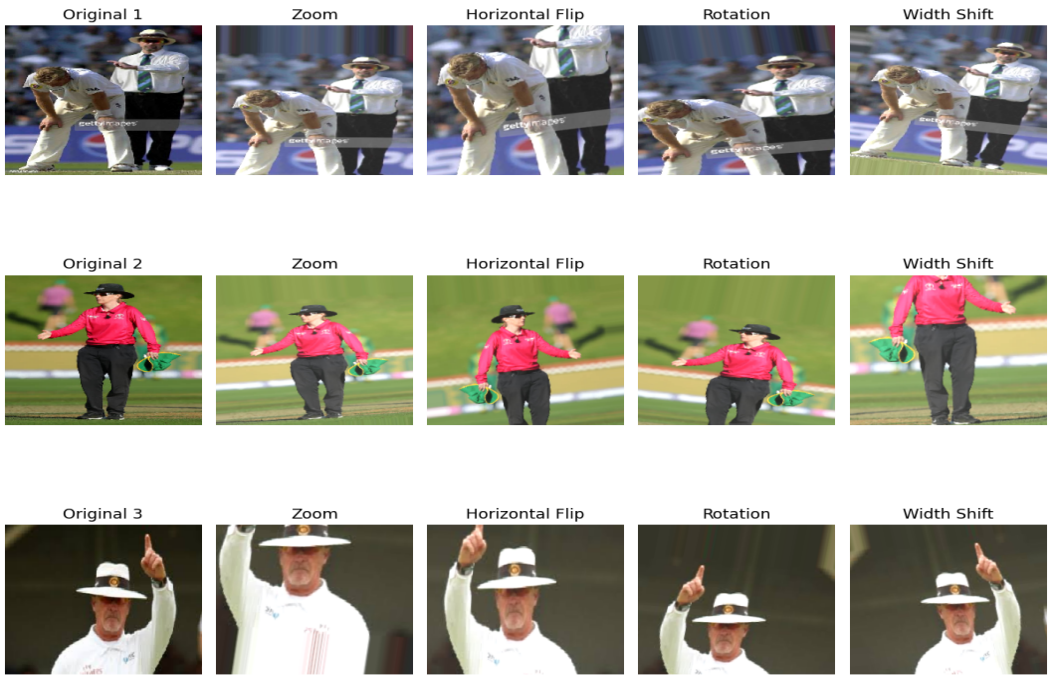


Figure 3.3: Apply augmentation on a sample of three images from the dataset.

Augmentation increases the size of our dataset, and the variations in size are calculated by measuring the dataset before and after augmentation as shown in figure 3.4

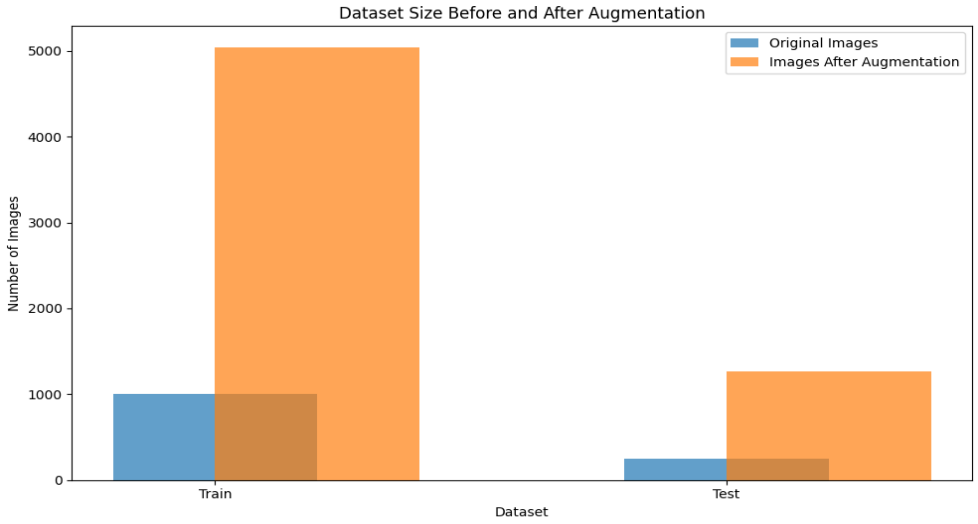


Figure 3.4: Dataset before and after augmentation.

3.2 Proposed Methodology

A described approach to recognizing umpire gestures in matches of cricket and creating a summary video that captures the recognized gestures and their context. The methodology consists of numerous related processes, ranging from data preprocessing through the building of deep learning models, to frame extraction and video reconstruction.

3.2.1 CNN Model

CNNs (Convolutional Neural Networks) are a deep learning architecture used to identify images [34, 35]. Gesture recognition is an important and extensively used approach to natural interaction between humans and computers, providing speed and ease by eliminating the time-consuming manual feature extraction of older approaches [36].Convolutional Neural Networks have become known as a recognized preferred method of recognition, computer vision, and image classification task [37]

The CNN model is the best option to classify images as "activity" or "no activity." There are several layers in the architecture for feature extraction and classification based on Activity in frames.

Convolutional Layers: To capture precise characteristics in the images, many layers of convolution with more depth were used.

Max Pooling Layers: Max-pooling layers have been used to reduce over-fitting and reduce feature maps.

Fully Connected Layers: To minimize over-fitting, flattening features from layers of Convolutional neural networks were passed through thick layers, including dropout regularization. For binary classification, the last layer is an output layer using one neuron and a sigmoid activation function. This model has 14 layers in total Details are shown in table 3.2.

Table 3.2: CNN Model Architecture and components

Layer Type	**Output Shape**	**Parameters**
Conv2D (conv2d_8)	(None, 222, 222, 32)	896
MaxPooling2D (max_pooling2d_8)	(None, 111, 111, 32)	0
Conv2D (conv2d_9)	(None, 109, 109, 64)	18,496
MaxPooling2D (max_pooling2d_9)	(None, 54, 54, 64)	0
Conv2D (conv2d_10)	(None, 52, 52, 128)	73,856
MaxPooling2D (max_pooling2d_10)	(None, 26, 26, 128)	0
Conv2D (conv2d_11)	(None, 24, 24, 256)	295,168
MaxPooling2D (max_pooling2d_11)	(None, 12, 12, 256)	0
Flatten (flatten_2)	(None, 36,864)	0
Dense (dense_6)	(None, 512)	18,874,880
Dropout (dropout_4)	(None, 512)	0
Dense (dense_7)	(None, 256)	131,328
Dropout (dropout_5)	(None, 256)	0
Dense (dense_8)	(None, 1)	257

3.2.2 MobileNetV2 Model

MobileNetV2 [38] is a convolutional neural network architecture designed for on-device computer vision applications, namely on mobile and embedded

devices. It starts with a convolution layer and then moves on to the next more novel "Inverted Residual Blocks" such as depthwise separable convolutions, linear bottlenecks, plus skip connections. There is an additional expansion layer between these blocks that gradually increases the number of channels. The network ends with a fully connected classification layer. The versatility of MobileNetV2 is increased by hyperparameters such as the width multiplier for altering channel sizes, the resolution multiplier for controlling input resolution, and an expansion ratio for determining the number of output channels. Because of these design choices, MobileNetV2 is an excellent candidate for resource-constrained contexts, finding a compromise between computing efficiency and accuracy in activities such as classification and detection. Train a custom dataset using the mobileNetV2 model to improve outcomes and perform classification tasks.

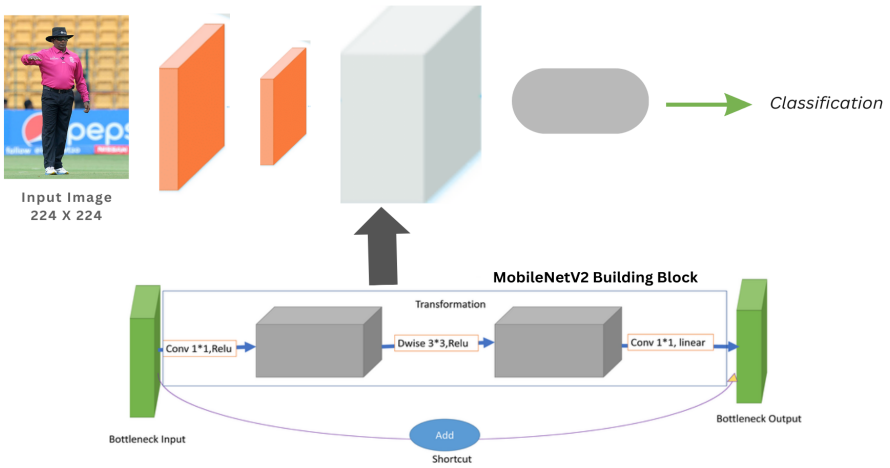


Figure 3.5: MobileNetV2 architecture

3.2.3 Architecture of the VGG16 model

Using transfer learning, we used the VGG16 model [39] for binary classification on our umpire gesture dataset. The pre-trained VGG16's top layers were frozen, keeping learned features. The input layer expects RGB pictures of 224x224 pixels. The core architecture had five sets of convolutional layers, followed by max-pooling, which increased the complexity of the filters. Each convolutional layer used three 3x3 filters while max-pooling used two 2x2 windows with a stride of two. Following the convolutional layers were three thick layers, the first two of which included 4096 ReLU-activated neurons. The final dense layer, which was originally designed for ImageNet's 1000-class classifica-

tion, has been replaced with a single neuron and sigmoid activation for binary classification. A Dropout layer with a dropout rate of 0.5 improved model generalization. The model was built using the Adam optimizer and the binary cross-entropy loss. Early stopping, stopping when validation loss stands, and learning rate reduction on level all assist training by dynamically adjusting the learning rate. These architecture and training changes were critical to improving model performance for binary classification.

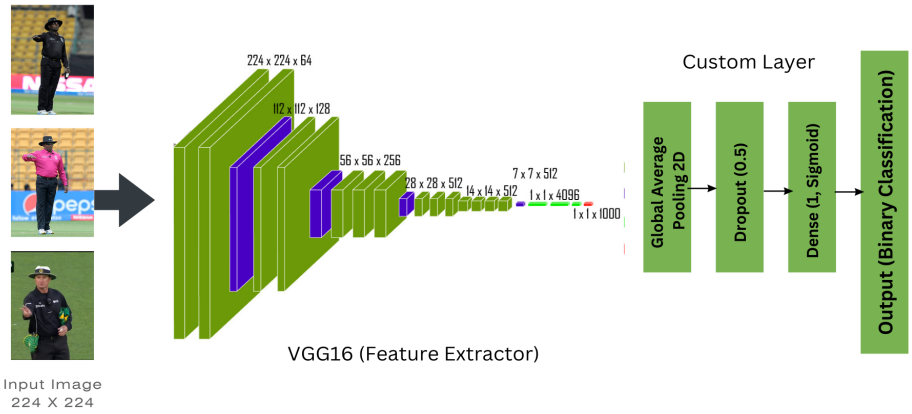


Figure 3.6: VGG16 model architecture

3.2.4 Architecture of Vision Transformer model

We used a Vision Transformer (ViT) [40] model for binary image classification in our research on the UGID custom dataset, with the goal of identifying frames with activity. To do this, we used the Hugging Face Transformers library to easily integrate a pre-trained ViT model into our workflow. We used a ViT architecture that has numerous key components. To begin, an input embedding layer was used to transform image patches into meaningful embeddings. Positional encodings were implemented concurrently to provide spatial information to the model. The architecture's heart was created by a stack of Transformer encoder layers, each capable of capturing intricate global correlations among patches within images using multi-head self-attention processes and feedforward neural networks. For use in binary classification, we added a classification head to the model, which is commonly implemented as a fully connected layer followed by a softmax activation function. It's worth noting that our ViT model's specific architectural details and hyperparameters were customized for the task in the same direction, with these critical settings de-

rived from a pre-trained model chosen from the extensive Hugging Face model hub, ensuring information and reliability in our methodology.

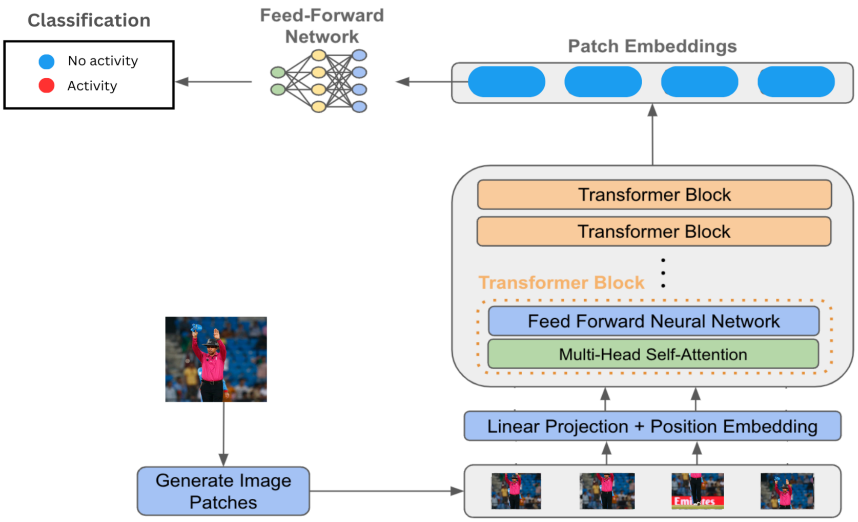


Figure 3.7: ViT model architecture

3.2.5 Frame Extraction:

videos that were over 5 minutes long were uploaded. Extraction of frames at a constant rate of 5 frames per second produced a collection of frames, from the videos. These frames are properly sorted, with distinct frame numbers and timestamps matching the original video sequence are shown in figure 3.8. A structured CSV file is used for easily storing all frames along with temporal information. These selected frames, enriched with potential activity indications, have been introduced into the predictive model, allowing it to determine the presence or missing activity.

Figure 3.8: Frame Extraction

3.2.6 Frame Classification and Prediction

Extracted frames via the videos were sent into the CNN model for classification, which determined whether they indicated activity. A set of cricket match frames was used for the frame analysis. Utilizing the pre-trained weights and architecture, the model predicted "activity" or "no activity" for each frame. The findings were saved in a CSV file, which included the frame number along with timestamps. The number of activity frames was calculated by the number of predictions that were less than or equal to 0.5, indicating the existence of an umpire gesture. Similarly, the absence of activity frames correlated to predictions greater than 0.5, indicating the absence of meaningful gestures. This research enabled us to determine the occurrence of umpire gestures in cricket match frames, offering insights into the frequency and time of critical events. Each frame was assigned a prediction score and a class label (0 for activity, 1 for no activity) by the model that is shown in figure 3.8. For prediction scores, a 0.5 threshold was used. Frames with successive activity were pooled, and the frame with the earliest timestamp was chosen. This method ensured that critical activity moments would be collected.

3.2.7 Summary Video Creation

A clip was created for each selected frame by mixing the previous 120 frames and the next 20 frames. These frames were combined to form a unified clip expressing a specific activity event. The resulting video clips were combined to create a summary video. This detailed video highlighted the crucial events from the original videos, providing an informative overview of the events spotted in the cricket match.

3.2.8 Optimization

The Adam optimizer is a well-known adaptive learning rate technique that is used to train machine learning and deep learning models, particularly neural networks. We used it as a model for our training. It mixes RMSprop and Momentum method features. Adam keeps gradient moving averages, changes learning rates for each parameter adaptively, and handles sparse gradients effectively. These characteristics make it useful for accelerating convergence and enhancing optimization in a variety of applications.

3.2.9 Early Stopping

The early stopping technique was used as a critical component of our model training strategy in the context of our research. Early stopping examines

the model's performance on a separate validation dataset during training and stops the training process if it finds a lack of improvement in validation loss over a predetermined period of consecutive epochs. This prevents the model from overfitting to the training data and improves its capacity to generalize to new, previously unknown data. We ensured that our model established an optimal balance between training and generalization by incorporating early stopping with a patience parameter of 5, contributing to the overall effectiveness of our research in binary classification tasks.

3.2.10 Binary Cross Entropy

In the research we conducted, we used Binary Cross-Entropy (BCE) as a key component in the training phase of our model for binary classification tasks. BCE is a popular loss function that calculates the difference between expected and actual binary labels (0 or 1) for each data point. The BCE loss for a single data point is determined as follows shown in the equation.

$$L(y, p) = -[y \cdot \log(p) + (1 - y) \cdot \log(1 - p)] \tag{3.1}$$

Here, y represents the true binary label, and p is the predicted probability for the positive class. BCE (Binary Cross-Entropy) encourages well-calibrated probability predictions and is essential for gradient-based optimization during training. By minimizing BCE, our model learned to make precise and calibrated binary predictions, which played a pivotal role in achieving accurate results across various binary classification tasks within our research.

CHAPTER 4

ANALYSIS & RESULTS

This section covers the results of experiments performed to evaluate the performance of the recommended methodologies. We investigated this study using the custom umpire gesture image dataset UGID. The experimental protocol and results are presented first. Later, we will demonstrate many scenarios of experimentation for video summarizing. Finally, provide a summary by comparing several CNN variations approaches.

4.1 Model Performance Metrics

Evaluation metrics are crucial tools for evaluating the performance of our CNN model. They provide quantifiable metrics of the model's performance in achieving its intended goals. Precision measures the model's ability to make precise positive predictions, whereas recall measures the model's ability to identify true positive occurrences. Together, these metrics provide a thorough assessment of the model's ability to accurately recognize and perform classification tasks.

Precision

Precision is a metric that quantifies the correctness of positive predictions generated by the CNN model. It indicates how many of the events displayed as positive were correct. Precision is an important evaluation parameter in our binary classification issue, which predicts the presence or absence of activity based on images. It measures the precision of CNN the model's positive predictions. In terms of mathematics:

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{1}$$

True Positives (TP) are images that were accurately classified as "activity," whereas False Positives (FP) are images that were incorrectly classified to be

"activity" when they really were not. A high precision result indicates that our model is frequently correct when predicting "activity," restricting false positives.

Recall

Recall is the percentage of positive examples that are retrieved out of the entire number of positive examples. The model's recall, also known as sensitivity, reflects its ability to capture all of the important experiences of "activity."

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{2}$$

False Negatives (FN) are images that are fully "activity" but are misclassified as "no activity." A high recall score indicates that the model detects the majority of actual "activity" instances, lowering the chance of missing important results.

F1 Score

The F1 Score acts as a balanced metric that combines accuracy as well as recall into only one value. It provides an overall evaluation of the model's performance, taking into account both false positives and false negatives. In terms of mathematics:

$$\text{F1 Score} = \frac{2 \cdot (\text{Precision} \cdot \text{Recall})}{\text{Precision} + \text{Recall}} \tag{3}$$

This metric has a value between 0 and 1, with higher numbers indicating greater model performance. It is especially important when we want to achieve a compromise between producing accurate positive predictions and identifying all relevant positive examples.

4.2 Experimental Protocol and Results

We preferred Google Colab Notebook to be our Integrated Development Environment (IDE) for coding and implementation in the framework of our experimental protocol and results because of its user-friendly design, comprehensive capabilities, and GPU support, which significantly sped up our image dataset processing tasks. We methodically constructed the code for both the training and testing phases of our model. TensorFlow library version 2.13.0-rc2 was used for training and testing our Convolutional Neural Network (CNN)

model. We additionally employed OpenCV (Open Source Computer Vision Library), a popular open-source software library for computer vision and Deep Learning tasks. We used OpenCV to perform tasks like frame extraction and creating videos from frames. We used Google Drive to store and manage frames and their related results. Our system configuration included an 11th Generation Intel(R) Core(TM) i7-7500U CPU @ 2.70GHz 2.90 GHz processor, which provided enough computing power for our tasks. In addition, we outfitted our laptop with 8GB of DDR3 RAM and a 512GB hard drive, as well as the Windows 11 operating system. This solid hardware arrangement enabled a smooth and high-performance computing environment for our tests, allowing us to acquire consistent and exact findings throughout our research activities. We carefully created an evaluation dataset concentrating on umpire gestures in cricket to examine the performance of our approach thoroughly. We uploaded cricket videos, acquired frames from them, and classified them using our model. This classification assisted us in identifying and distinguishing frames with and without umpire gestures. Then we chose specific frames and used them to create a video. We only used images from platforms that provide free access to their content to ensure that they complied with copyright regulations.

4.2.1 Evaluation of MobileNetV2 Model

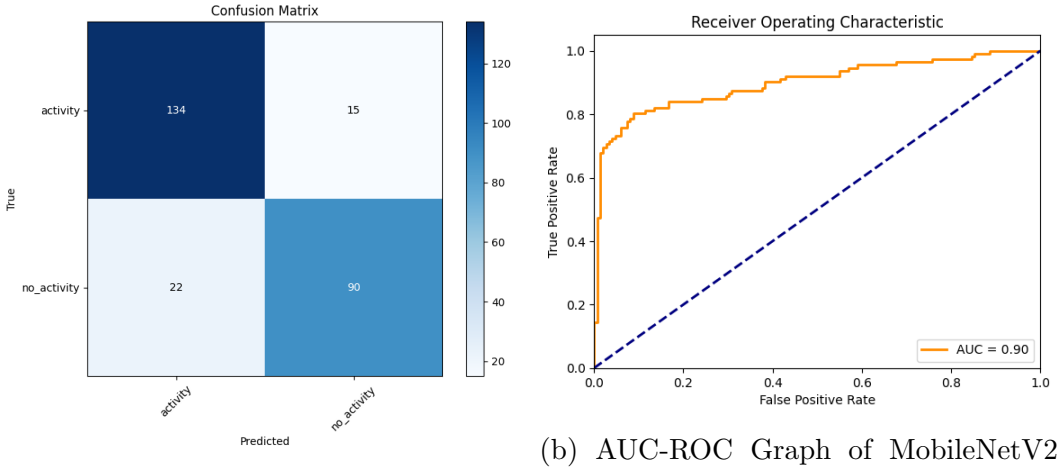
The MobileNetV2 model, employed for binary classification on the Umpire Gesture Image Dataset (UGID), demonstrated exceptional results. The model achieved an overall accuracy of 86%, underscoring its proficiency in distinguishing between "activity" and "no activity" frames in cricket videos. For the "activity" class, the model exhibited a precision of 86%, signifying that it correctly identified 86% of frames with umpire gestures. The recall for the "activity" class was 90%, indicating the model's robust ability to capture frames containing umpire gestures. The F1-score for "activity" was 88%, highlighting the model's balanced performance in identifying these critical moments. Similarly, for the "no activity" class, the model displayed a precision of 86% and an F1-score of 83%, with a recall of 80%. This indicates that the model excels in classifying frames without umpire gestures, with a good balance between precision and recall details shown in table 4.1.

Table 4.1: Classification report result of MobileNetV2 Model

Metric	Activity Class	No Activity Class	All Accuracy
Precision	86%	86%	86%
Recall	90%	80%	85%
F1-Score	88%	83%	86%

Confusion matrix

The confusion matrix provides a detailed breakdown of the model's performance, including true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN).



(a) Confusion Matrix of MobileNetV2

(b) AUC-ROC Graph of MobileNetV2 Model

Figure 4.1: Results of MobileNetV2

Hyperparameters

To further understand the model's performance and potential for optimization, various hyperparameter experiments were conducted, including adjustments to learning rate, batch size, number of epochs, dropout rate, and optimizer. These experiments help fine-tune the model for better results.

Table 4.2: Hyperparameters for MobileNetV2 model

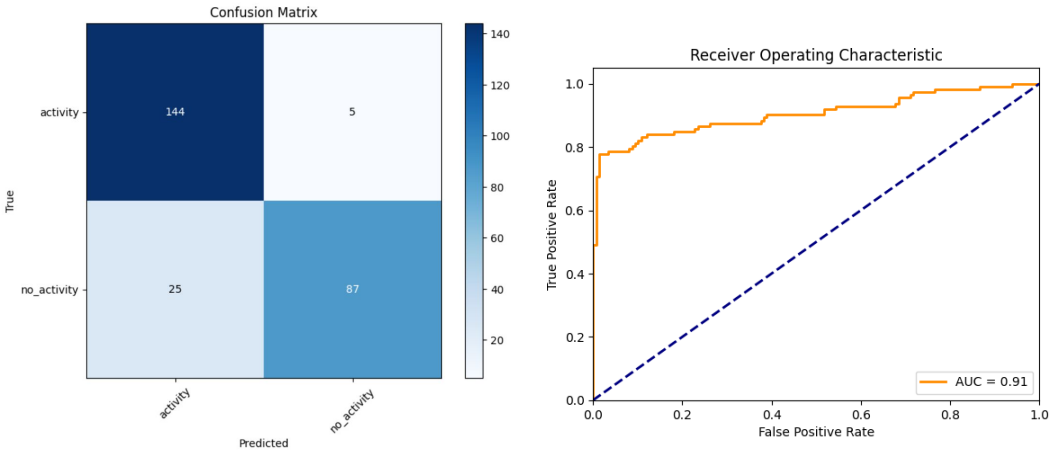
Hyperparameter	**Experiment 1**	**Experiment 2**	**Experiment 3**
Learning Rate	0.001	0.01	0.0001
Batch Size	16	32	64
No. of Epochs	50	50	50
Dropout Rate	0.5	0.2	0.3
Optimizer	Adam	SGD	Adam
Test accuracy	86	80	79

4.2.2 Evaluation of VGG16 Model

The VGG16 model produced good results for binary classification on the Umpire Gesture Image Dataset (UGID). The model has an overall accuracy of 88%, demonstrating its ability to differentiate between "activity" and "no activity" frames in cricket videos. The model achieved a precision of 85% for the "activity" class, indicating that it correctly detected 85% of frames involving umpire gestures. The recall for the "activity" class was 97%, demonstrating that the model could catch a significant amount of frames with umpire gestures. The F1-score for "activity" was 91%, confirming the model's strong ability to detect these crucial moments. In contrast, for the "no activity" class, the model exhibited a precision of 95% and an F1-score of 85%, with a slightly lower recall of 78%. This suggests that while the model excels in classifying frames without umpire gestures, there is room for improvement in recall for this class. classification report results shown in table 4.3

Table 4.3: Classification report result of VGG16 Model

Metric	**Activity Class**	**No Activity Class**	**All Accuracy**
Precision	85%	95%	90%
Recall	97%	78%	88%
F1-Score	91%	85%	88%



(a) Confusion Matrix of VGG16 Model

(b) AUC-ROC Graph of VGG16 Model

Figure 4.2: Results of VGG16

Table 4.4: Hyperparameter of VGG16 model

Hyperparameter	**Experiment 1**	**Experiment 2**	**Experiment 3**
Learning Rate	0.001	0.01	0.0001
Batch Size	32	64	16
No. of Epochs	50	50	50
Dropout Rate	0.5	0.3	0.2
Optimizer	Adam	SGD	Adam
Test accuracy	88	61	80

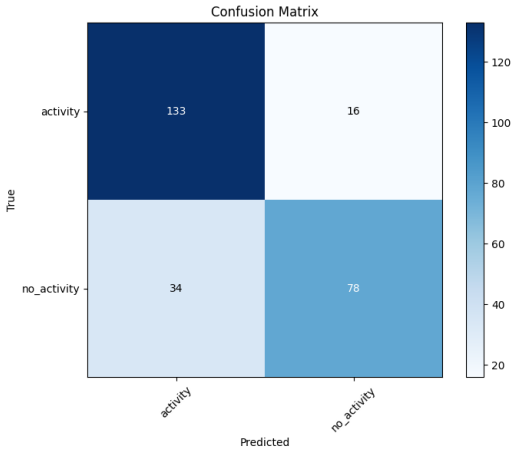
4.2.3 Evaluation of Custom CNN Model

The custom Convolutional Neural Network (CNN) model, developed for binary classification on the Umpire Gesture Image Dataset (UGID), yielded promising results. The model demonstrated an overall accuracy of 81%, indicating its ability to effectively distinguish between "activity" and "no activity" frames in cricket videos. For the "activity" class, the model achieved a precision of 80%, signifying that it correctly identified 80% of frames with umpire gestures. The recall for the "activity" class was 89%, indicating the model's ability to capture a significant portion of frames containing umpire gestures.

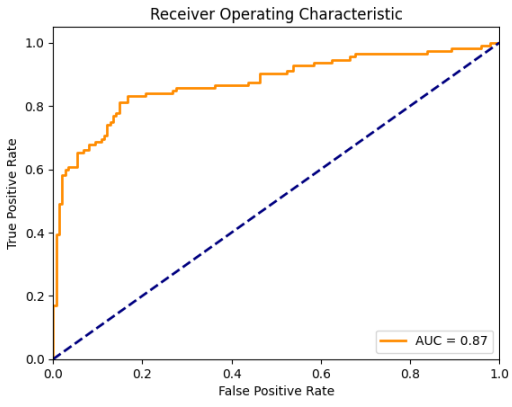
The F1-score for "activity" was 84%, underlining the model's solid performance in identifying these key moments. In contrast, for the "no activity" class, the model exhibited a precision of 83% and an F1-score of 76%, with a recall of 70%. This suggests that while the model excels in classifying frames without umpire gestures, there is some room for improvement in recall for this class.

Table 4.5: Classification report result of CNN Model

Metric	**Activity Class**	**No Activity Class**	**All Accuracy**
Precision	80%	83%	82%
Recall	89%	70%	80%
F1-Score	84%	86%	85%



(a) Confusion Matrix of CNN Model



(b) AUC-ROC Graph of CNN Model

Figure 4.3: Results of CNN

Table 4.6: Hyperparameters of CNN model

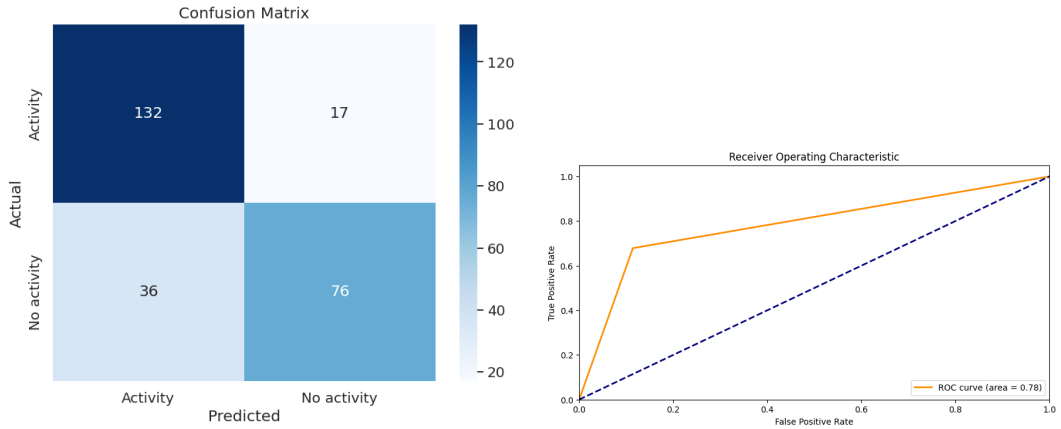
Hyperparameter	**Experiment 1**	**Experiment 2**	**Experiment 3**
Learning Rate	0.01	0.001	0.0001
Batch Size	32	16	64
No. of Epochs	40	50	30
Dropout Rate	0.5	0.3	0.5
Optimizer	Adam	SGD	Adam
Test accuracy	81	69	73

4.2.4 Evaluation of Pre-trained Vision Transformer

The pre-trained Vision Transformer model performed satisfactorily in binary classification, obtaining an accuracy of 86% during training and slightly decreasing to 79.69% during testing. When looking at the classification report, it is clear that the model performed in identifying "activity", with an F1-score of 0.83, indicating high overall performance in this category. The precision to acquire Class 1 was 0.79, indicating that the model correctly detected 79% of frames with activity, whereas the recall for Class 1 was 0.89, indicating that the model was proficient in identifying the majority of frames with activity. The model's performance remained a little lower for Class 2, reflecting "no activity," with an F1-score of 0.74. No activity class achieved a precision of 0.82, showing accurate identification of frames with no activity, but a recall of 0.68, indicating that there is potential for improvement in collecting all frames with no activity. These findings illustrate the model's ability to detect cricket action frames while also suggesting potential improvements for frames with no activity.

Table 4.7: Classification report result of ViT Model

Metric	**Activity Class**	**No Activity Class**	**All Accuracy**
Precision	79%	82%	80%
Recall	89%	68%	78%
F1-Score	83%	74%	78%



(a) Confusion Matrix of Vision Transformer Model

(b) AUC-ROC Graph of ViT Model

Figure 4.4: Results of ViT

4.2.5 Discussion

In this study, we propose to conduct a complete performance analysis of four distinct models used for binary classification on the Umpire Gesture Image Dataset (UGID). VGG16, a well-known convolutional neural network (CNN); Custom CNN, an exclusive architecture customized to our specific task, MobileNetV2, recognized for its efficiency and adaptability in computer vision tasks along with the Pre-trained Vision Transformer, an emerging model regarding the potential to transform image analysis tasks, are among these models. Our primary objective is to evaluate these algorithms' ability to appropriately classify umpire gestures in the UGID dataset. Umpire gestures are an important part of cricket matches, impacting decision-making and overall game dynamics. The ability to accurately classify these gestures is critical for producing cricket highlights based on activity frames. We intend to find that the model is best at accurately classifying umpire gestures by thoroughly evaluating their performance. This review will provide us with information on the model that can be used to efficiently extract and evaluate major events from cricket matches, allowing us to create attractive and informative cricket highlights.

Comparison of Precision

In the precision evaluation, which measures the models' accuracy in classifying frames, VGG16 stood out as the best performer, achieving the highest precision of 95% for the 'no activity' class. Both MobileNetV2 and the Custom

CNN model achieved competitive but slightly lower precision values for the 'no activity' class, both at 86%. In the 'activity' class, MobileNetV2 displayed the highest precision (86%), followed closely by VGG16 at 85%, and the Custom CNN model at 80%, Meanwhile ViT transformer gave results of precision for activity class 79% and 82% for no activity class. these details are shown in figure 4.5

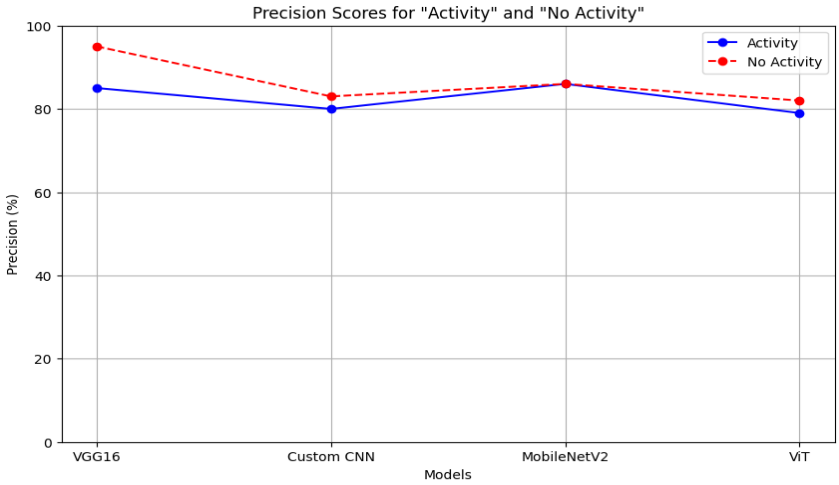


Figure 4.5: Comparison of Precision

Comparison of F1 Score

The F1-score, a metric that balances precision and recall, identified VGG16 as the most effective in the 'activity' class, achieving the highest score of 91%, signifying its robust performance in distinguishing umpire gestures. MobileNetV2 followed closely with an F1-score of 88% for 'activity,' demonstrating its efficiency in capturing these pivotal moments. While delivering competitive results, the Custom CNN model achieved an F1-score of 84% for 'activity.' In the 'no activity' class, VGG16 exhibited the highest F1 score at 85%, whereas MobileNetV2 and the Custom CNN model recorded F1 scores of 83% and 76%, respectively. Meanwhile, we Tried the vision transformer on UGID dataset it recorded the F1 score of activity and no activity class is 83% and 74% as shown in the figure 4.6.

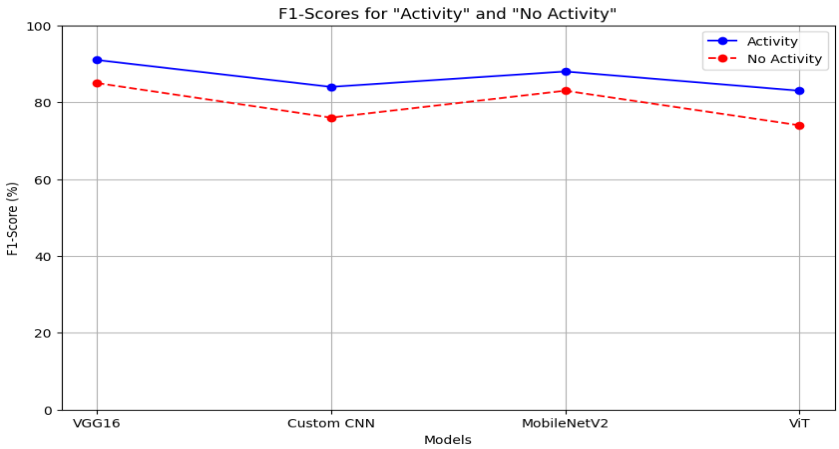


Figure 4.6: Comparison of F1 Score

Comparison of recall

MobileNetV2 has a 90% recall rate for the 'activity' class, demonstrating its capacity to retrieve frames involving umpire gestures. VGG16 came in second with an even greater recall of 97%, highlighting its exceptional performance in detecting these situations. The Custom CNN model had an impressive recall of 89% for 'activity.' When it came to the 'no activity' class, MobileNetV2 had an 80% recall, whereas VGG16 had a slightly lower recall of 78%. For the 'no activity' class, the Custom CNN model obtained the lowest recall of 70%. Meanwhile, the ViT model recall the results of activity class is 89% and 68% of no activity class which is shown in figure 4.7.

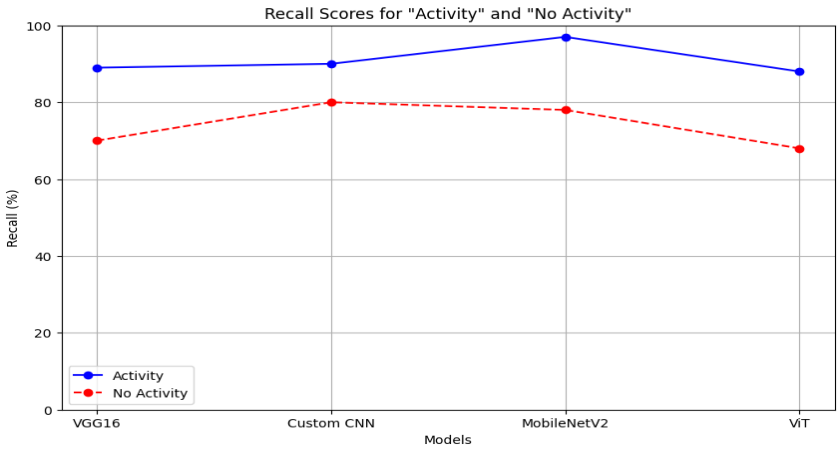


Figure 4.7: Comparison of recall

Comparison of overall accuracy

With regard to the measure of overall accuracy, VGG16 came out on top with an accuracy of 88%, establishing itself as the best performer in this parameter. MobileNetV2 came in second with an overall accuracy of 86%, demonstrating its ability to differentiate between 'activity' and 'no activity' frames. The Custom CNN model has an overall accuracy of 81% and Pre-train Vision transformer has an overall test accuracy of 79% demonstrating its ability to accurately classify frames in cricket match videos. Overall accuracy comparisons of the four models are shown in figure 4.8 highlight the high accuracy to other models in table 4.8. and comparison are visualized in form of a bar chart in figure 4.8

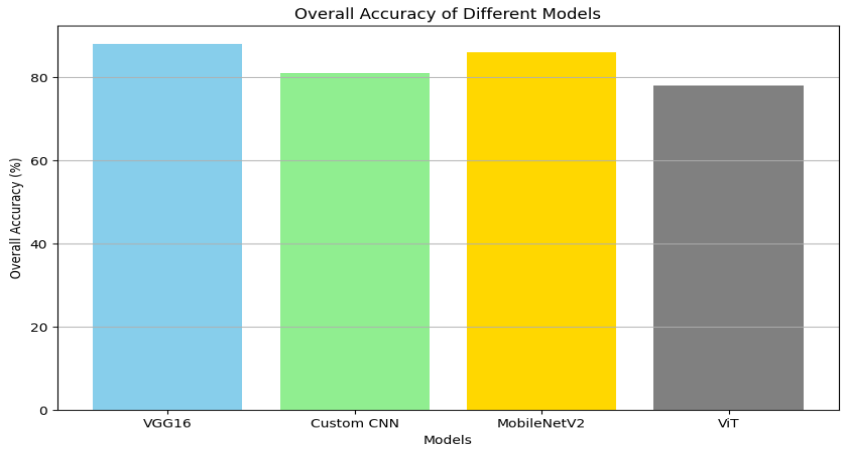


Figure 4.8: Comparison of overall accuracy

Table 4.8: Comparative Analysis of overall accuracy

Model	**LR**	**Batch Size**	**Epochs**	**Dropout**	**Optimizer**	**Accuracy**
VGG16	**0.001**	**16**	**50**	**0.5**	**Adam**	**88**
MobileNetV2	0.001	32	50	0.5	Adam	86
CNN	0.01	32	40	0.5	Adam	81
ViT	0.0001	32	40	0.5	Adam	79

4.2.6 Not Comparing with State-of-the-Art Models

For various convincing reasons, we preferred not to directly compare our results with state-of-the-art models in this study. In the beginning, our study

stumbled into the limitation of only a few datasets created specifically for the task of cricket video summarizing, forcing us to develop a new dataset customized to our research objectives. Second, the lack of standardized, publicly available datasets for cricket video summarizing made meaningful comparisons with existing algorithms difficult. Our primary research goal was to evaluate the effectiveness of our suggested model in the setting of our custom dataset. We intended to investigate the complexity and specifics unique to cricket video summarizing, which may differ from the objectives or characteristics of the data of cutting-edge models created for many different fields or events. As a result, we decided not to compare our method to current models since we recognize the specialized nature of our work and the need to evaluate our technique separately and among different CNN variants. While we recognize the value of benchmarking against state-of-the-art models in different circumstances, we feel that our findings provide useful insights and suggestions for similar tasks or domains.

4.2.7 Activity Prediction for Cricket Video Summarization

An efficient technique is used in the realm of cricket video summarization to predict and classify frames as "Activity" or "No activity," which is essential for identifying important moments in the match. The prediction loop loops through each frame, loading, preprocessing, and running it through a binary classification model. A specified threshold value 0.5 decides whether a frame has "Activity" (for example, umpire gesture) or "No activity" that is shown in the figure 4.9, Results are regularly captured, along with frame identifiers, and counts of "Activity" and "No activity" frames are kept. These forecasts form the basis for creating a cricket summary video, with a concentration on "Activity" frames to generate a brief, interesting glimpse of the cricket match's highlights.

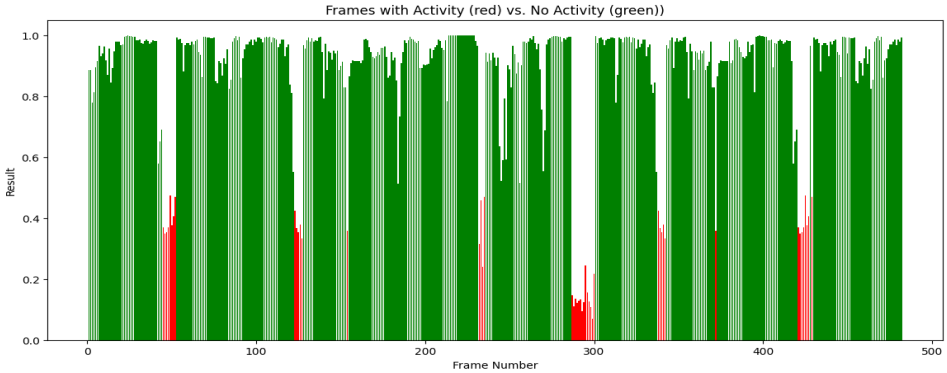


Figure 4.9: Activity prediction example

4.2.8 Activity Prediction results on test data

Prediction: Activity | Confidence Score: 1.4214781458576908e-06

Prediction: Activity | Confidence Score: 0.40458008646965027

Prediction: No Activity | Confidence Score: 0.9709804654121399

Figure 4.10: Prediction with confidence score

CHAPTER 5

CONCLUSION & FUTURE WORK

5.1 Conclusion

We accomplished significant achievements in cricket video summarizing by effectively completing our research objectives in this study. We started by developing an exclusive dataset of umpire gestures for cricket highlights, which allows for more accurate event extraction and summary production. In this study, I experiment with different CNN variants on our custom UGID dataset. Our model performance research revealed that VGG16, MobileNetV2, and Custom CNN were capable of classifying umpire gestures, with VGG16 having the highest level of performance. The significant improvements in overall accuracy, precision, and model performance demonstrated the significance of transfer learning. Additionally, we created a custom cricket video dataset, which addresses the lack of specific cricket video datasets while also preparing the way for future study in cricket video summarization.The lack of a standard dataset for cricket video highlights creation was a significant limitation in our research. There is currently no publicly available dataset, which makes benchmarking and further model enhancement difficult. It's worth noting that the availability of a larger dataset could significantly increase our model's accuracy. If such a dataset is created in the future, it has the potential to dramatically improve the accuracy and robustness of cricket video summarizing systems.

5.2 Future Work

We would like to broaden our research in the future. In addition to our research, we intend to investigate more pre-trained models. Our primary focus will be on considerably growing our umpire gesture dataset, which we believe will increase the accuracy of our techniques. With a larger dataset, we aim to investigate the use of vision transformers for image classification to improve overall accuracy, allowing us to generate more effective cricket summary videos. In addition, we propose to enhance our collection of cricket

video data and develop end-to-end video summarizing methodologies based on vision transformer models. This expansion is essential to ensure that will allow advanced techniques to reach their maximum efficiency. Our goal also indicates an existing lack of standard cricket video datasets, which restricts progress in video summarizing research. In the future, we intend to explore techniques for larger generalization by examining large dataset and comparing them to state of the art models , there by contributing to the continued progress of cricket video summarizing approaches.

REFERENCES

[1] S. Wan, S. Ding, C. Chen, Edge computing enabled video segmentation for real-time traffic monitoring in internet of vehicles, Pattern Recognition 121 (2022) 108146.

[2] S. Ding, S. Qu, Y. Xi, S. Wan, A long video caption generation algorithm for big video data retrieval, Future Generation Computer Systems 93 (2019) 583–595.

[3] H. J. Shingrakhia, Automatic cricket highlight generation using event-driven and excitement-based features using deep learning, Ph.D. thesis, GUJARAT TECHNOLOGICAL UNIVERSITY AHMEDABAD (2023).

[4] R. Summerley, The development of sports: A comparative analysis of the early institutionalization of traditional sports and e-sports, Games and Culture 15 (1) (2020) 51–72.

[5] V. Parikh, J. Mehta, S. Shah, P. Sharma, Comparative analysis of keyframe extraction techniques for video summarization, Recent Advances in Computer Science and Communications (Formerly: Recent Patents on Computer Science) 14 (9) (2021) 2761–2771.

[6] K. A. Tran, O. Kondrashova, A. Bradley, E. D. Williams, J. V. Pearson, N. Waddell, Deep learning in cancer diagnosis, prognosis and treatment selection, Genome Medicine 13 (1) (2021) 1–17.

[7] R. K. Mishra, S. Urolagin, J. A. A. Jothi, P. Gaur, Deep hybrid learning for facial expression binary classifications and predictions, Image and Vision Computing 128 (2022) 104573.

[8] Y. Liu, H. Pu, D.-W. Sun, Efficient extraction of deep image features using convolutional neural network (cnn) for applications in detecting and analysing complex food matrices, Trends in Food Science & Technology 113 (2021) 193–204.

[9] K. R. Raval, M. M. Goyani, A survey on event detection based video summarization for cricket, Multimedia Tools and Applications 81 (20) (2022) 29253–29281.

[10] B. T. Naik, M. F. Hashmi, N. D. Bokde, A comprehensive review of computer vision in sports: Open issues, future trends and research directions, Applied Sciences 12 (9) (2022) 4429.

[11] A. Bhalla, A. Ahuja, P. Pant, A. Mittal, A multimodal approach for automatic cricket video summarization, in: 2019 6th international conference on signal processing and integrated networks (SPIN), IEEE, 2019, pp. 146–150.

[12] N. Harikrishna, S. Satheesh, S. D. Sriram, K. Easwarakumar, Temporal classification of events in cricket videos, in: 2011 National conference on communications (NCC), IEEE, 2011, pp. 1–5.

[13] B. Dange, D. Kshirsagar, H. Khodke, S. Gunjal, Automatic video summarization for cricket match highlights using convolutional neural network, in: 2022 International Conference on Smart Technologies and Systems for Next Generation Computing (ICSTSN), IEEE, 2022, pp. 1–7.

[14] R. Baranwal, Automatic summarization of cricket highlights using audio processing (2021).

[15] S. Nandyal, S. L. Kattimani, Cricket event recognition and classification from umpire action gestures using convolutional neural network, International Journal of Advanced Computer Science and Applications 13 (6) (2022).

[16] M. Nasir, A. Javed, A. Irtaza, H. Malik, M. Mahmood, Event detection and summarization of cricket videos, Journal of Image and Graphics 6 (1) (2018) 27–32.

[17] M. Rafiq, G. Rafiq, R. Agyeman, G. S. Choi, S.-I. Jin, Scene classification for sports video summarization using transfer learning, Sensors 20 (6) (2020) 1702.

[18] D. Gaikwad, S. Sarap, D. Dhande, Video summarization using deep learning for cricket highlights generation., Journal of Scientific Research 14 (2) (2022).

[19] S. H. Emon, A. Annur, A. H. Xian, K. M. Sultana, S. M. Shahriar, Automatic video summarization from cricket videos using deep learning, in: 2020 23rd International Conference on Computer and Information Technology (ICCIT), 2020, pp. 1–6. doi:10.1109/ICCIT51783.2020.9392707.

[20] H. Shingrakhia, H. Patel, Sgrnn-am and hrf-dbn: a hybrid machine learning model for cricket video summarization, The Visual Computer 38 (7) (2022) 2285–2301.

[21] A. Kumar, J. Garg, A. Mukerjee, Cricket activity detection, in: International Image Processing, Applications and Systems Conference, IEEE, 2014, pp. 1–6.

[22] D. Kshirsagar, U. Kulkarni, A generalized neuro-fuzzy based image retrieval system with modified colour coherence vector and texture element patterns, in: 2016 IEEE International Conference on Advances in Electronics, Communication and Computer Technology (ICAECCT), IEEE, 2016, pp. 68–75.

[23] S. Jadon, M. Jasim, Unsupervised video summarization framework using keyframe extraction and video skimming, in: 2020 IEEE 5th International Conference on Computing Communication and Automation (ICCCA), 2020, pp. 140–145. doi:10.1109/ICCCA49541.2020.9250764.

[24] J. Gao, X. Yang, Y. Zhang, C. Xu, Unsupervised video summarization via relation-aware assignment learning, IEEE Transactions on Multimedia 23 (2021) 3203–3214. doi:10.1109/TMM.2020.3021980.

[25] Y. Yuan, J. Zhang, Unsupervised video summarization via deep reinforcement learning with shot-level semantics, IEEE Transactions on Circuits and Systems for Video Technology 33 (1) (2023) 445–456. doi:10.1109/TCSVT.2022.3197819.

[26] T. Liu, Q. Meng, J.-J. Huang, A. Vlontzos, D. Rueckert, B. Kainz, Video summarization through reinforcement learning with a 3d spatio-temporal u-net, IEEE Transactions on Image Processing 31 (2022) 1573–1586. doi:10.1109/TIP.2022.3143699.

[27] M. Gygli, H. Grabner, L. Van Gool, Video summarization by learning submodular mixtures of objectives, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 3090–3098.

[28] R. Panda, A. Das, Z. Wu, J. Ernst, A. K. Roy-Chowdhury, Weakly supervised summarization of web videos, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 3657–3666.

[29] S. Paul, S. Roy, A. K. Roy-Chowdhury, W-talc: Weakly-supervised temporal activity localization and classification, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018.

[30] S. Cai, W. Zuo, L. S. Davis, L. Zhang, Weakly-supervised video summarization using variational encoder-decoder and web prior, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018.

[31] J. Ma, S. K. Gorti, M. Volkovs, G. Yu, Weakly supervised action selection learning in video, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 7587–7596.

[32] T. Badamdorj, M. Rochan, Y. Wang, L. Cheng, Joint visual and audio learning for video highlight detection, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2021, pp. 8127–8137.

[33] Y. Yuan, Y. Lyu, X. Shen, I. W. Tsang, D.-Y. Yeung, Marginalized average attentional network for weakly-supervised learning, arXiv preprint arXiv:1905.08586 (2019).

[34] S. Albawi, T. A. Mohammed, S. Al-Zawi, Understanding of a convolutional neural network, in: 2017 international conference on engineering and technology (ICET), Ieee, 2017, pp. 1–6.

[35] I. Banerjee, Y. Ling, M. C. Chen, S. A. Hasan, C. P. Langlotz, N. Moradzadeh, B. Chapman, T. Amrhein, D. Mong, D. L. Rubin, et al., Comparative effectiveness of convolutional neural network (cnn) and recurrent neural network (rnn) architectures for radiology text report classification, Artificial intelligence in medicine 97 (2019) 79–88.

[36] J. Yu, H. Li, S.-L. Yin, S. Karim, Dynamic gesture recognition based on deep learning in human-to-computer interfaces, Journal of Applied Science and Engineering 23 (1) (2020) 31–38.

[37] S. Sharma, K. Guleria, Deep learning models for image classification: Comparison and applications, in: 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), 2022, pp. 1733–1738. doi:10.1109/ICACITE53722.2022.9823516.

[38] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.-C. Chen, Mobilenetv2: Inverted residuals and linear bottlenecks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 4510–4520.

[39] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556 (2014).

[40] N. Park, S. Kim, How do vision transformers work?, arXiv preprint arXiv:2202.06709 (2022).

MS thesis

ORIGINALITY REPORT

6%	4%	3%	1%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	d-scribes.philhist.unibas.ch Internet Source	<1%
2	onlineresource.ucsy.edu.mm Internet Source	<1%
3	www.waset.org Internet Source	<1%
4	M. Jagadeesh, Rithesh S, Sagar Y. "Cricket Shot Detection Using Deep Learning: A Comprehensive Survey", 2023 International Conference on Networking and Communications (ICNWC), 2023 Publication	<1%
5	arxiv.org Internet Source	<1%
6	scholarworks.utrgv.edu Internet Source	<1%
7	P. Josephin Shermila, Alapati Devi Anusha, Akila. M, Abirami. S. "Pneumonia Detection from X- Ray Images using Convolutional Neural Networks", 2023 Eighth International	<1%

	Conference on Science Technology Engineering and Mathematics (ICONSTEM), 2023 Publication	
8	Vili Podgorelec, Špela Pečnik, Grega Vrbančič. "Classification of Similar Sports Images Using Convolutional Neural Network with Hyper-Parameter Optimization", Applied Sciences, 2020 Publication	<1%
9	Submitted to University of Greenwich Student Paper	<1%
10	research.edgehill.ac.uk Internet Source	<1%
11	Submitted to National University of Singapore Student Paper	<1%
12	technodocbox.com Internet Source	<1%
13	www.tnsroindia.org.in Internet Source	<1%
14	Aman Bhalla, Arpit Ahuja, Pradeep Pant, Ankush Mittal. "A Multimodal Approach for Automatic Cricket Video Summarization", 2019 6th International Conference on Signal Processing and Integrated Networks (SPIN), 2019 Publication	<1%

15	Submitted to Higher Education Commission Pakistan Student Paper	<1%
16	spectrum.library.concordia.ca Internet Source	<1%
17	Christos Tsanas, Erling H. Stenby, Wei Yan. "Calculation of multiphase chemical equilibrium in electrolyte solutions with non-stoichiometric methods", Fluid Phase Equilibria, 2018 Publication	<1%
18	Submitted to Ngan Po Ling College Student Paper	<1%
19	dspace.iiti.ac.in:8080 Internet Source	<1%
20	www.banking.org.za Internet Source	<1%
21	acris.aalto.fi Internet Source	<1%
22	www.mdpi.com Internet Source	<1%
23	core.ac.uk Internet Source	<1%
24	onshow.iadt.ie Internet Source	<1%

25	www.ir.juit.ac.in:8080 Internet Source	<1%
26	Neha Jain, Shishir Kumar, Amit Kumar, Pourya Shamsolmoali, Masoumeh Zareapoor. "Hybrid deep neural networks for face emotion recognition", Pattern Recognition Letters, 2018 Publication	<1%
27	Rabia A. Minhas, Ali Javed, Aun Irtaza, Muhammad Tariq Mahmood, Young Bok Joo. "Shot Classification of Field Sports Videos Using AlexNet Convolutional Neural Network", Applied Sciences, 2019 Publication	<1%
28	downloads.hindawi.com Internet Source	<1%
29	fsktm.um.edu.my Internet Source	<1%
30	spiral.imperial.ac.uk Internet Source	<1%
31	www.biomedcentral.com Internet Source	<1%
32	"Machine Learning and Knowledge Discovery in Databases", Springer Science and Business Media LLC, 2023 Publication	<1%

33	David Martínez Muñoz. "Optimal Deep Learning Assisted Design of Socially and Environmentally Efficient Steel Concrete Composite Bridges under Constrained Budgets", Universitat Politecnica de Valencia, 2023 Publication	<1%
34	Mohammad Shahin, F. Frank Chen, Ali Hosseinzadeh, Neda Zand. "Using Machine Learning and Deep Learning Algorithms for Downtime Minimization in Manufacturing Systems: An Early Failure Detection Diagnostic Service", Research Square Platform LLC, 2023 Publication	<1%

Exclude quotes	On	Exclude matches	< 3 words
Exclude bibliography	On		