REPORT APPROVAL

Date:..........................

The thesis / dissertation / report entitled “Video Summarization using Deep Learning Technique” by Shri. Sourabh Das is approved for accomplishment of degree of Master of Technology in Data Science.

Dr. Ankur Biswas (**Head of the Department,**)	**Signature of External Examiner**

DECLARATION

Date:..........................

I hereby declare that the material, which I am submitting now for the evaluation of study leading to the award of Master of Technology, is totally my own work and nothing has been taken from the work of other people. The portions which has been taken from other person's works has been cited and acknowledged within the text of my own work. This is an original work and has not been submitted to any other institutions for achieving any kind of degree.

....
Shri Sourabh Das
Roll No : 216702011
Registration No : 017925 of 2016 - 2017
Tripura Institute of Technology,
Agartala – 799009, India

CERTIFICATE

This is to certify that the work contained in this thesis entitled “Video Summarization using Deep learning Technique” submitted by Sourabh Das, Registration no : 017925 of 2016–2017, for the degree of Masters in Data Science is a record of original works carried out in the department of Computer Science and Engineering, Tripura Institute of Technology under my guidance and supervision. We further certify that this work has previously not finished in degree. All rules and regulations laid down by the Institute for the fulfillment of requirement for the degree had been followed by him. The work is worthy of consideration for the award of Masters in Data Science.

....
Shri. Gautam Pal
Project Supervisor,
Assistant Professor, CSE Dept.
Tripura Institute of Technology,
Agartala – 799009, India

Acknowledgments

I would like to take this opportunity to express my deep sense of gratitude to all who helped me directly or indirectly during this Project work. Firstly, I would like to thank my supervisor, Gautam Pal (Assistant Professor, CSE Dept. TIT Agartala) for being a great mentor and the best adviser I would ever have. Their advice, encouragement, and critics are source of innovative ideas, inspiration and causes behind the successful completion of this dissertation. The confidence shown on me by them was the biggest source of inspiration for me. It has been a privilege working with my last one year. I was highly obliged to all the faculty members of Computer Science and Engg. Dept. for their support and encouragement. I also thank you Dr. Ankur Biswas (H.O.D, of CSE Dept.) for providing excellent computing and other facilities and without which this work could not achieve its quality goal. Finally, I was grateful to my loving parents, friends for their support. It was impossible for meto complete this project work without their love, blessing and encouragement.

....
Shri Sourabh Das
Roll No : 216702011
Registration No : 017925 of 2016 - 2017
Tripura Institute of Technology,
Agartala – 799009, India

Contents

List of Figures

Abstract

Video summarization involves creating a concise summary of a longer video by selecting and presenting the most relevant or interesting content. Key frame extraction is a crucial aspect of video summarization, offering a quick interpretation of video content. Traditional methods partition the video into clips through short boundary detection and then extract key frames via frame clustering. While various shot boundary detection methods exist, they often face complexity limitations.

This paper introduces a simple yet efficient shot boundary detection method, the KEGMS scheme, with the goal of accelerating the detection process and simplifying it without compromising recall and accuracy. The proposed method utilizes global motion features, specifically global horizontal motion, for fine-grained partitioning of video clips. This leads to more key frames extraction, providing insights into relevant events. Global motion statistics are then employed to identify candidate key frames. The final step involves extracting representative key frames based on spatial-temporal consistency and hierarchical clustering, with redundant frames removed.

The proposed scheme is evaluated using a dataset, and experimental results showcase its state-of-the-art performance, attributed to the integration of global motion statistics.

Chapter 1

INTRODUCTION

In recent times, there has been a notable surge in technological progress. With the increased accessibility and affordability of cost-effective, high-quality video recording devices, the volume of video data has witnessed significant growth. According to a recent survey, YouTube usage has nearly tripled since 2014, with approximately 400 hours of fresh video content uploaded every minute. This data solely pertains to the most popular video search engine; if we were to aggregate information from other sources, such as real-time video feeds from surveillance cameras, the resulting statistics could be even more remarkable. Managing such an extensive amount of data poses a considerable challenge.

One potential remedy to this issue is video summarization, often denoted as video abstraction. Video summarization offers a brief and meaningful representation of a video, streamlining the data processing task. The video can be summarized by either eliminating redundant video content or selecting salient elements. There exist two primary types of video summaries: static summaries and dynamic summaries, illustrated in Figure 1. Static summaries involve selecting significant frames or representative frames of the video, commonly known as key frame extraction. On the other hand, dynamic summaries, or video skims, are segments within the video itself that encapsulate vital content. Video skims hold a higher level of meaning and are akin to movie trailers. They are visually captivating and often provide a more comprehensive understanding of the situation to the viewer. Achieving dynamic video summarization is a intricate task that necessitates distinct modules for handling diverse types of information.

A video constitutes a multimedia sequence of images that may also incorporate audio. For processing efficiency, a lengthy video may be decomposed into smaller segments based on scenes or shots. A scene is a collection of semantically and temporally related elements within a video that collectively convey a higher-level meaning. Various techniques for video summarization are rooted in scene-level decomposition. A shot, defined as a sequence of actions captured by a single camera with minimal alterations in visual content, represents a fundamental concept. Video summarization techniques often rely on identifying shot boundaries marked by abrupt or gradual changes in frames. The representative frame from each shot is then chosen to construct a summary of the entire video.

1.1 Goal

The specific goals of a video summarization project can vary depending on the project's objectives, application domain, and the needs of the intended users. Here are some common goals that a video summarization project might aim to achieve:

- **Content Compression:** Condense lengthy video content into shorter, more manageable summaries while retaining the essential information and key events. This goal aims to provide users with a quick overview without the need to watch the entire video.
- **Information Retrieval:** Enable efficient access to specific information within videos, making it easier for users to find relevant content quickly. This goal is particularly important when dealing with large video databases or archives.
- **User Engagement:** Create summaries that are engaging and capture the audience's attention. This can involve selecting visually compelling frames or clips to make the summary more appealing to viewers.
- **Decision Support:** Help users make informed decisions about whether to invest time in watching the full video. The summary should provide enough information for users to assess the video's relevance to their interests or needs.
- **Application-Specific Focus:** Tailor summarization techniques to meet the requirements of specific applications. For example, summarization goals for surveillance videos may differ from those for educational videos or news broadcasts.
- **Automatic Processing:** Develop algorithms and methods for automated video summarization to streamline the process and make it scalable for large video datasets. Automation is crucial for real-world applications where manual summarization may be impractical.
- **Usability Evaluation:** Conduct usability studies to assess the effectiveness of the generated summaries. This involves evaluating factors such as informativeness, coherence, and user satisfaction to ensure the summaries meet the project's objectives.

- **Adaptability to User Preferences:** Design summarization algorithms that can adapt to different user preferences and needs. Customizable summaries may be more effective in catering to a diverse audience.
- **Integration with Other Systems:** Integrate video summarization capabilities with other systems or applications, such as content management systems, recommendation engines, or search platforms, to enhance overall functionality.
- **Scalability and Efficiency:** Ensure that the video summarization process is scalable and computationally efficient, allowing for the analysis of large volumes of video data in real-time or near real-time scenarios.

These goals collectively contribute to the successful implementation of a video summarization project, addressing the specific requirements and challenges posed by the chosen application domain.

1.2 Motivation

Motivation for a Video Summarization project can be derived from various perspectives, and it's essential to align the motivation with the project's goals, potential benefits, and societal or business needs. Here are several motivational factors for undertaking a Video Summarization project:

1. **Efficient Information Retrieval:**
   - Streamline access to relevant information within videos.
   - Users can quickly find and extract essential content without investing time in watching lengthy videos.
2. **Time-Saving and Convenience:**
   - Provide users with concise video summaries.
   - Users can save time by obtaining key information without watching entire videos, enhancing overall convenience.
3. **Scalability in Video Analysis:**
   - Develop automated algorithms for large-scale video summarization.

- Enables the processing of vast amounts of video data efficiently, which is crucial for real-world applications.

4. **Enhanced User Experience:**
    - Improve the overall user experience in interacting with video content.
    - Users can engage with videos more effectively, leading to increased satisfaction and usability.

5. **Decision Support:**
    - Assist users in making informed decisions about video relevance.
    - Users can quickly determine if a video is worth exploring further based on the summary, aiding decision-making processes.

6. **Adaptability to Various Applications:**
    - Tailor summarization techniques for diverse applications (e.g., news, education, surveillance).
    - Enhances the applicability and usefulness of video summarization across different domains.

7. **Content Representation:**
    - Create concise representations of video content.
    - Provides a more digestible format for conveying the main themes and events within videos.

8. **Usability and User Satisfaction:**
    - Conduct usability studies to ensure the quality of summaries.
    - Ensures that users find the summaries informative, coherent, and satisfying, contributing to overall project success.

9. **Integration with Existing Systems:**
    - Seamlessly integrate summarization capabilities with other systems.
    - Enhances the functionality of existing platforms, such as content management systems or recommendation engines.

10. **Potential for Innovation:**
    - Explore innovative approaches in video analysis and summarization.

- Contributes to the advancement of technology and may lead to new applications or services.

By focusing on these motivational factors, a Video Summarization project aims to address practical challenges, improve user experiences, and offer tangible benefits in terms of time efficiency and decision support. These motivations underscore the significance of video summarization in today's data-rich and time-constrained environments.

1.3 Video Summarization

Video summarization entails creating a concise overview of a lengthy video document by carefully choosing and showcasing the most pertinent or compelling content for potential viewers. The resulting summary typically consists of a series of key frames or edited video clips extracted from the original material. The primary goal of video summarization is to streamline the exploration of extensive video collections, enabling efficient access and representation of the video's content. Through viewing the summary, users can swiftly assess the video's utility and relevance. The evaluation of a summary often includes usability studies to gauge the informativeness and quality of the presented content, a process contingent on the specific applications and target audience. Essentially, video summarization serves as a brief encapsulation of the content within a longer video, catering to the preferences and needs of the intended users.

1.4 Shot Boundary Detection

Shot boundary detection (SBD) is a crucial step in video exploration and retrieval, playing a significant role in content-based video indexing. The primary objective of SBD is to identify the transitions and delineate the boundaries between successive shots. Shots, in this context, refer to consecutive sequences of frames captured in a single take, representing a cohesive viewpoint. This process is essential for effective video analysis and editing, especially in scenarios where quick identification of shot changes is necessary for further content-based applications.

In the realm of video segmentation, the initial phase involves splitting a video into distinct shots, and this is precisely what video shot boundary detection accomplishes. There are two main types of shot boundaries: Cut Transition (CT) and Gradual Transition (GT). CT marks a transition characterized by an abrupt change between two consecutive frames, indicating

a sudden shift in the scene. On the other hand, GT is a more gradual transition that may extend over several frames, sometimes spanning tens of frames. This distinction is crucial for understanding the nature and dynamics of the scene changes within the video.
In simpler terms, shot boundary detection is a process integral to video analysis and editing, serving the purpose of identifying transitions between shots or scenes. Shots, which represent continuous sequences of frames captured without interruption, can vary in their content and viewpoint. Shot boundary detection aids in automatically pinpointing moments where shots undergo changes, including cut transitions and other types of transitions. This automated identification is valuable for efficient video exploration, retrieval, and editing, contributing to the overall enhancement of video-based applications.

1.5 Basics of Video

1. **Digital Image Description:**
   - **First Description:** A digital image is a numerical representation of a two-dimensional function (x, y), consisting of pixels arranged in rows and columns. These numeric values correspond to the intensity levels associated with each pixel.
   - **Second Description:** Digital images are essentially a grid of pixels, forming a numeric representation of a two-dimensional function (x, y). These pixels hold intensity values, collectively composing the visual content of the image.
2. **Video Definition:**
   - Video is a sequence of image frames in motion, where each frame is a still picture.
   - Video is a dynamic series of electronic signals that creates the illusion of motion through a sequence of individual image frames. Each frame represents a static picture, collectively producing the moving visual experience. Purpose of Video:
3. **Purpose of Video:**
   - A video is employed to generate a continuous stream of still images, utilizing electronic signals to simulate movement.
   - In essence, video serves as an electronic medium for recording, copying, broadcasting, and displaying moving visual content. It incorporates graphics, pictures, or text and is applied for entertainment, education, and various other purposes.

These descriptions convey the same fundamental information about digital images and videos, but the language used provides different perspectives on the concepts involved.

The first set of descriptions focuses on the technical aspects of digital images and the sequential nature of video frames, while the second set emphasizes the electronic medium's versatility and its applications in entertainment, education, and beyond.

1.6 Types of Video Summaries

Video summaries can take various forms, catering to different preferences, purposes, and application scenarios. Here are some common types of video summaries:

1. **Key Frame Summaries:**
    - Selects representative key frames from the video.
    - Each key frame captures a significant moment or scene.
2. **Storyboard Summaries:**
    - Presents a sequence of key frames in a storyboard format.
    - Offers a visual overview of the video's main events.
3. **Temporal Summaries:**
    - Focuses on specific time intervals within the video.
    - Highlights important segments or events during those intervals.
4. **Object-Centric Summaries:**
    - Emphasizes specific objects or subjects within the video.
    - Highlights scenes where particular objects are featured prominently.
5. **Event-Centric Summaries:**
    - Centers around significant events or occurrences in the video.
    - Captures key moments that drive the video's narrative.
6. **Motion-Centric Summaries:**
    - Emphasizes dynamic elements and motion within the video.
    - Highlights scenes with notable movement or action.

7. **Speech-Centric Summaries:**
    - Focuses on spoken content, such as dialogue or speeches.
    - Emphasizes scenes with important verbal information.
8. **Textual Summaries:**
    - Converts spoken or visual content into text for summarization.
    - Provides a condensed written overview of the video's content.
9. **Facial Expression Summaries:**
    - Highlights facial expressions and emotions of individuals in the video.
    - Emphasizes scenes with significant emotional impact.
10. **Semantic Concept Summaries:**
    - Summarizes based on specific semantic concepts or themes.
    - Focuses on scenes related to predefined concepts like "happy moments" or "action scenes."
11. **Hierarchical Summaries:**
    - Creates a hierarchical structure of summaries at different levels.
    - Allows users to explore varying levels of detail based on their preferences.
12. **Genre-Specific Summaries:**
    - Tailors the summary format to specific genres (e.g., news, sports, documentaries).
    - Considers the unique features and requirements of different content genres.

The choice of the type of video summary depends on the goals of the summarization task, user preferences, and the nature of the video content. Different types of summaries may be suitable for different applications, ranging from quick content browsing to in-depth analysis.

1.7 Structure of a Videos

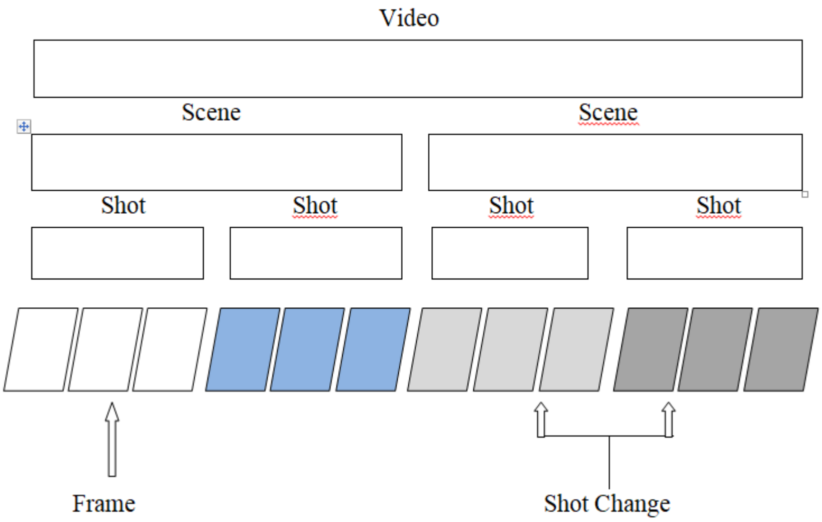


Figure 1.1: **Structure of a video sample1**

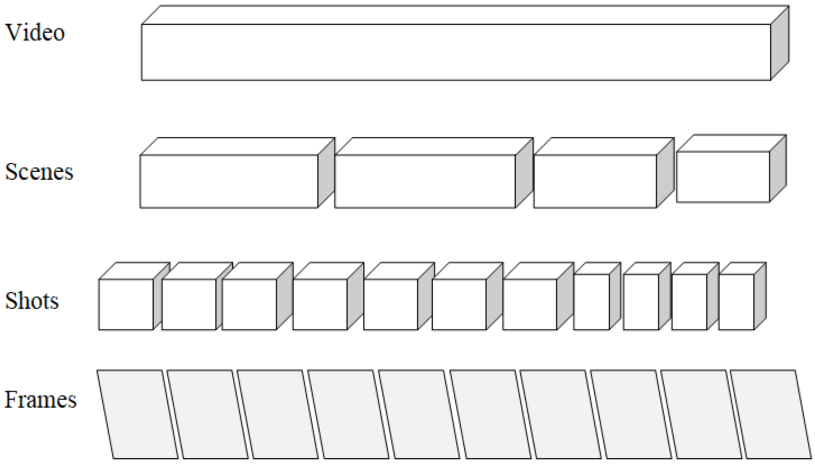


Figure 1.2: **Structure of a video sample2**

1.8 General Description of Shot Boundary Detection

Shot boundary detection (SBD) holds significant importance in the realm of video exploration and retrieval. It specifically targets the identification of transitions and their boundaries between consecutive shots within a video. The essence lies in leveraging shots with substantial information for content-based video indexing and retrieval. As the most fundamental form of temporal video segmentation, SBD is intricately tied to the video production process, making it a natural choice for breaking down a video into more manageable segments.

By detecting shot transitions, the process effectively divides a video into basic temporal units known as shots. A shot, in this context, comprises a series of sequentially captured frames by a single camera, encapsulating a continuous action over time and space. This operation is highly beneficial in post-production video editing and is a foundational step in various video analysis tasks.

The significance of shot boundary detection extends to automated indexing, content-based video retrieval, and summarization applications. Constructing a video index referring to whole shots rather than individual frames enhances efficiency and usability. This process becomes indispensable as the initial step in boundary detection since it effectively marks the points where shots undergo changes, encompassing various transitions such as cuts.

The technology employed in news videos over the past few decades has significantly increased the volume of video data available on the web. News videos, with their effective reporting on global events, especially in politics and local news, have gained attention from both researchers and audiences. In this context, the detection of human faces emerges as a crucial area of research, impacting applications like face recognition, facial expression recognition, tracking, gender classification, and more.

In essence, shot boundary detection remains a pivotal process in video analysis and editing, serving the purpose of identifying transitions between shots or scenes. These shots, consisting of consecutive frames captured in a single take, represent a continuous viewpoint. This automated identification aids in locating points where shots change, encompassing various transitions such as cuts and other types.

Chapter 2

Related Work

The extraction of key frames stands as a foundational task in the realm of video analysis and summarization. This process serves as a potent tool in encapsulating video content by carefully selecting a subset of summary key frames that effectively represent the entire video sequence. The established methodologies typically consist of a two-step approach. Initially, the video undergoes segmentation into distinct video clips through shot boundary detection. Subsequently, key frames are culled from each video clip, with the utilization of various types of features.

A diverse array of features is commonly employed in this process, encompassing visual, motion, deep, histogram, color histogram difference, and local ternary pattern features. These features collectively contribute to the robust extraction of key frames, ensuring that the essence of the video is accurately captured. Through the incorporation of these features, the key frame extraction process becomes a nuanced and comprehensive approach to summarizing video content, allowing for efficient representation and analysis of diverse video sequences.

2.1 Visual feature:

A visual feature encapsulates a unique property inherent in an entire image or an object contained within the image, and it may manifest as a local attribute or a global characteristic of the image. Visual features serve as discernible elements or attributes extracted from visual data, spanning images or videos. These features delineate specific facets of the visual content and find application in various domains, including computer vision, image processing, and machine learning. They encompass diverse characteristics such as color information, texture patterns, edges, shapes, and more intricate attributes like key points of objects.

The role of visual features is pivotal in aiding algorithms to comprehend and distinguish between various objects, scenes, or patterns within visual data. In the context of video analysis, visual features play a crucial role in tasks such as shot boundary detection. For instance, in the study by Zhou et al. in 2021, video shot boundary detection is accomplished through the collaborative utilization of multi-level visual features. This approach highlights the significance of integrating various visual characteristics to enhance the precision and effectiveness of video analysis algorithms.

2.2 Motion feature:

Motion features encompass attributes that describe the characteristics or patterns of movement in various domains, such as computer vision, machine learning, and animation. These features play a crucial role in detailing how objects or entities navigate through both space and time. Parameters integral to motion features include aspects like speed, direction, acceleration, rotation, and trajectory. In applications like computer vision and machine learning, motion features are often extracted from video data to comprehend and analyze the dynamic behavior of objects.

The process of describing motion features typically involves dividing each video into multiple blocks, each of size N × M × L. This segmentation approach entails scaling each optical flow matrix to a size of N × M, with each block containing L optical flow instances. This strategy allows for a comprehensive representation of motion features, capturing the nuances of how objects move within the video.

In the context of video shot boundary detection, the study conducted by Zhou et al. in 2021 employs motion features as part of a multi-level feature collaboration approach. This emphasizes the utilization of various features, including motion, to enhance the efficacy of shot boundary detection algorithms. By integrating motion features at multiple levels, the collaborative approach proposed by Zhou and colleagues aims to improve the accuracy and robustness of the shot boundary detection process.

2.3 Deep feature

Deep features are numerical descriptors often derived from a Convolutional Neural Network (CNN), extensively employed for tasks such as classification and recognition. These features encapsulate information pertaining to texture and shape, primarily. Deep features represent abstract and high-level representations of data, particularly in the context of image data, where they are extracted using deep learning models, notably Convolutional Neural Networks.

The term "deep features" denotes the capability of these descriptors to offer intricate and sophisticated representations, thanks to the hierarchical learning achieved by deep neural networks. This learning process allows deep features to capture complex patterns and nuances within the data automatically. The use of deep features extends across diverse domains, including computer vision, natural language processing, and speech recognition. This widespread application is attributed to the innate ability of deep features to autonomously learn pertinent and meaningful representations from raw data.

In the study conducted by Zhou et al. in 2021, deep features play a pivotal role in the context of video shot boundary detection. The approach proposed by Zhou and colleagues involves the collaborative use of multi-level features, where deep features contribute to the intricate understanding of video content, enhancing the effectiveness of shot boundary detection algorithms.

2.4 Histogram feature

A histogram is a visualization tool that partitions the potential values within a dataset into distinct classes or groups. Each group corresponds to a range of values, and for each group, a rectangle is constructed. The base length of the rectangle equals the range of values within that specific group, and the height equals the number of observations falling into that group. Essentially, a histogram is a graphical representation that illustrates the frequency distribution of numerical data through the use of rectangles.

The graph provides a visual depiction of how frequently different values occur within the dataset. The height of each rectangle in the histogram reflects the frequency or count of observations falling within the respective range. This method of representation aids in understanding the distribution of a variable, with taller rectangles indicating higher frequency in certain value ranges.

In the context of the features utilized in the work by Chen in 2022, particularly in sports video panorama synthesis technology based on edge computing and video shot boundary detection, histograms may play a role in capturing and visualizing certain characteristics of the data, contributing to the analysis and synthesis processes.

2.5 Color Histogram Difference feature

A histogram difference is a metric that exhibits lower sensitivity to motion, proving to be a effective measure for gauging similarity between images. This method involves identifying substantial alterations in the weighted color histogram of two images, culminating in a more resilient measure for establishing image correspondence. In the context of features employed in the work by Chakraborty and colleagues in 2022, particularly in their ALO-SBD (A Hybrid Shot Boundary Detection) technique designed for video surveillance systems, the utilization of histogram difference emerges as a crucial element. This approach enhances the robustness of shot boundary detection by leveraging the discerning capabilities of histogram differences, particularly in scenarios where motion sensitivity needs to be minimized for accurate detection in surveillance videos.

2.6 Local ternary pattern feature

The Local Ternary Pattern (LTP) builds upon the foundation laid by the Local Binary Patterns (LBP), introducing a departure from the traditional binary thresholding approach. Unlike LBP, which classifies pixels into binary values of 0 and 1, LTP takes a step further by employing a threshold constant that categorizes pixels into three distinct values.

In the work by Chakraborty, Singh, and Thnaojam in 2022, their innovative bi-fold-stage shot boundary detection algorithm, designed to be invariant to both motion and illumination, incorporates the Local Ternary Pattern (LTP) as a pivotal element. This integration of LTP serves to enhance the algorithm's capabilities, ensuring robust shot boundary detection in video sequences.

The use of LTP in this context proves crucial for achieving resilience against variations in both motion and illumination. By incorporating a ternary classification of pixels, the algorithm becomes more adept at handling diverse and challenging conditions in video sequences, ultimately contributing to the algorithm's effectiveness in detecting shot boundaries with improved accuracy and robustness.

Chapter 3

Literature Review

Video summarization in the digital world faces challenges due to inefficiencies in processing long-duration videos using existing deep learning methods. Extensive analysis of these methods reveals shortcomings in identifying and summarizing essential activities. Key issues include event detection, categorization, and activity feature summarization. Limitations related to each category are discussed, along with concerns about detecting low activity using deep networks on public datasets. The paper suggests viable strategies for evaluating and improving video summaries and outlines potential applications based on the literature. Additionally, it discusses deep learning tools for experimental analysis and presents future directions for further research in video summarization using deep learning[1].A novel video shot boundary detection algorithm, based on feature fusion and clustering technique (FFCT), addresses issues of low accuracy and high complexity in detecting gradual shot boundaries and long shots. The algorithm involves selecting interval frames, converting them to gray images, and scaling through sampling. Speed-up robust features (SURF) and fingerprint features are extracted from both non-compressed and compressed domains, fused, and clustered using K-means. Linear discriminant analysis (LDA) enhances cohesion within classes and looseness among classes. The algorithm achieves superior accuracy in coarse and fine detection of shot boundaries, especially for gradual and long shots, outperforming the latest representative algorithms. Additionally, it reduces average time consumption, demonstrating high accuracy and efficiency, particularly in challenging scenarios.[2].This article introduces a unified framework for Shot Boundary Detection (SBD) in video content analysis, addressing both Abrupt Transitions (AT) and Gradual Transitions (GT). The method utilizes multiscale geometric analysis of the Non-Subsampled Contourlet Transform (NSCT) for feature extraction from video frames. Principal Component Analysis (PCA) is employed to reduce the dimension of the feature vectors, enhancing computational efficiency and performance. A cost-efficient Least Squares Support Vector Machine (LS-SVM) classifier categorizes frames into No-Transition (NT), AT, and GT classes based on the extracted features. The article proposes an efficient method for training set generation, reducing training time while improving performance. Empirical results on TRECVID 2007 and TRECVID 2001 test data demonstrate the effectiveness of the proposed algorithm compared to state-of-the-art SBD methods.[3].This study introduces a bifold-stage approach for robust shot boundary detection in videos, addressing challenges posed by unforeseen illumination changes and motion effects. In the initial stage, local ternary patterns feature extraction is employed on each frame, utilizing novel adaptive thresholds γ and β to identify potential transition frames. The confirmation stage employs Lab color difference with an adaptive threshold δ to extract true transition frames, effectively handling both illumination and motion effects. Experimental results using TRECVid 2001 and 2007 datasets demonstrate the proposed technique's superiority over contemporary shot boundary detection methods, showcasing its effectiveness in mitigating motion effects in the initial stage and addressing illumination and motion challenges in the confirmation stage[4].This paper introduces a streamlined approach for video shot boundary detection (SBD) in content-based video retrieval. The proposed method aims to enhance the efficiency of boundary detection while maintaining high recall and accuracy. Key components include a top-down zoom rule, image color features, and local descriptors, combined with a motion area extraction algorithm. Candidate transition segments are selected using color histogram and speeded-up robust features, followed by cut transition detection through uneven slice matching, pixel difference, and color histogram. Gradual transition detection is performed using motion area extraction, scale-invariant feature transform, and even slice matching. Evaluation on TRECVid2001 and TRECVid2007 datasets demonstrates improved recall, accuracy, and detection speed compared to other SBD methods[5].This study introduces a novel Shot Boundary Detection (SBD) method employing the Ant Lion Optimizer (ALO), a nature-inspired algorithm. The ALO is utilized to optimize weights in a Feed-Forward Neural Network (FNN), enhancing system performance. The approach incorporates a hybrid technique involving a continuity matrix (ϕ) and an outlier to identify potential transition frames. Actual transition frames are then extracted using a threshold $\delta 1$. Experimentation with challenging videos from TRECVid 2001 and 2007 datasets demonstrates that ALO-SBD achieves superior performance in terms of F1 score, surpassing recent state-of-the-art techniques in the field of SBD [6].This paper addresses the development requirements for sports panorama synthesis technology, emphasizing the utilization of modern network technology and a departure from traditional basketball training methods. The incorporation of video panorama technology in training, facilitated by sports video analysis, responds to the evolving needs of students' physical education and mental growth. The focus is on shot segmentation, a crucial element for hierarchical video structure, necessitating accurate detection of various shot boundaries and effective differentiation of motion changes. The study explores sports video panorama synthesis through edge computing and shot boundary detection, utilizing boundary features to compare against predetermined thresholds, enabling the identification of shot shearing for comprehensive action freezing and analysis[7]. Importance of content-based video retrieval (CBVR) in the context of multimedia and technological advancements. Emphasizing the need for efficient indexing and retrieval of specific points of interest rather than entire videos, the primary focus is on shot boundary detection—a crucial step in CBVR. The segmentation of videos into shots is highlighted as essential for streamlined indexing and retrieval processes. The study underscores the pivotal role of segmentation in digital media processing, pattern recognition, and computer vision. Various approaches to addressing the shot boundary detection problem are presented, offering insights into methodologies aimed at achieving effective video segmentation for improved content-based retrieval. The proliferation of online videos is attributed to advanced multimedia devices, robust communication technologies, and affordable storage options. Current video storage practices, primarily through text annotation in databases, hinder efficient content-based video browsing and retrieval. The large size and extensive information in video databases necessitate automated video structure analysis. Shot Boundary Detection (SBD) emerges as a critical component in this process, aiming to identify transitions and boundaries between consecutive shots for effective content-based video indexing and retrieval. This paper provides a thorough review of various SBD approaches, examining their advantages, disadvantages, and developed algorithms. The analysis also highlights challenges and offers recommendations in this domain[8]. A novel video shot boundary detection algorithm, the Feature Fusion and Clustering Technique (FFCT), addresses the challenges of low accuracy and high complexity in detecting gradual shot boundaries and long shots. The algorithm selects interval frames, converts them to gray images, and scales them through sampling. Extracting Speed-Up Robust Features (SURF) and fingerprint features from both non-compressed and compressed domains, the algorithm fuses these features. Utilizing K-means clustering and linear discriminant analysis (LDA), the fused features are grouped for improved cohesion and separation. Correlation calculation between feature classes enables coarse and fine detection of video shot boundaries. Experimental results demonstrate higher accuracy, particularly in gradual shot boundary and long shot detection, with reduced average time consumption compared to recent algorithms, showcasing the proposed algorithm's effectiveness and efficiency[9].A two-stage method for shot boundary detection (TSSBD) to accurately identify both abrupt and gradual shot transitions in videos. The first stage employs color histogram and deep features fusion to detect abrupt shot changes between frames. Subsequently, a C3D-based deep analysis is utilized to locate gradual shot changes within video segments. The method effectively classifies clips into specific gradual shot change types. An innovative merging strategy is proposed to precisely determine the positions of gradual shot transitions. The experimental analysis demonstrates the TSSBD's capability to detect both abrupt and gradual shot transitions with high accuracy, addressing the complexities arising from varying shot lengths and content variations in diverse videos[10].

Chapter 4

Methodology

Methodology encompasses the comprehensive strategy and rationale behind my research project, involving an examination of methods and underlying theories in the field. The aim is to develop an approach aligned with the project's objectives. Research methodology is a systematic means of addressing the research problem, regarded as the science of studying how research is conducted scientifically. It involves scrutinizing the steps typically undertaken by a researcher along with the underlying logic. Specifically, research methodology pertains to the procedures and techniques employed to identify, select, process, and analyze information about a topic. In a research paper, the choice of methodology enables readers to assess the overall validity and reliability of the study. Another perspective on research methodology views it as the systematic process for designing, conducting, and analyzing research studies. This entails the selection of appropriate methods, techniques, and tools for data collection and interpretation. A well-defined research methodology is essential for generating credible and meaningful results across various fields, spanning sciences to social sciences and humanities.

4.1 Proposed Frame work of KEGMS

In this study, the chosen approach is the KEGMS scheme, selected for its superior performance compared to other methods. The KEGMS method stands out for its efficiency, swiftly identifying crucial key frames within extended video segments. To elaborate on its functionality, the method initiates with shot boundary detection, segmenting the video into distinct clips. These clips are then categorized into game shots and non-game shots, relying on the RGB color model. Non-relevant clips, such as close-ups and advertisements, are subsequently eliminated. Next, the optical flow of video frames is utilized to estimate global motion, leading to further partitioning of video clips based on changes in horizontal global motion. The global motion features, encompassing translation distance and zooming magnitude, are then compared. Finally, representative key frames are extracted, culminating in the formation of the definitive key set. The schematic representation of the KEGMS scheme is illustrated in the accompanying figure.

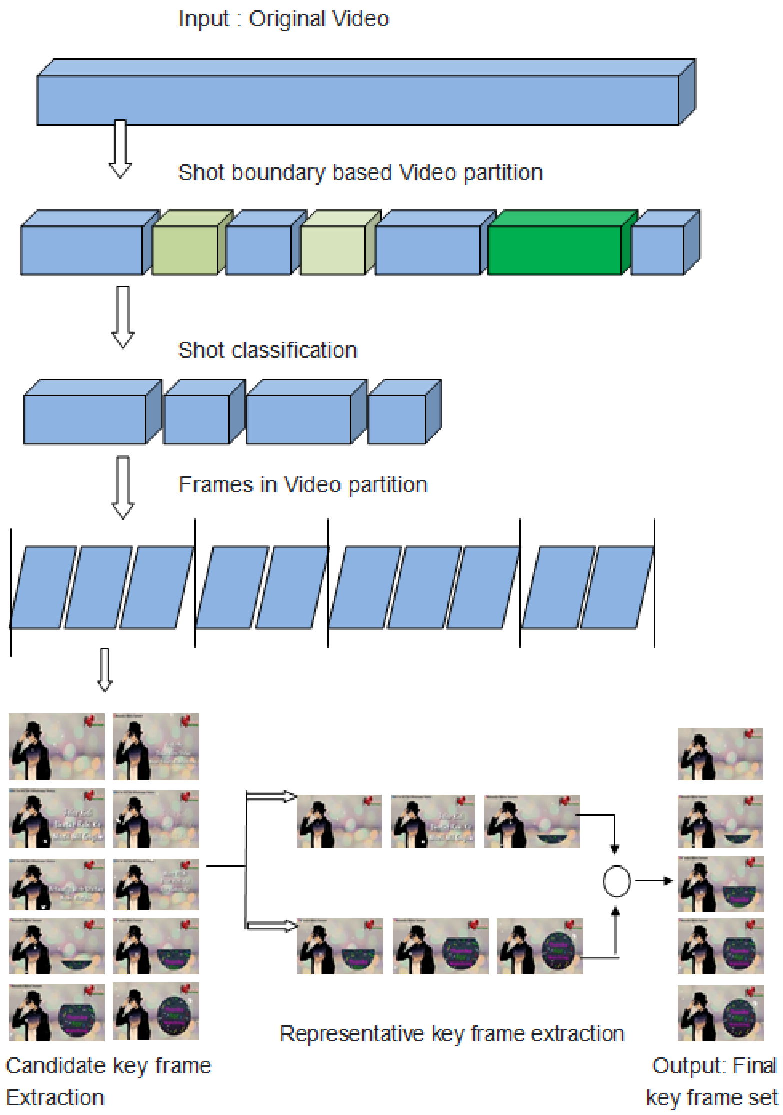

Input : Original Video
Shot boundary based Video partition
Shot classification
Frames in Video partition
Candidate key frame Extraction
Representative key frame extraction
Output: Final key frame set

Figure 4.1: **Framework of the proposed KEGMS**

Video Summarization Objective

Video summarization serves the key objective of condensing extensive video content into a more concise and representative form. The primary goals include:

1. **Content Compression:** Compressing lengthy videos into shorter versions, aiding in storage, transmission, and analysis.
2. **Information Retention:** Retaining the most relevant and informative content to quickly convey key events and actions.
3. **User Engagement:** Enhancing user engagement by providing a quick overview, allowing users to decide if the content aligns with their interests.
4. **Applications in Retrieval and Analysis:** Supporting efficient content retrieval and analysis in fields such as surveillance and research.
5. **Adaptability:** Creating summaries adaptable to different genres, content types, and user preferences for widespread applicability.

The overarching objective of video summarization is to provide an effective and concise representation of video content, addressing information overload challenges and enhancing usability across various domains.

4.2 RESEARCH METHODS

4.2.1 Color Histograms

Color histograms provide a quantitative representation of the distribution of colors within an image. Mathematically, a color histogram is constructed by dividing the color space into discrete bins and counting the number of pixels that fall into each bin. For a typical RGB color space, the three channels (Red, Green, and Blue) are quantized into a predefined number of bins. Let N represent the total number of bins, and n index each bin.

Mathematically, the color histogram H is defined as:

$$H = \{h_n | n = 1, 2, ..., N\}$$

where h_n represents the frequency or count of pixels falling into the n-th bin. The histogram provides insights into the prevalence of different colors in the image. This information can be utilized for various image processing tasks, including shot boundary detection.

To extract a color histogram from an image, each pixel's color values are examined, and the corresponding bin indices are incremented. This process is formalized as:

$$h_n = \sum_{i=1}^{M} \sum_{j=1}^{N} \delta(c_i, j, n)$$

where M and N are the image dimensions, c_i represents the color value of the pixel at position (i, j), and $\delta(\cdot)$ is the Kronecker delta function, indicating whether the pixel's color falls into the n-th bin.

The resulting color histogram is a concise representation of the image's color distribution, capturing the prominence of different hues. Such histograms play a crucial role in content-based image retrieval and shot boundary detection algorithms, where the comparison of color distributions aids in identifying significant changes between consecutive frames.

4.2.2 Texture Features

Texture features in image processing are mathematical descriptors capturing patterns and variations in pixel intensities, providing valuable information about the local structure of an image. A common method for texture analysis involves Local Binary Patterns (LBP). For a grayscale image, the LBP operator examines the relationship between the intensity of a central pixel and its surrounding neighbors within a predefined neighborhood. Mathematically, the LBP value for a pixel at coordinates (x, y) is calculated as:

$$LBP_{P,R}(x, y) = \sum_{p=0}^{P-1} s(g_p - g_c) \cdot 2^p$$

where g_p and g_c represent the intensities of the neighboring pixels and the central pixel, respectively. The function $s(x)$ is defined as 1 if $x \geq 0$ and 0 otherwise. The parameters P and R denote the number of neighbors and the radius of the circular neighborhood, respectively.

The resulting LBP values form a binary pattern that characterizes the texture around the central pixel. This approach provides a robust representation of texture patterns, facilitating tasks such as texture-based image classification and segmentation. Other texture features, such as Gabor filters, may also be employed for capturing more complex textures based on frequency and orientation characteristics.

4.2.3 Deep Learning Embeddings

Deep learning embeddings are representations of data learned by neural networks, capturing intricate patterns and hierarchical features. In the context of image processing, convolutional neural networks (CNNs) are often employed to generate informative embeddings. Let CNN(x) represent the output of a CNN for an input image x. The embeddings, denoted as $E(x)$, are typically extracted from the intermediate layers of the network.

Mathematically, the embedding extraction process can be expressed as:

$$E(x) = \text{CNN}(x)$$

where CNN(x) encompasses the activations from specific layers. The power of deep learning embeddings lies in their ability to capture complex visual features, such as edges, textures, and object parts, as well as high-level semantic information.

These embeddings can be leveraged for various computer vision tasks, including image recognition, object detection, and content-based image retrieval. The learned representations enable the model to discern and differentiate between intricate visual nuances, making deep learning embeddings a pivotal component in modern image analysis and understanding.

4.3 Motion Features

Motion features in video processing play a crucial role in capturing the dynamics and changes between consecutive frames. Optical flow is a common technique used to quantify motion within a sequence of frames. Let $I_t(x, y)$ represent the intensity of a pixel at coordinates (x, y) in frame t. The optical flow, denoted as (u, v), represents the pixel's displacement in the horizontal (u) and vertical (v) directions.

Mathematically, optical flow is computed as:

$$I_x u + I_y v + I_t = 0$$

where I_x and I_y are the spatial gradients of intensity, and I_t is the temporal gradient.

Motion features can be derived from optical flow, such as the overall motion vector, speed, and direction. Additionally, frame differencing, involving subtracting consecutive frames, provides insights into pixel-wise changes between frames.

These motion features are invaluable for tasks like action recognition, video segmentation, and shot boundary detection. They contribute to a comprehensive understanding of temporal dynamics within video sequences, enabling the extraction of meaningful information related to object movements and scene changes.

4.3.1 Statistical Measures

Statistical measures serve as essential tools in image processing to characterize pixel intensity distributions and overall image properties. Let $I(x, y)$ represent the intensity of a pixel at coordinates (x, y) in an image.

1. **Mean (μ):** The mean is a measure of central tendency and is computed as the average pixel intensity across all pixels in the image:

$$\mu = \frac{1}{N} \sum_{x=1}^{M} \sum_{y=1}^{N} I(x, y)$$

where N is the total number of pixels in the image.

2. **Variance (σ^2):** Variance quantifies the spread of pixel intensities around the mean and is calculated as:

$$\sigma^2 = \frac{1}{N} \sum_{x=1}^{M} \sum_{y=1}^{N} [I(x, y) - \mu]^2$$

3. **Skewness (γ):** Skewness measures the asymmetry of the intensity distribution and is given by:

$$\gamma = \frac{1}{N} \sum_{x=1}^{M} \sum_{y=1}^{N} \left[\frac{I(x, y) - \mu}{\sigma} \right]^3$$

4. **Kurtosis (κ):** Kurtosis reflects the tail behavior of the intensity distribution and is computed as:

$$\kappa = \frac{1}{N}\sum_{x=1}^{M}\sum_{y=1}^{N}\left[\frac{I(x,y)-\mu}{\sigma}\right]^4 - 3$$

These statistical measures offer insights into the overall characteristics of the image intensity distribution, aiding tasks such as image segmentation and quality assessment.

4.3.2 Edge Detection

Edge detection is a fundamental process in image processing that focuses on identifying abrupt changes in intensity, representing significant transitions in the image structure. One popular method for edge detection is the Canny edge detector.

Let $I(x, y)$ denote the intensity of a pixel at coordinates (x, y) in the image. The Canny edge detector operates by computing the gradient magnitude (G) and orientation (θ) at each pixel, where the gradient is calculated as:

$$G(x,y) = \sqrt{(I_x)^2 + (I_y)^2}$$

with I_x and I_y being the spatial gradients in the horizontal and vertical directions, respectively. The orientation (θ) of the gradient is given by:

$$\theta(x,y) = \arctan\left(\frac{I_y}{I_x}\right)$$

Edges are then identified by thresholding the gradient magnitude and applying non-maximum suppression to retain only local maxima along the detected edges.

The Canny edge detection algorithm can be expressed mathematically as:

$$E(x,y) = \begin{cases} 1 & \text{if } G(x,y) > \text{High threshold} \\ 0 & \text{if } G(x,y) < \text{Low threshold} \\ \text{interpolate} & \text{if Low threshold} < G(x,y) < \text{High threshold} \end{cases}$$

The resulting binary image $E(x, y)$ highlights the edges in the original image, providing a basis for further analysis and feature extraction.

Edge detection is fundamental for tasks such as object recognition, image segmentation, and pattern analysis in computer vision.

4.3.3 Local Descriptors

Local descriptors play a vital role in image processing by capturing distinctive patterns and features within specific regions of an image. SIFT (Scale-Invariant Feature Transform) and SURF (Speeded-Up Robust Features) are popular local descriptors known for their robustness against changes in scale and orientation.

Let $I(x, y)$ denote the intensity of a pixel at coordinates (x, y) in the image. For a local region around a key point, SIFT extracts features based on gradient magnitudes and orientations, forming histograms of gradient orientations. Mathematically, the SIFT descriptor D for a key point is represented as a vector:

$$D = \{d_1, d_2, ..., d_N\}$$

where d_i corresponds to the i-th bin in the gradient orientation histogram.

Similarly, SURF computes local descriptors using integral images to accelerate the process. Let $H(x, y)$ represent the integral image. The SURF descriptor D for a key point is calculated as:

$$D = \{d_1, d_2, ..., d_N\}$$

where d_i is the sum of Haar wavelet responses within specific subregions.

These local descriptors provide a compact representation of local image features, making them invaluable for tasks such as object recognition, image matching, and image stitching. The robustness and distinctiveness of these descriptors contribute to their widespread use in computer vision applications.

4.3.4 Spatial Information

Spatial information in image processing refers to the arrangement and relationships of pixels in an image's spatial domain. It plays a crucial role in understanding the structure, layout, and patterns within an image. Spatial information encompasses concepts like spatial resolution, spatial frequency, and spatial relationships.

1. **Spatial Resolution:** Spatial resolution refers to the level of detail captured in an image. Mathematically, for an image with dimensions M (width) and N (height), the spatial resolution is often defined as the number of pixels per unit distance:

$$\text{Spatial Resolution} = \frac{M}{\text{Width}} \times \frac{N}{\text{Height}}$$

2. **Spatial Frequency:** Spatial frequency characterizes the rate of change of pixel intensities across space. It is often analyzed using Fourier Transform techniques. The spatial frequency (f) can be calculated as the reciprocal of the wavelength (λ):

$$f = \frac{1}{\lambda}$$

3. **Spatial Relationships:** Spatial relationships involve understanding how pixels or regions relate to each other in terms of distance, adjacency, or connectivity. This information is crucial for tasks like segmentation and object recognition.

Spatial information is fundamental for a wide range of image processing tasks, influencing decisions related to image quality, feature extraction, and spatial transformations. Understanding and manipulating spatial information are essential steps in the analysis and interpretation of digital images.

4.3.5 Temporal Information

Temporal information in video processing refers to the dynamics and changes that occur over time within a sequence of frames. Understanding temporal aspects is essential for tasks such as motion analysis, action recognition, and video summarization.

1. **Frame Rate (f_r):** Frame rate is a key parameter representing the number of frames per second in a video. Mathematically, for a video with T frames and duration D in seconds,

the frame rate is given by:

$$f_r = \frac{T}{D}$$

2. **Temporal Resolution:** Temporal resolution quantifies the ability to capture changes over time. It is often defined as the reciprocal of the time interval between consecutive frames:

$$\text{Temporal Resolution} = \frac{1}{\text{Time Interval}}$$

3. **Motion Information:** Temporal information is closely tied to motion analysis. Optical flow and motion vectors are used to describe the movement of objects between frames, providing insights into dynamic changes.

Temporal information is crucial for tasks like shot boundary detection, tracking moving objects, and recognizing temporal patterns. The proper analysis of temporal dynamics enhances the understanding of video content and supports various applications in video processing and computer vision.

4.3.6 Normalization and Scaling

Normalization and scaling are fundamental preprocessing techniques in data analysis and machine learning to ensure that features or variables have comparable scales and follow a standardized distribution. These techniques are essential for preventing certain features from disproportionately influencing the model due to their larger magnitudes.

1. **Normalization:** Normalization involves transforming the values of a variable to a standard scale, typically between 0 and 1. The formula for normalization is given by:

$$x_{\text{normalized}} = \frac{x - \text{min}(X)}{\text{max}(X) - \text{min}(X)}$$

where x is the original value, min(X) is the minimum value in the dataset, and max(X) is the maximum value.

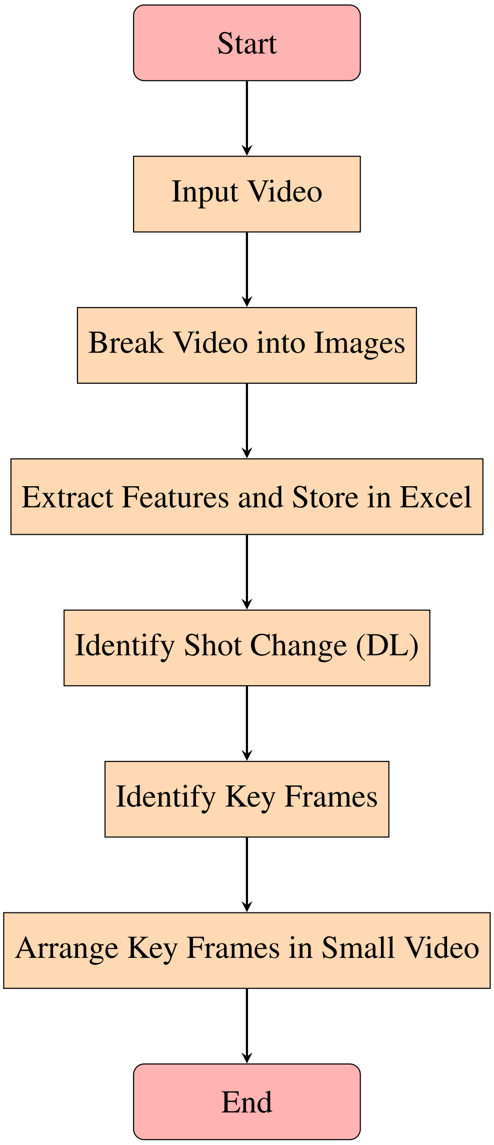
2. **Scaling:** Scaling aims to standardize the range of different variables. Common scaling methods include z-score scaling, where each value is transformed to have a mean of 0 and a standard deviation of 1:

$$x_{\text{scaled}} = \frac{x - \text{mean}(X)}{\text{std}(X)}$$

where mean(X) is the mean and std(X) is the standard deviation of the dataset.

These techniques are crucial for machine learning models that are sensitive to the scale of features. By normalizing or scaling the data, models become more robust and performant across diverse datasets, ensuring fair consideration of all features in the learning process.

4.4 Proposed Algorithm


Start
Input Video
Break Video into Images
Extract Features and Store in Excel
Identify Shot Change (DL)
Identify Key Frames
Arrange Key Frames in Small Video
End

4.5 Algorithm Expaltion

Step 1: Input any Video

This step involves loading a video file, denoted as V.

Step 2: Break the Video into Sequence Images

Let I_t represent the t-th frame in the video sequence. The video can be represented as a sequence of images: $V = [I_1, I_2, ..., I_T]$, where T is the total number of frames.

Step 3: Extract Necessary Features and Store in Excel Sheet

Let F_t represent the features extracted from the t-th frame. The set of features extracted from all frames is denoted as $F = [F_1, F_2, ..., F_T]$. These features could include:

- Color Histograms: Describing the distribution of color intensities.
- Motion Vectors: Representing the movement of objects between frames.
- Texture Features: Describing spatial variations in pixel intensities.
- Deep Learning Embeddings: Extracted from pre-trained models, such as ResNet or a custom video summarization model.

Each F_t is stored in an Excel sheet.

Step 4: Identify Shot Changes using Deep Learning

Utilize a deep learning model to identify shot changes. Let S_t be a binary variable indicating whether a shot change occurred at frame t. The identification process can be expressed as a function:

$$S_t = \text{IdentifyShotChange}(F_t)$$

Step 5: Identify Key Frames from Each Shot

For each identified shot, select one key frame. Let K_t represent the key frame selected from the t-th shot. The key frame selection can be expressed as a function:

$$K_t = \text{SelectKeyFrame}(I_{t1}, I_{t2}, ..., I_{tN})$$

Where $t1, t2, ..., tN$ are the frames within the t-th shot.

Step 6: Arrange Key Frames into a Summarized Video

Create a video sequence using the selected key frames. Let $K = [K_1, K_2, ..., K_M]$ be the sequence of key frames, and M be the total number of key frames. The summarized video is formed as:

$$\text{Summarized Video} = [K_1, K_2, ..., K_M]$$

The summarized video provides a condensed version of the original video, highlighting key moments and shot changes.

Chapter 5

Experiment Result

5.1 Continuous Predictions Plot

- It represents the output probabilities from the deep learning model for shot change detection.
- The y-axis shows the continuous probability scores, and the x-axis represents the frame numbers.
- This plot provides a sense of how confident the model is at different points in the video regarding shot changes.

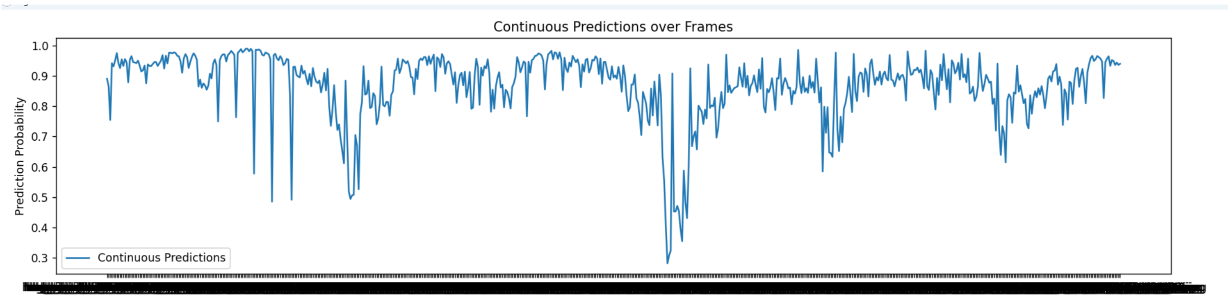


Figure 5.1: **Continuous Predictions Plot**

5.2 Binary Predictions Plot

- It represents the binary predictions obtained by applying a threshold to the continuous predictions.
- The y-axis shows binary predictions (0 or 1), and the x-axis represents the frame numbers.
- This plot simplifies the predictions into a binary decision, making it easier to identify where shot changes are predicted.

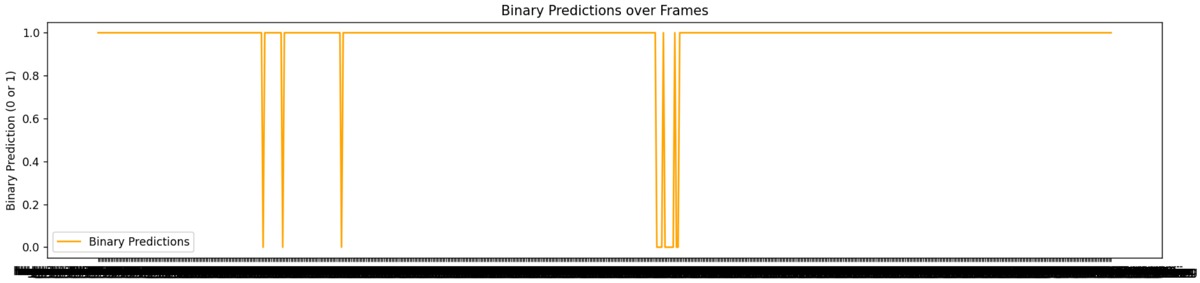
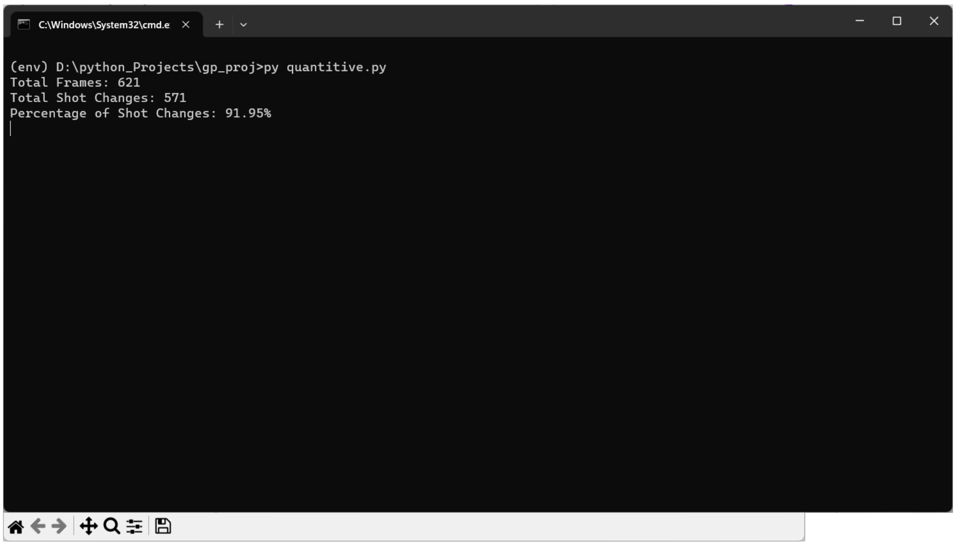


Figure 5.2: **Binary Predictions Plot**

5.3 Quantitative Analysis

- The printed information provides quantitative metrics such as the total number of frames, total detected shot changes, and the percentage of shot changes.
- This information helps in evaluating the performance of the algorithm.

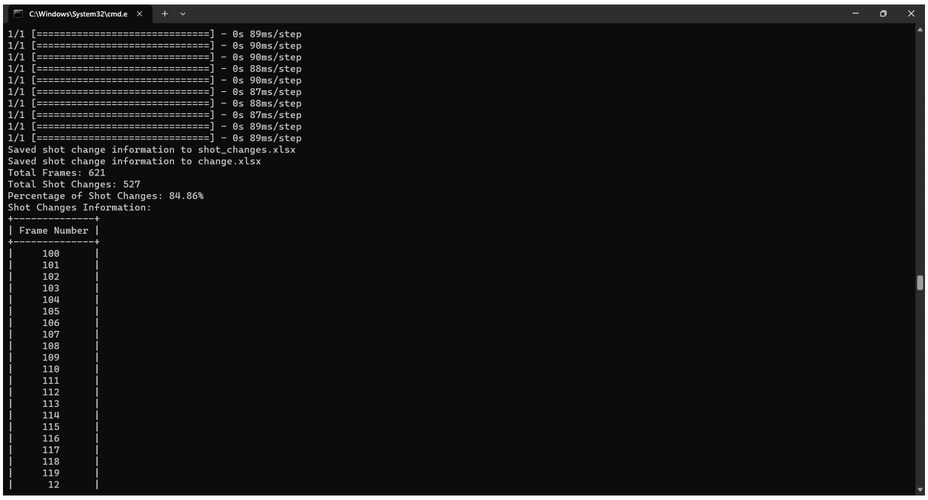


```
(env) D:\python_Projects\gp_proj>py quantitive.py
Total Frames: 621
Total Shot Changes: 571
Percentage of Shot Changes: 91.95%
```

Figure 5.3: **Quantitative Analysis**

5.4 Shot Changes Table

The table displays the frame numbers where shot changes are detected, giving a clear summary of the identified shot changes.



```
1/1 [==============================] - 0s 89ms/step
1/1 [==============================] - 0s 90ms/step
1/1 [==============================] - 0s 90ms/step
1/1 [==============================] - 0s 88ms/step
1/1 [==============================] - 0s 90ms/step
1/1 [==============================] - 0s 87ms/step
1/1 [==============================] - 0s 88ms/step
1/1 [==============================] - 0s 87ms/step
1/1 [==============================] - 0s 89ms/step
1/1 [==============================] - 0s 89ms/step
Saved shot change information to shot_changes.xlsx
Saved shot change information to change.xlsx
Total Frames: 621
Total Shot Changes: 527
Percentage of Shot Changes: 84.86%
Shot Changes Information:
+--------------+
| Frame Number |
+--------------+
|     100      |
|     101      |
|     102      |
|     103      |
|     104      |
|     105      |
|     106      |
|     107      |
|     108      |
|     109      |
|     110      |
|     111      |
|     112      |
|     113      |
|     114      |
|     115      |
|     116      |
|     117      |
|     118      |
|     119      |
|      12      |
```

Figure 5.4: **Shot Changes Table**

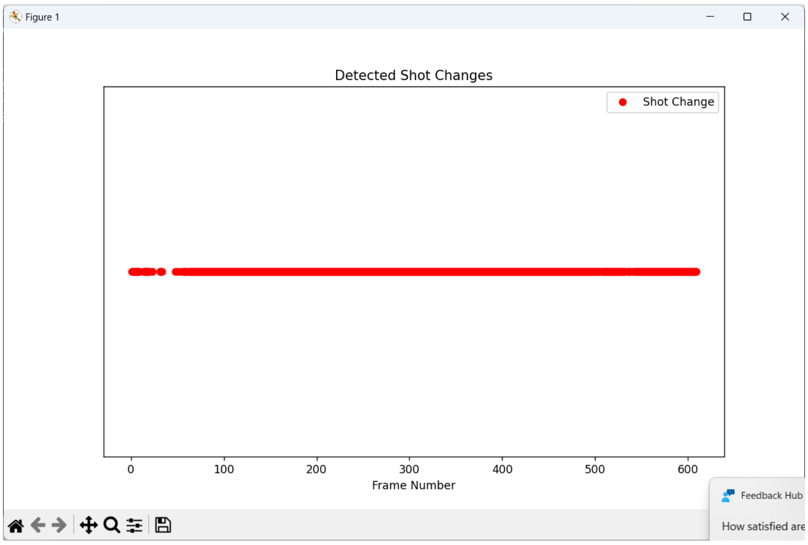


Figure 5.5: **Detected the Shot Changes**

Chapter 6

Conclusion & Future Work

6.1 Conclusion

he shot change detection algorithm presented in this study incorporates a combination of frame differencing, HOG feature visualization, and a deep learning model based on ResNet50. Through quantitative analysis and visualizations, the algorithm demonstrates effective shot change detection. The evaluation metrics, including total frames, total shot changes, and the percentage of shot changes, provide a comprehensive understanding of its performance. The continuous predictions and binary predictions visualizations offer insights into the model's confidence at different frames.

In conclusion, the algorithm showcases promising results in identifying shot changes in videos. However, there is still room for improvement and future work. Fine-tuning the deep learning model, implementing adaptive thresholding techniques, and exploring real-time implementation are potential avenues for enhancement. Additionally, incorporating temporal information in post-processing, evaluating the algorithm on diverse datasets, and developing a user-friendly interface could contribute to its robustness and usability. As the field of shot change detection continues to evolve, the algorithm's future iterations could benefit from benchmarking against state-of-the-art methods and integration with video editing software for broader applications.

6.2 Future Work

The following areas are identified for future work to enhance the shot change detection algorithm:

1. **Fine-Tuning Deep Learning Model:** Conduct further experiments to fine-tune the deep learning model's architecture and parameters for improved performance.
2. **Dynamic Thresholding:** Implement adaptive thresholding techniques based on video characteristics for better adaptability.
3. **Real-Time Implementation:** Optimize the algorithm for real-time shot change detection to support live video analysis.
4. **Enhanced Post-Processing:** Refine post-processing by incorporating temporal information for accurate shot change identification.
5. **Dataset Diversity:** Evaluate the algorithm on diverse datasets to ensure robustness across different scenarios.
6. **User Interface:** Develop a user-friendly interface for easy interaction and parameter tuning.
7. **Integration with Video Editing Software:** Explore integration possibilities with video editing software for post-production applications.
8. **Benchmarking and Comparison:** Benchmark the algorithm against state-of-the-art methods to assess performance.

Chapter 7

Reference

1. Video summarization using deep learning techniques:a detailed analysis and investigation,Parul Saini,Krishan Kumar,Shamal Kashid,Ashray Saini,Alok Negi,Artifcial Intelligence Review (2023) 56:12347–12385 https://doi.org/10.1007/s10462-023-10444-0

2. Video Shot Boundary Detection Based on Feature Fusion and Clustering Technique, Received October 24, 2020, accepted November 17, 2020, date of publication November 26, 2020, date of the current version December 10, 2020. Digital Object Identifier 10.1109/ACCESS.2020.3040861.

3. Chakraborty, S., Singh, A., & Thounaojam, D. M. (2020). A novel bifold-stage shot boundary detection algorithm: invariant to motion and illumination. The Visual Computer. https://doi.org/10.1007/s00371-020-02027-9.

4. Zhou, S., Wu, X., Qi, Y., Luo, S., & Xie, X. (2021). Video shot boundary detection based on multi-level features collaboration. Signal, Image and Video Processing, 15(4), 627–635. https://doi.org/10.1007/s11760-020-01785-2

5. Chakraborty, S., Thounaujam, D. M., Singh, A., & Pal, G. (Year). ALO-SBD: A Hybrid Shot Boundary Detection Technique for video surveillance System.

6. Chen, X. (Year). Sports Video Panorama Synthesis Technology Based on Edge Computing and Video Shot Boundary Detection. Wuhan Technical College of Communications, Wuhan, Hubei, 430065, China. Wireless Communications and Mobile Computing, 2022(Article ID 4060852), 9 pages. DOI: https://doi.org/10.1155/2022/4060852.

7. Abdulhussain, S. H., Ramli, A. R., Saripan, M. I., Mahmmod, B. M., Al-Haddad, S. A. R., & Jassim, W. A. (2018). Methods and Challenges in Shot Boundary Detection: A Review. Entropy, 20(4), 214. DOI: 10.3390/e20040214.

8. Duan, F. F., & Meng, F. (2020). Video Shot Boundary Detection Based on Feature Fusion and Clustering Technique. IEEE Access. DOI: 10.1109/ACCESS.2020.304086.

9. Wu, L., Zhang, S., Jian, M., Lu, Z., & Wang, D. (2019). Two Stage Shot Boundary Detection via Feature Fusion and Spatial-Temporal Convolutional Neural Networks. IEEE Access. DOI: 10.1109/ACCESS.2019.2922038

10. Abdulhussain Sadiq H., Ramli AbdRahman, Saripan M. I., Mahmmod B. M., Al-Haddad S. A. R., Jassim W. A. (2018) Methods and challenges in shot boundary detection: a review. Entropy 20(4):1–42.

11. Boccignone G., Chianese A., Moscato V., Picariello A. (2005) Foveated shot detection for video segmentation. IEEE Transactions on Circuits and Systems for Video Technology 15(3):365–377.

12. Cai C., Lam K., Tan Z. (2005) TRECVID 2005 experiments in the Hong Kong polytechnic university: shot boundary detection based on a Multi-Step comparison scheme, Proc. Int. Conf.

13. . Farshid A., Arding H., Ming-Yee C. (1993) Image Processing on Compressed Data for Large Video Databases. Association for Computing Machinery, New York, NY, USA, pp 267–272. https://doi.org/10.1145/166266.166297 isbn 0897915968 MULTIMEDIA '93 Anaheim, California,USA.

14. Gao Y., Yong J., Cheng F, Yong J., Cheng F. (2011) Video shot boundary detection using Frame-Skipping technique.

15. Gaurav S., Wu W., Dalal E. (2005) The CIEDE2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations. Color Research Application. 30(1):21–30.

16. Gengshen W. U., Liu L, Guo Y., Ding G., Han J., Shen J., Shao L (2017) Unsupervised deep video hashing with balanced rotation. IJCAI 17:3076 3082.https://doi.org/10.24963/ijcai.2017/429.

17. Gong Y., Liu X. (2000) Video Shot Segmentation and Classification

18. HongJiang Z., Kankanhalli A., Smoliar S. (1993) Automatic partitioning of full-motion video. Multimedia Systems. 1(1):10–28

19. Hongjiang Z., Chien Yong L., Smoliar S. (1995) Video Parsing and browsing using compressed data. Multimedia Tools Appl. 1(1):89–111.

20. Kar T., Kanungo P. (2017) A motion and illumination resilient framework for automatic shot boundary detection. Signal Image and Video Processing 11(7):1237–1244.