Story Segmentation and Retrieval of News Videos in a Multi-modal Framework

A

Thesis submitted

in Partial Fulfilment of the Requirements

for the degree of

DOCTOR OF PHILOSOPHY

By

PRANABJYOTI HALOI

Department of Electronics and Electrical Engineering,
Indian Institute of Technology Guwahati,
Guwahati-781039, India,
May, 2024

Declaration

This is to certify that the thesis entitled "**Story Segmentation and Retrieval of News Videos in a Multi-modal Framework**", submitted by me to the *Indian Institute of Technology Guwahati*, for the award of the degree of Doctor of Philosophy, is a bonafide work carried out by me under the supervision of Prof. M.K. Bhuyan. The contents of this thesis, in full or in parts, have not been submitted to any other University or Institute for the award of any degree or diploma.

Signed:

Pranabjyoti Haloi
Department of Electronics and Electrical Engineering,,
Indian Institute of Technology Guwahati,
Guwahati-781039, Assam, India.

Date:

Certificate

This is to certify that the thesis entitled "**Story Segmentation and Retrieval of News Videos in a Multi-modal Framework**", submitted by Pranabjyoti Haloi (11610224), a research scholar in the *Department of Electronics and Electrical Engineering,*, *Indian Institute of Technology Guwahati*, for the award of the degree of Doctor of Philosophy, is a record of an original research work carried out by him under my supervision and guidance. The thesis has fulfilled all requirements as per the regulations of the institute and in my opinion has reached the standard needed for submission. The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

Signed:

Supervisor: Prof. M.K. Bhuyan
Department of Electronics and Electrical Engineering,,
Indian Institute of Technology Guwahati,
Guwahati-781039, Assam, India.

Date:

Acknowledgements

I have the pleasure to acknowledge with the deepest sense of gratitude, the help, and advice given to me, by many people, without which this work would not have been possible.

To begin with, I express my deepest and most sincere gratitude to my Ph.D supervisor Prof. M.K. Bhuyan sir who kindly accepted me to be in his research group and guided me throughout my research work. He has always been supportive to me, motivated me in my difficult times, and always encouraged me for new reserach findings. I have learned many new things about the research process from him.

I am very much thankful to the members of my doctoral committee, Prof. Prabin Kumar Bora sir, Prof. Kannan Karthik sir, and Prof. Arijit Sur sir for their support, encouragement and valuable suggestions on my research work. I would like to thank the faculty members and office-staff of Electronics and Electrical Engineering Department, IIT Guwahati for their help in carrying out the research work. I am also grateful to Late Arnab Kisor Bordoloi, Dibyajyoti Chatterjee, Pooja Rani Borah, and Ariyan Ali who helped me a lot during the collection of the video datasets.

I am thankful to my friends and juniors at the Computer Vision and Image Processing Laboratory of IIT Guwahati, who were always ready for any kind of help required.

Above all, I am deeply grateful to my parents, wife, and other family members. It would not have been possible to complete my work without their love, support, and sacrifice.

Sincerely
Pranabjyoti Haloi

Abstract

Shot segmentation, categorization, indexing, and news story formation are the most important and primary steps in building an efficient and well-sorted video storage and retrieval system. News channels have evolved as one of the primary sources of information. However, in recent times, with the increase in the number of news channels, a plethora of news content is available on air, and it has become difficult to store and retrieve the news videos effectively. Commercials are also included in a news video, containing considerably less information. These commercials are to be filtered out, and the remaining news video will be segmented meaningfully. Segmentation of news videos is a crucial process for efficient storage and categorizing of the videos. The segmented stories also facilitate the easy retrieval and finding of the desired news. In this work, we developed different algorithms for shot segmentation, categorization, indexing, and retrieval of news videos. Our methods are independent of different temporal and spatial structures of various news channels and require a minimal manual inputs.

The spatial and temporal features of a news channel are captioned texts, overlaid texts, logo position, anchorperson of the actual screen, and the position of multiple screens in case of a split-screen configuration. The content of each overlaid text has some unique characteristics. We employed the number, position, and contents within text boxes features for shot segmentation categorization, indexing, and story segmentation. However, the earlier methods do not consider the contents within text boxes, which could have been employed for shot segmentation and categorization.

In Chapter 2, we proposed an adaptive threshold-based multistep approach to detect shot boundaries based on a YIQ colour model. The shot segmentation algorithm compares the histograms between two consecutive frames based on an adaptive threshold. A shot boundary is detected if the similarity measure is less than the threshold. For the measurement of similarity and dissimilarity, cosine comparison is used. Proposed shot segmentation algorithms based on the YIQ model can success-

fully detect the presence of all shot boundaries through segmentation at fade-in/out, dissolve and cut transition.

Commercials constitute a significant chunk of the videos aired on a news channel. These videos are substantially different from the ones containing news, and the inclusion of commercials in between the news videos makes it challenging to segment the stories. In Chapter 2, we also proposed two shot categorization algorithms, where we utilized the information on the presence of text blocks in the spatial arrangement of the news frame, and the color content present in the news frame is selected as a feature for classification.

In Chapter 3, we proposed a novel indexing method based on the text appearing in a news frame. The indexed shot corresponding to similar news content are combined to form individual videos depicting a single story.

We proposed a novel technique for story segmentation and indexing of broadcast news video in a multi-modal framework in Chapter 4. The visual similarity, silence in the audio, and the text in the text boxes of a news video are parameters to define the story boundaries. Each of these parameters is used to create an index, and these three indices are fed to a probabilistic multi-modal algorithm, which then predicts the story breaks.

In Chapter 5, we proposed novel image and video searching and retrieval algorithms based on edge-based and compressed domain features. In image-based searching, the query image is converted into a gray-scale image, and its pixel values are mapped and matched with those in the broadcast news database. In video-based searching, an edge detection algorithm is applied, and the similarity between the detected edges is determined for the data extracted from the video shot and the broadcast news video. For rectification, we checked the similarity between the detected edges of a particular video shot and the broadcast news. To refine our algorithm of edge similarity, we also performed a Structural Similarity Index (SSIM) measure to achieve the most desired results. Finally, we proposed a video searching and retrieval algorithm using scene classification using compressed domain features. The performance of the said feature in classifying the scenes is evaluated by considering different feature lengths. Subsequently, a novel searching algorithm is proposed based

on the entropy of the image. This method incorporates temporal information from the video during video retrieval. Additionally, the proposed system has the facility to query both by video and by image.

The efficacy of the proposed methods is evaluated by performing a large no of experiments on news videos.

Contents

List of Figures

List of Tables

1

Introduction

Contents

1.1 Introduction

A video can be defined as a sequence of images that are shown to a viewer continuously at a velocity that satisfies the persistence of vision for humans. Considering the advancement of digital media, videos are one of the most common considered medium as a conveyer of information. The images must have contextual relationship amongst them so that when viewed in the form of a video has a meaningful layout.

Graphically, video can be structured as shown in Fig. 1.1 according to hierarchically of video clips, scenes, shots, and frames, which are not specific to the genre. Each image in the sequence of images is called a frame, and the playback speed of the sequence is called frame rate per second.

A camera captures a video, and this captured process is not continuous. There is a point in the video where the transition between two abrupt frames occurs. Using modern editing tools, a long video is formed by concatenating two or more discontinuous videos. A continuous stretch of video without any abruption is called a shot [11]. The process of separating shots is called shot segmentation. The transition between the two shots is generally presented as cut, zoom, wipe, etc. A group of successive shots which produce semantically meaningful content is called a scene. or unit.

Broadcast videos are classified into two categories based on their utility (a) Entertainment videos and (b) News videos. With the increasing number of news broadcasters, keeping track of every news on the channel has become challenging. Hundreds of news are displayed daily without general patterns or categorization. Though the news channels have figured out many different shows, their contents vaguely remain the same. Categorization of news videos is one of the main problems of news videos considering the amount of information they convey. The main aim of categorization is to segment the shots of news videos meaningfully. One of the prime objectives of categorization is to filter out commercials and news videos and store

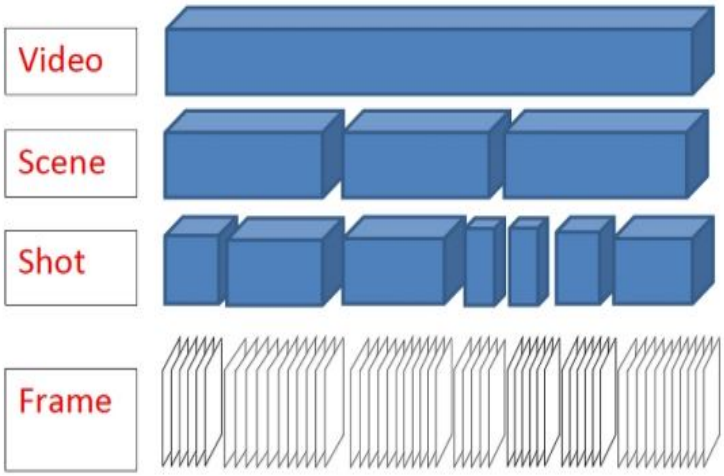


Fig. 1.1: Graphical structure of a news video.

them separately. After the news is separated from the commercials, the news video can be further separated into shots. The shot separation is based on the transition from one shot to another.

Simply shots separation does not convey useful information, and the shots need to be grouped effectively. The shots, when grouped meaningfully, are called stories. A story may contain different types, such as anchorperson shots, field shots, split-screen shots, etc. In today's world, where people don't want to spend much of their time in front of the television, story segmentation can be the one-stop solution for efficient data accumulation from the news videos.

After clustering shots into meaningful stories, the stories are to be indexed so that the user can find relevant stories that he searches for. Without the indexing of stories, they are just a cluster of similar shots. The indexing is done for the convenience of the user and the broadcasters.

Retrieval is the process of retrieving the data of a particular video shot and comparing it with a part of broadcast news, and checking its occurrence in the broadcast news videos

1.2 Spatio-Temporal Structure of a News Video

Different multimedia videos produce based on different semantic structures. Among all news, a video structure is unique. A complete news program consists of news stories combined with commercials and some miscellaneous sections[14, 15]. News stories comprise one or more stories and transitional effects between the consecutive shot.

Each news channel has spatial and temporal formats based on semantic domain knowledge for presenting news [16]. Different News channels have different news presentation forms depending upon their editorial policies, and it varies with the genre, time slot, the topic being discussed, the program's content, etc. For efficient video classification and program genre (Headline news, breaking news, discussion, interview, political news, national, international, and local news) identification, the domain knowledge of news programs' spatial and temporal format is important. Each of these features is being discussed from a general perspective in the literature:

1.2.1 Spatial structure of a news video

The spatial features of a news channel include the text boxes, logo, position of the actual screen, and the position of multiple screens in case of a split-screen configuration. Though the spatial and temporal features of one broadcaster may vary from another, the basic contents of a news video always remain the same. A news channel logo is generally present on the screen, while the number of text boxes and split screens may vary from frame to frame. In some cases, the actual news screen can shift to show advertisements or special announcements.

The text block appears in a news video frame can be classified into Local Text Block, Global Text Block, and Scrolling Text Block depending on context as shown in Fig. 1.2.

- Local text block:- The Local Text Block contains text that relates to the

Fig. 1.2: Spatial structure of a News Frame

current news being broadcast on the news channel. It basically consists of three to five sentences that are periodically repeated until the current news being broadcast changes.

- Global text block:- The text in the Global Text Block shows in turn all the news in the day either randomized or following a particular pattern. This text may or may not have any relation with the current news being broadcast on the channel.

- Scrolling Text block:-The Scrolling Text Block text usually scrolls throughout the screen showing advertisements, sports scores, business-related data, etc,.

All or at least a few of the mentioned blocks (Global and Local Text Block) are always present in all the news programs of different news channels. However, during a commercial, one or more of the text blocks are removed to provide more screen areas to display the advertisement. A frame of news and commercal has shown in Fig. 1.3 and Fig. 1.4 respectively.

One of the major addition to the modern spatial structure of a news program is the presence of split screens. Such a type of screen as shown in Fig. 1.5 has the screen split into two parts, each part showing a story in the same context, yet

Fig. 1.3: A news frame

Fig. 1.4: A commercial frame

different in their content. This allows the news program to display reports from different field for the same news story.

(a) *(b)*

Fig. 1.5: Spatial structure of frames in case of split screens

1.2.2 Temporal structure of a news video

The news videos generally consist of two types of videos *i.e.,* news and commercials. News videos are informative, while commercials consist of considerably less information. Commercials are inserted in between the news on the channel. The news section consists of a variety of shots as studio or anchorperson shots, field shots etc. The commercial section consists of commercial shots along with some other special shots as sport scores, weather reports, breaking news etc,. Though

these special shots are present in the commercial section yet they are of informative nature of the news and should be categorized along with the news shots. A repetitive unit of news section and commercial section makes an entire news program. The temporal structure of a news video is shown in Fig. 1.6.

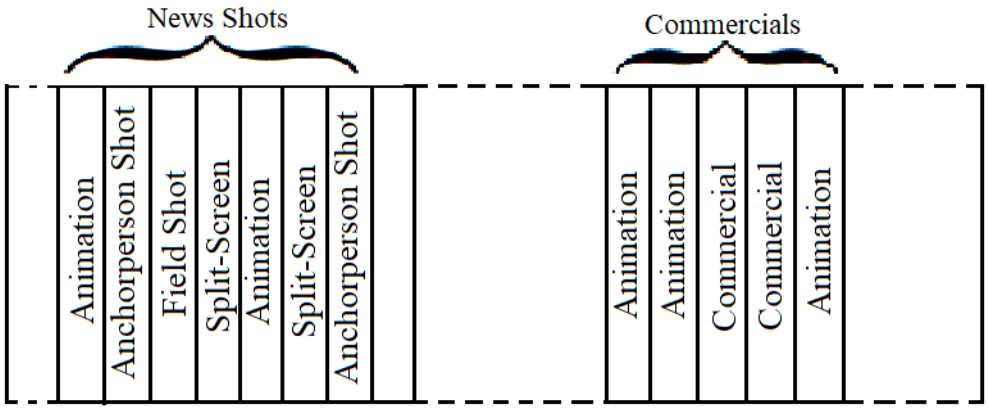


Fig. 1.6: Temporal strucure of a news video

1.2.3 Types of shots in a news video

A news shot is usually an integration of several segments or shots. These shots or segments can be classified and described based on the location it is recorded and the subjects being filmed during the duration of the shot. These segments are described as follows.

1. *News section :* Any shot in the duration of the news program which contains information that portrays anything which can be bounded in the scope of news is called a news shot. News shots can be classified as follows:

   (a) *Studio Shot:* As the name says, in a studio shot the recording usually takes place inside the studio. Game scores, Weather reports, political debates are examples of studio shots.

   (b) *Anchorperson Shot:* Anchorperson shot consists of a news reader. That portion of the news program where one or two anchors read the news is described as a news shot. An anchorperson shot is also a studio shot.

(c) *Field Shot:* A field shot portrays the news from the actual place of happening. A field shot consists of a field reporter. An anchorperson shot is usually followed by a field shot.

(d) *Animation shot:* An animation shot usually displayed at the beginning of a news program portrays the logo of news channel or the name of the show through some computer edited techniques.

2. *Commercial section:* A commercial shot is displayed within news programs and acts as an advertisement of various products and services.

1.3 Description of Shot Transitions

Shot segmentation in news videos is possible as it shows some definite patterns during shot transitions. These patterns include change in content, change in illumination, shifting of pixels, etc. Shot segmentation refers to the detection of these patterns. There are five types of shot transitions that are commonly used in news videos. They are: -

Cut: It is the most common type of shot transition. A cut occurs when a shot is suddenly replaced by other. It is an abrupt change of the content of the frame. It is shown in Fig. 1.14.

Fade in/ Fade out: In this type of transition, brightness of the frames gradually decreases and a complete black frame appears, after that frame, again the brightness increases gradually. Or the brightness of the frames increases to a complete white frame and gradually decreases again as shown in Fig. 1.15.

Zoom: In case of a zoom transition, there is a radial motion of the pixels about the centre in the frames undergoing the transition as shown in Fig. 1.16.

Wipe: In a wipe transition, a portion of the frame is replaced by that of a new frame gradually. It is shown in Fig.1.17.

Dissolve: The process of dissolve takes place through a number of frames. The content of last frames of a shot disappears into background while simultaneously

Fig. 1.7: Anchorperson Shot

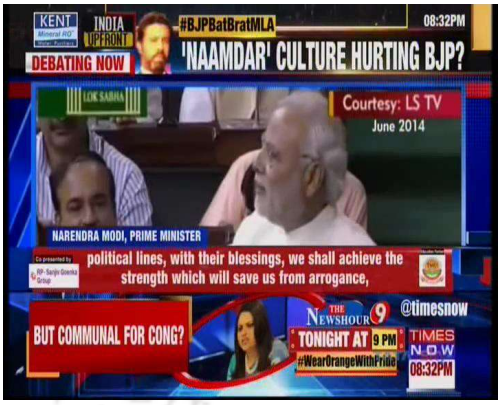


Fig. 1.8: Field Shot

Fig. 1.9: Studio Shot

Fig. 1.10: Split-Screen Shot

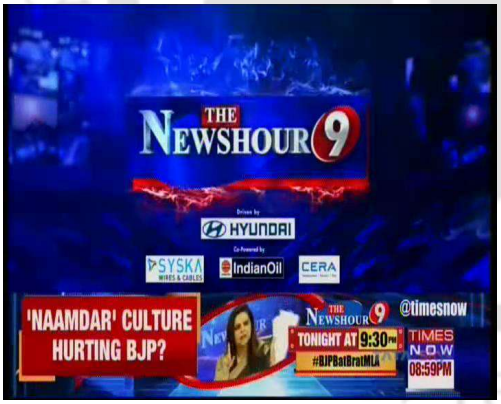


Fig. 1.11: Animation Shot

Fig. 1.12: Commercial Shot

Fig. 1.13: The frame structure of different shots

the content of first frames of next shot appears in the foreground. It is shown in Fig. 1.18.

(a) (b)

Fig. 1.14: Cut Transition

(a) (b) (c)

(d) (e) (f)

Fig. 1.15: Fade Transition

(a) (b) (c)

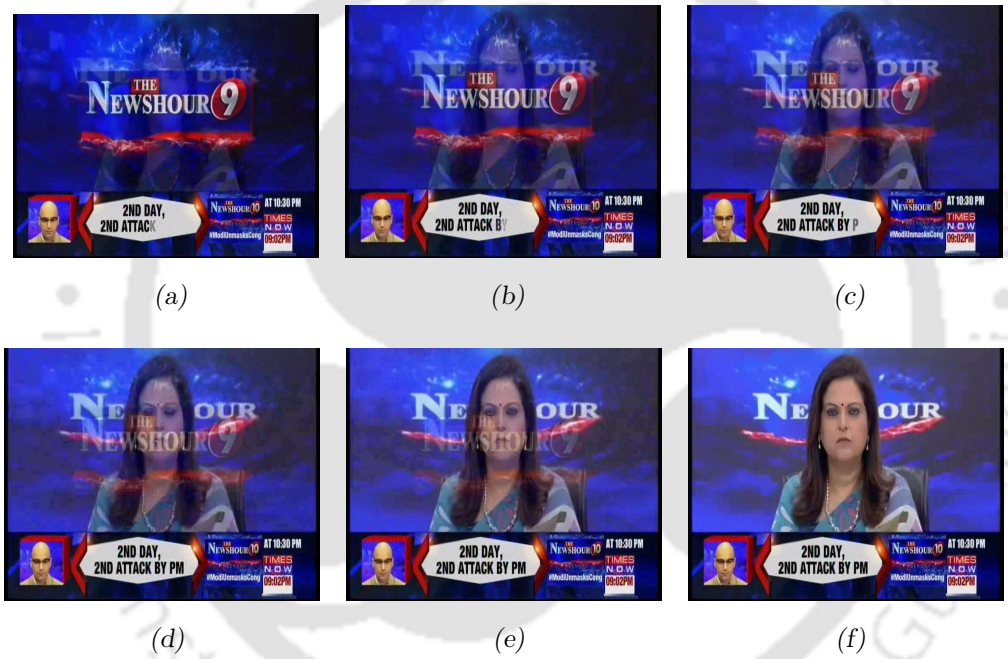
(d) (e) (f)

Fig. 1.16: Zoom Transition

(a) (b) (c)

(d) (e) (f)

Fig. 1.17: Wipe Transition



(a) (b) (c)

(d) (e) (f)

Fig. 1.18: Dissolve Transition

1.4 Literature Survey

Although televising news is only a few decades old, there has been tremendous progress in the classification of the video aired on television. Such a classification of off-the-shelf metadata at the consumer end eliminates any discrepancy that might arise due to broadcasters' vested interest and thus is a necessary tool for keeping a check on the broadcasters. Since the inception of news channels, researchers have been curious to devise methods to classify aired content. A few approaches have been discussed.

1.4.1 Literature survey relating to video format analysis

Acquiring domain knowledge about spatial and temporal presentation formats is very important to analyze formats of a video. It can help us extract information from broadcast news videos for semantic classification and indexing the news stories according to genre and topic. Each news channel has a predefined particular spatial and temporal format guidelines to present news stories to make presentation attractive. Based on this spatial and temporal format, Swanberg in [1], Zhang, Gong, Smoliar, and Tan in [16] developed an approach to locate and identify frame structure. They showed that within the anchorperson shot, there is a spatial structure, and there is a temporal structure between the shot. In Fig. 1.19, author showed different region models: anchorperson position and number, news icon location, title bar existence, anchorperson name bar, background, etc. Based on the spatial arrangement of regions, a frame can be modeled. A particular shot (Ancorperson shot, News shot, Commercial shot) is modeled based on the sequence of frame models. In Fig. 1.20 author showed the temporal structure of a news program. This temporal structure is used to locate shot boundaries. According to their model, a starting shot occurs, like animation, followed by an anchorperson shot. Each news shot starts and ends with an anchor person. In a completed news cycle, some other shots like commercial brake and weather reports exist, which are followed by an

anchorperson shot. At last, there is an ending sequence to complete the program. The model may vary from channel to channel and at different times. It may also have some computational complexity. Gunsel *et al.*, in [17] used template matching

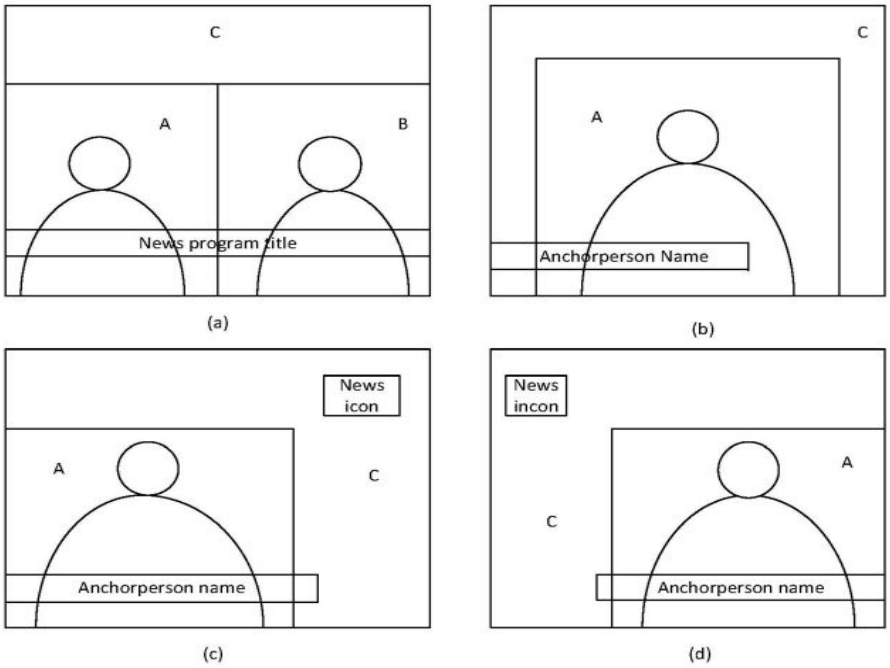


Fig. 1.19: Region based anchorperson model. A,B and C denoted area where the region model image are selected [16].

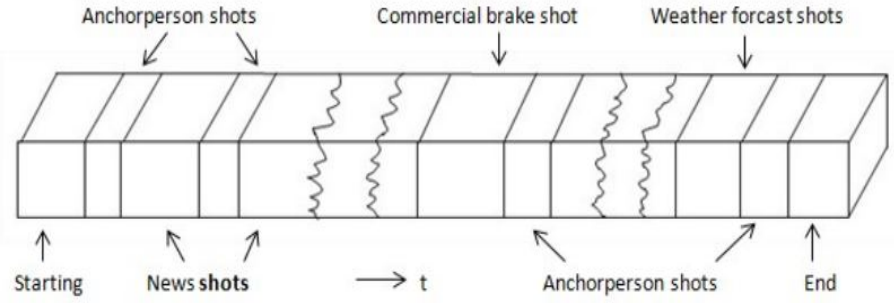


Fig. 1.20: The temporal structure of a typical news program [16].

and color classification to identify anchorperson shots. First, they used the anchorperson region model. They used the anchor person's skin color and the histogram intersection method to extract the anchorperson region. Then they compared this

extracted anchorperson region with templates stored in the database. This type of template matching is also time-consuming and depends upon the application data. Meng Cai *et al.*, in [2] exploited shot change frequency and logo detection to separate commercials from news shots. Ozay and Sankur in [3] also used the same channel logo and product trademark information for separating commercials from the news. According to them, TV logos partly or fully disappear during a commercial, and this information can be exploited in segmenting out commercials. In [14], Hanjalic, Lagensijk, and Biemod used a spatial formate-based template for detecting the anchorperson shot. They considered the same background for different anchorperson models. However, it is only valid in some cases. Due to varying camera angles, different anchorperson models have different backgrounds. Xibon Gao and Xiaoou Tang in [18] presented an unsupervised shot boundary technique. They used a "two-pass modified fuzzy c-means" algorithm for shot boundary detection. Moreover, for classifying video shots as anchor shots, they used the graph-theoretical cluster (GTC) analysis method-a model-free anchorperson shot classification analysis method. According to them, each field shot is followed by an anchorperson shot. Moreover, indexing is used based on simple temporal syntax as mentioned in [16]. This model-free approach is much faster and simpler than the model-matching method. Dean et al. [4] presented a multimodel (visual+audio+text) approach to segment out commercials and carry out semantic content analysis to detect commercial segments. Liu, Zhao, et al. [5] used text features with an adabost classifier to segment out commercials from news videos. They divided the screen area into some rectangular grids and found the presence of text or not. Jindal, Tiwari, and Ghosh [6] used the location of ticker text to segment out news and commercials from the news videos. Agnihotri et al. [7] used black frame detection using the luminance DC value feature of the entire frame by applying a threshold-based technique. Duygulu, Pan, and Forsyth in [8] proposed a novel method for detecting and tracking the evolution of news over time using visual and textual information, which helps in summarizing the event into a documentary, and facilitate indexing and retrieval. In

[9], combination of text quarries and visual cues are used to semantically classify videos. In [5], text is used as a feature to classify commercials and non-commercials. Adrian-Gabriel and Sbastien Fournier in [10] used the presence of video transcripts of a video to detect shot boundaries using a method based on lexical chains. They defined a framework SegChain W2V for video segmentation.

1.4.2 Literature survey relating to the classification of video shots

The amount of video a viewer can access nowadays from the internet and television is large. So, it is challenging to find a video of interest for a viewer without going through all the videos. To achieve this objective, automatic video classification is essential. Video classification is essential for various applications like story segmentation, retrieval, and summarization. It bridges the gap between the structural organization of videos and low-level features extracted from the video. For the classification of videos, all the videos are generally placed into some categories and assigned to a meaningful label corresponding to each type [16].

For the video classification, the video is first segmented into shots based on temporal structural analysis. A shot is a consecutive sequence of frames with strong content correlation captured by a camera action between start and stop operation, marking shot boundaries [19]. A collection of conjugative shots coherent with a theme or subject forms a continuous video. Hence, shots can be considered the fundamental unit for video classification. Camera shot transitions are mainly abrupt or cut and gradual transitions. Gradual transitions include dissolve, fade in, fade out, wipe, etc. [19]. For the detection of shot boundary, features are extracted, such as low-level image features, like color, texture, and motion features from each of the frames, and various high-level features like object, textual, and audio features. Then, similarity measures are determined (including pairwise and window-based [20]) between successive frames based on the extracted features. The shot boundary can be extracted using different similarity measures like “threshold-based approach” or “statistical learning-based approach” [21]. In [21] and [22], the author presented

a detailed literature survey on shot boundary detection.

The second step of video classification is assigning videos to some predefined meaningful shot categories based on extracted features and category labels assigned to each shot category. This label should reflect the shot content, which can be used for the subsequence stage of segmenting, classifying, retrieval [19], [29], [30], [31] and summarizing [19]], [32]. Some examples of shot categories are [33] Intro/Highlight, Anchor, 2Anchors, Meeting/Gathering, Speech/Interview, Live-repot, Still image, Sports, Text-scenes, Special, Finance, Weather, and Commercials. The final step is the shot classification [19] using different models or classifiers. In this step, some keywords are selectively indexed to shots according to the appropriate correspondence of typical shot classes. The semantic attributes of keywords are based on video edit effect, video program genre, video event, and objects.

Histogram and pixel pair comparisons are used to detect shot boundaries in [16]. It uses a twin comparison method for the detection of gradual transitions. It compares this with a basic model of SBC consisting of a shot, a combination of frames, and regions for detecting anchorperson shots. In order to reduce computational complexity, it involves the detection of potential shot candidates depending on the detection of mean and variance over a shot of its intensity values (Anchorperson shots have minimum variance and threshold). After defining an anchorperson depending on the results of the region-based comparison, a model anchorperson is defined by averaging the frames of the anchorperson shots. Based on the location of anchorperson shots, it detects a news story. In [16] developed an algorithm based on comparison measures that are very primitive, like pixel comparisons, averaging, difference, etc. Modern broadcast news consists of multiple split screens, animation edits, moving anchorpersons, etc. With the help of CCV (Colour Coherence Vector), a refined histogram, a signature formation technique has been tried for shot segmentation [23]. The histogram differs from an average histogram in that it defines it based on the number of pixels of the colour level in homogenous regions and those not in homogenous regions. It also uses a Point of Interest descriptor, a

unique feature of a news program, for detecting cues in program segmentation.

Topic-based segmentations involve the structural model formation using the detection of the anchorperson. For this, it automatically forms a model of the anchorperson frame using face detection with the help of the Adaboost filter. Anchorperson shots are used to identify the internal structure of the news program. The algorithm discussed in [23] relies on distinctive signature features present in different time intervals that can be used for program segmentation. Similarly, it uses the presence of anchorperson shots for topic-based segmentation. In the case of animated news background, the concept of background is nullified. Besides, close-ups of reporters also act as anchorpersons. There are common cases when the same news story has the presence of more than one anchorperson shot or the occurrence of split screens.

With the introduction of modern video editing processes, the presumption of a particular framework or model dramatically narrows the scope of the technique. Combining features or models makes the algorithm a slow memory consumer. Besides this, modern broadcast styles vary widely from news channel to channel. The discussed algorithm in this work promises to be independent of any particular structure.

A news video consists of text blocks for a detailed display of news. However, the number of text blocks present in a particular frame depends on the section of the news program being broadcast [6]. For example, in the case of commercials, the frames contain only one text block. In the case of news, the number of text blocks increases to display more information.

Shot classification and subsequent story segmentation, proper choice of shot categories, and selection of features are important. The assigned category tags should reflect the respective shot content and can be utilized correctly in the preceding stages of segmentation, classification, retrieval [19], [29], [30], [31], and summarizing [19], [32]. The literature presents various semantic concepts in the form of multimedia vocabularies, taxonomies, and classification schemes. These categories

and taxonomies are formed based on domain knowledge, mainly video program genre, video topic, subject(s), or action(s). The "National Institute of Standards and Technologies" has created some high-level features for early evaluations, but they provide limited coverage of semantics. The “Large-Scale Concept Ontology for Multimedia” (LSCOM) created a taxonomy considering 1000 semantic concepts [38] to describe multimedia news videos. The LSCOM ontology is developed based on news broadcast video archives from BBC and CNN, with a small version of the ontology shown in Fig. 1.21. LSCOM ontology is among the popular ontologies for semantically classifying broadcast news videos. In [33], Chaisorn *et al.*, explored

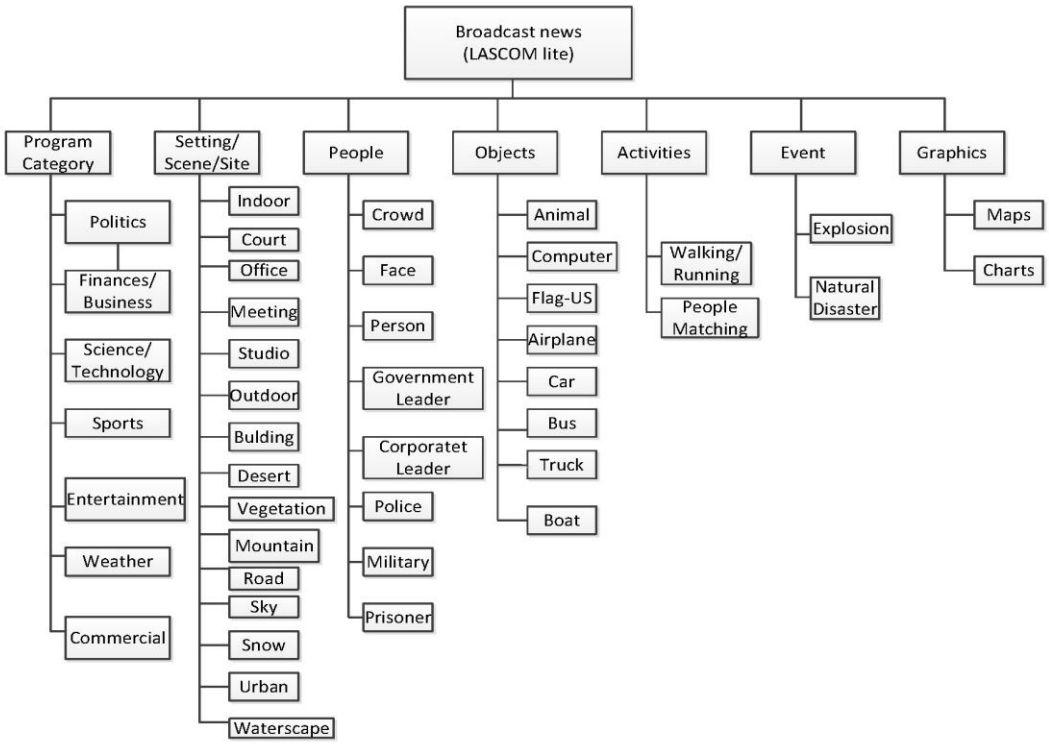


Fig. 1.21: Few top levels concept of the LSCOM taxonomy

17 shot categories based on a taxonomy of the TV Any-Time model as a guide. They aimed to cover nearly all essential types of shots, including "Intro/Highlight, One Anchor, Two Anchors, Meeting/Gathering, Speech/Interview, Life-reporting, Still image, Sports, Text-scene, Special, Finance, Weather, Commercial, Leads, Top,

Sport, Play, and Health Logo Shot.

1.4.3 Literature survey relating to story or scene segmentation

A scene is a collection of shots along with the transition effects in between successive shots with meaningful semantic units. The scene has a higher level of semantics than the shots, which motivates the researcher to research story segmentation. Semantic story segmentation is archived by collecting successive shots with some similarity measures into meaningful semantic units, and different literature presents various similarity measure methods, which are made based on some multimodal cues, including textual, visual, and audio features. These segmented stories are then indexed using different multi-model features extracted at different levels of segmentation. Most broadcast channels broadcast the same news receptively in the same channel and vary their prospective channels. Hence, it desires to organize the news video archives by linking semantic stories.

Story segmentation was a task during the Text Retrieval Conference Video Retrieval Evaluation (TRECVID) in 2003 and 2004, during which many research groups developed various techniques. The current story segmentation techniques can be broadly divided into three classes [34, 35]:

(a) Text-based methods: Features used for text-based methods include closed caption text, automatic speech recognition, and text tiling information.

(b) Heuristic Rule-based Techniques: These techniques exploit domain knowledge of news video structures and multimedia production techniques. Features used in heuristic rule-based techniques include anchor shot appearance, silence, blank frames, cue phrases, etc.

(c) Machine-Learning-based Techniques: Various automatic machine learning methods that incorporate a combination of audio, visual, and text features are used.

In [35], Chua, Chang, *et al.*, evaluated the performance of TRECVID 2003 in terms of the F1 measure, as shown in Figure 1.22. Segmentation is based on (a)

Text features (only ASR output), (b) Audio-visual features, and (c) A combination of audio-visual-text features. They employed rigorous machine learning techniques and judiciously used full multi-modality features for segmentation, finding that combining visual, closed caption text, and ASR yields good segmentation performance.

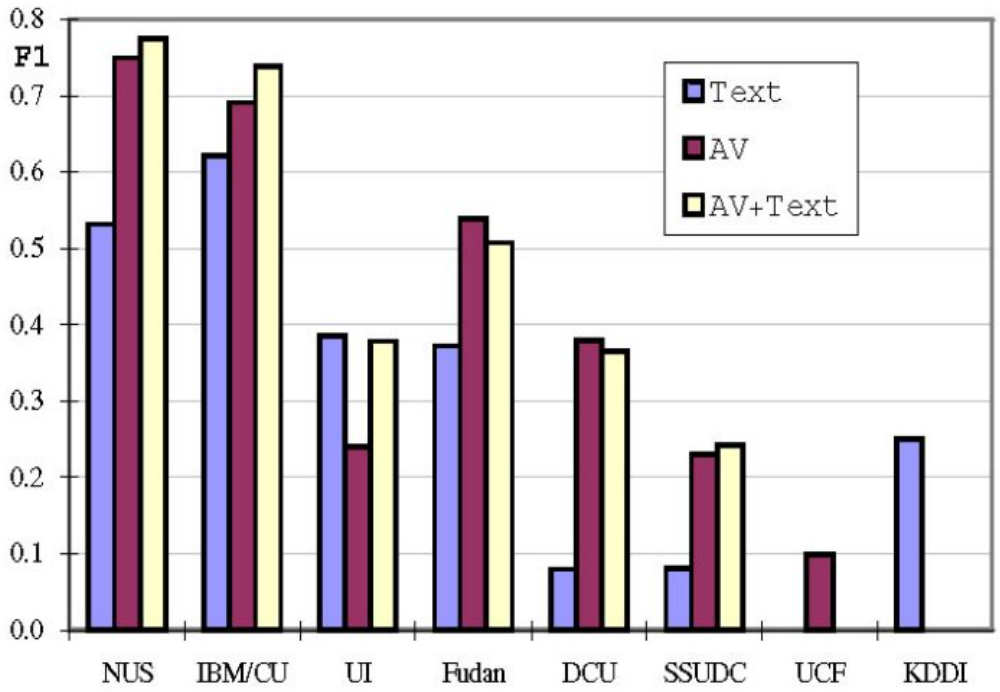


Fig. 1.22: text, AV, and combination of features based segmentation[35]. The groups are abbreviated as NUS [36], IBM/CU [37], UI, Fudan, DCU, SSUDC, KDDI, and UCF

Utilizing multimodal features for news story boundary detection and classification, as discussed in [24], based on categories such as Introduction, Local News, Local Politics, Sports, Finance, World Politics, World News, and Weather Forecast, has shown promising results. This approach employs Bayesian Networks and Hidden Markov Models. Different sensor inputs, including text, silence, and facial expressions, are used in Bayesian Networks to detect various scenes. Hidden Markov Models are then applied to segment these scenes.

At the second input level to Bayesian Networks, audio is analyzed, and the

semantics of the segmented scenes are classified. Another perspective involves the combination of multimodal features, as mentioned in [47], such as Face shot, text, junk frame, logos, and audio features like ASR, Silence, Speaker, Transition words, etc., for shot segmentation. Based on the detected features, they are fused, and a classifier is used to identify shot boundaries.

A scheme for semantic story segmentation, relying on anchor person detection, has been employed [39]. The proposed model utilizes a split and merge mechanism to identify story boundaries, incorporating visual features and text transcripts. Performance evaluation was conducted using TRECVid 2003 CNN and ABC news videos.

Another approach to news video segmentation, based on perceived shifts in content using features across multiple modalities, has been demonstrated [25]. Various multimedia features, acting as potential indicators of a change in the story, were investigated to identify the most effective ones. The efficiency of the approach is illustrated by a prototype's performance, where several feature combinations show up to an 18% improvement in WindowDiff score compared to other state-of-the-art story segmenters.

This work addresses two crucial components in a news video story parsing system: shot boundary detection and anchorperson detection [18]. Initially, an unsupervised fuzzy c-means algorithm is employed to detect video shot boundaries, segmenting a news video into distinct shots. Subsequently, a graph-theoretical cluster analysis algorithm is implemented to classify video shots into anchorperson shots and news footage shots.

A two-level multi-modal framework for segmenting and classifying news videos into single-story semantic units is presented [26]. Decision trees are utilized to classify shots into one of the 13 predefined categories or mid-level features. At the scene story level, HMM models are applied to locate story boundaries. Additionally, a global rule induction technique, commonly used for information extraction (IE) from text documents, is adapted for identifying story boundaries in news videos

within this framework.

A hybrid algorithm is introduced to classify each input video shot into one of the predefined genre types, followed by the application of a global rule induction technique to extract story boundaries from the sequence of classified shots. Two novel approaches are proposed: one utilizing the video stream and the other the close-caption text stream for segmenting TV news into stories [27]. The segmentation of the video stream into stories involves detecting anchorperson shots, while the text stream is segmented into stories using a Latent Dirichlet Allocation (LDA)-based approach. The benefit of the proposed LDA-based approach is that it provides the topic distribution associated with each segment along with the story segmentation.

In this study of video story segmentation, a set of key events is initially detected from heterogeneous multimedia signal sources, including a large-scale concept ontology for images, text generated from automatic speech recognition systems, features extracted from audio tracks, and high-level video transcriptions [28]. Subsequently, a discriminative evidence fusion scheme is investigated.

Goyal *et al.*, [39] worked on story segmentation based on detecting anchorperson shots. They assumed that a news video always began with anchorperson frames and considered the frames of the first 25-50 seconds of the video as a possible anchorperson template. These frames are compared with key frames from each shot, and the average dissimilarity is calculated. The frame with minimum variation from the average is considered a template for anchorperson shots. It is compared with the key frame of each shot, and the shots having dissimilarity greater than a threshold are discarded. The remaining shots are considered the beginning of a new story. The eigen difference between every second frame of a story and between the last and first frames of two consecutive stories are calculated and compared with a threshold to detect unsegmented and over-segmented ones. They also analyzed the application of SVM, ANN, J48 decision tree, and Naive Bayes classifiers to detect the anchorperson shots based on some features. The authors in [40] also used anchorperson frames to determine the beginning of a story. Short-time energy and

short-time average zero-crossing rate are used for silence frame detection. Captions that appear at the start or middle to express the story's meaning are also detected. While the assumption of an anchorperson frame at the story's beginning is prevalent, a few assumptions are tailored for specific features in a particular region. In [41]-[42] it is assumed that a news story has a topic caption frame and there are one or more silence clips before and after the appearance of the topic caption frame in the temporal axis. The shot boundary is detected using the (X^2) histogram matching method mentioned in [43]. They used frames containing topic captions that express the story's meaning at the start or middle for text detection. ZCR and energy measures are used to detect silence in audio. The authors in [44] used some assumptions on the length of sentences, the relative position of the sentences, the number of pronouns used, etc., at the end of a story. JRip Machine learning algorithm is used to train the classifier. Annotations produced by the NIGHTINGALE system, speech recognition transcripts, speaker identification, and sentence segmentation are the features used in the classifier. Wang et al. in [45] have used features like Program Oriented Information Images (POIM), textual content similarity, and audio scene change to achieve story segmentation. A basic structure of program content and POIM images is considered. The audio scene change is to spot the prominent audio change from speech to music during commercials. Some assumptions are made, such as patterns in POIM images with a clear background, uniform visual components, and rich induced texture. Latent semantic analysis is employed to model textual information, and ASR transcripts have been applied to news videos. The cosine distance between the shot document vectors calculates the textual content similarity. Park and Li [46] worked on detecting domain-specific keywords for indexing instructional videos and classified shots into fifteen possible categories using three visual labels and five audio labels. Out of these categories, it is assumed that salient keywords lie at the beginning of the narration, question-answer section, and transition shots. This information is used to recalculate the saliency of the detected words. An assumption of the logo being placed in the same place

and continuously in a news video, except during commercials, is also seen in [47]. Dumont *et al.* in [47] extracted relevant information on all one-second segments. Visual information includes the presence of a particular person, shot detection, junk frames, the presence of a channel logo, visual activity, and text. Audio information like the presence of silence and automatic speech recognition (ASR) techniques are used to extract semantic meaning from the audio. The multimodal features are merged using a classifier with a prediction score for story transition. In [33] and [48], the authors used ASR confidence level along with a combination of black frames and silence to detect commercials, while speech recognition system (Sphinx-II) in [49] is used to extract the spoken words in the video. In [47], the ASR uses continuous density HMM with the Gaussian mixture and four-gram statistics. Chairson et al. [33] employed the Hidden Markov Model(HMM) for story segmentation using audio-visual and text-based ASR features individually. They used a hierarchical approach in which the visual and temporal features are extracted based on the features, shots are segmented, and finally, story boundaries are detected and classified as 'news' or 'misc.' Multi-resolution analysis and wavelet transformation techniques are used to combine both types of features. In [48], the tagged category, location change, and cue phrases are used with the Hidden Markov Model (HMM) technique for story boundary detection. Feng *et al.* [50] have extracted audio classes, shot duration, the number of faces, motion activity, face size, topic caption, centralized video text, and zoom in/out. Anomalies created as short shots are combined with neighboring shots, and the Hidden Markov Model(or HMM) is employed to detect story boundaries.

While the cosine angle between two successive color signature vectors is compared with a threshold to detect a shot boundary [51], Shot identification based on histogram/ color histogram is also evident [48], [49], [52], [53]. In [48], the shots of the news video are identified based on a) Low-level feature, *i.e.,* color histogram, b) Temporal feature, *i.e.,* scene change, background noise, background music, motion activity, and shot duration and c) High-level feature like the number of faces, type

of the shot, *i.e.,* close up, medium and long-distance shot. After that, these shots are classified as commercial, anchorperson shots, weather, or sports news. Zhai *et al.*, in [54] identified anchorperson shots using the color histogram and face correlation techniques and are considered the beginning of a story. A Shot Connectivity Graph (SCG) is used to segment the stories. A single node represents similar shots of the news video, and the transition between those shots is represented by the edges connecting the nodes. Sports-related stories are identified by calculating the correlation between Automatic Speech Recognition(ASR) results with a database of sports-related keywords, whether weather stories are identified by analyzing the color pattern and the motion content. In [55], a graph structure with each frame as a node is considered, and visual similarity between two frames determines the weight of edges connecting them. Graph partition is performed within a temporal sliding window of frames to identify the shot boundaries. The authors can differentiate between news presenters and guests using image retrieval techniques. Hui *et al.*, in [53] differentiated between a pair of consecutive frames by Spatial Difference Metric (SDM) and Histogram Difference Metric (HDM). The feature space obtained is partitioned into the Significant Change (SC) and Non-Significant Change (NSC) by the use of the Fuzzy C-Means (FCM) clustering algorithm. The first frame of each shot is used as a representative to differentiate between anchor shots and field shots, and a graph-theoretical cluster analysis algorithm is used to detect studio shots of the same pattern. Hauptman and Witbrock [49] detected scene breaks using color histogram techniques and Lucas-Kanade optical flow analysis. Frame similarity and closed-captioned transcripts are used as a feature in the story segmentation process. Frame similarity is determined by comparing a key frame from each scene. Using MPEG Optical glow techniques, the scenes with motions are considered within a story and not at the boundary. The closed-captioned transcripts contain markers that indicate a story change and a speaker change. These spoken words are used to align the closed-captioned transcripts with the actual video footage through a dynamic time-wrapping procedure. Yeh *et al.*, [52] have used a color histogram

technique in the YUV color space to detect the substantial cuts in the video. The segment between two strong cuts is selected as a possible commercial if the number of cut transitions in a minute exceeds a threshold. To determine the commercial boundary, a color coherence graph between the key frames of the possible commercial shots is plotted. The first and the last local minima in that range, excluding the endpoints, are finally detected as the commercial boundary. Zedan *et al.*, [56] represented the frames in terms of the dominant colors of the R, G, and B channels. The difference in the dominant color component between two consecutive frames gives a dissimilarity vector, which is then used to detect the abrupt cut transitions in the scenes by training a Feed-Forward Neural Network with two hidden layers. In [51] the cosine angle between two successive color signature vectors is compared with a threshold to detect a shot boundary. The frame in a shot that has a color signature nearest to the average is considered a keyframe. Based on the color similarity of the keyframes and the temporal distance between the shots, they are categorized into different groups.

Semantic similarity and spatiotemporal features are two commonly used approaches in the story segmentation of news videos. However, more than two approaches are required to provide satisfactory results. Kannao and Guha [57] worked on the drawbacks of the two features and combined them with Conditional Random Field (CRF) models. Web-based news articles are attached to the overlay text from the news shots, and the Jaccard Index between two such articles is used to evaluate the text-based similarity feature. The shots with similar spatial features are combined using Grid-wise Edge Orientation Histogram vectors. In [58] the authors used six binary classifiers to categorize video shots, which are characterized by different presentation styles. The classifiers used are - a) Multi-view vs. Uni-view, (b) Graphics vs. Natural, (c) Non-informative graphics vs. Info-graphics, (d) Studio vs. Field (e) Indoor vs. Outdoor and (f) Face Shots vs. Non-face Shots. Ten semantic categories were found valid after sorting the possible label combinations. The shot sequences are classified as either news, interview, or debate using CRF

models. CRF model classifies the shots to be begin shot, middle shot, filler shot, and end shot, and a story is identified when a sequence of middle shots enclosed by begin and end shots appears [57] [58].

1.4.4 Literature survey relating to video searching and retrieval

The advancements in VLSI, broadband networks, mass storage devices, and computing and processing technologies have enabled the creation and storage of vast amounts of video data. However, searching for a video from video databases takes time and effort. Searching by keywords depends on the prior manual effort of labeling all the videos. Thus, effective content-based searching and retrieval based on multimedia video similarity is an important research goal. Several content-based searching and retrieval systems have been proposed and discussed in the literature [75]-[78].

A general block diagram of these approaches is shown in Fig. 1.23. Shot division and keyframe extraction are the introductory and most imperative steps in content-based video searching and retrieval. Shot detection approaches were addressed in [19]. Several ways have been proposed to extract keyframes [79]-[82]. A basic strategy is to extract a shot's first and last frame as key frames [79]. In [80], key frame extraction has been performed using perceived motion energy and a triangle model. In [81], clustering of the frames and entropy of the image is used to extract the keyframes. The next step after extracting key frames is the feature computation of an image. The image color, texture, and shape features are very important for low-level image feature extraction. Color and texture can be extracted directly from the pixel values of an image. However, there are added advantages to extracting features from their compressed form because that is how they are usually stored [83]-[85]. Since the principal objectives of compression and indexing are data extraction and compact representation, exploiting the commonalities between the two approaches is evident.

In image and video-based searching and retrieval, we have emphasized the

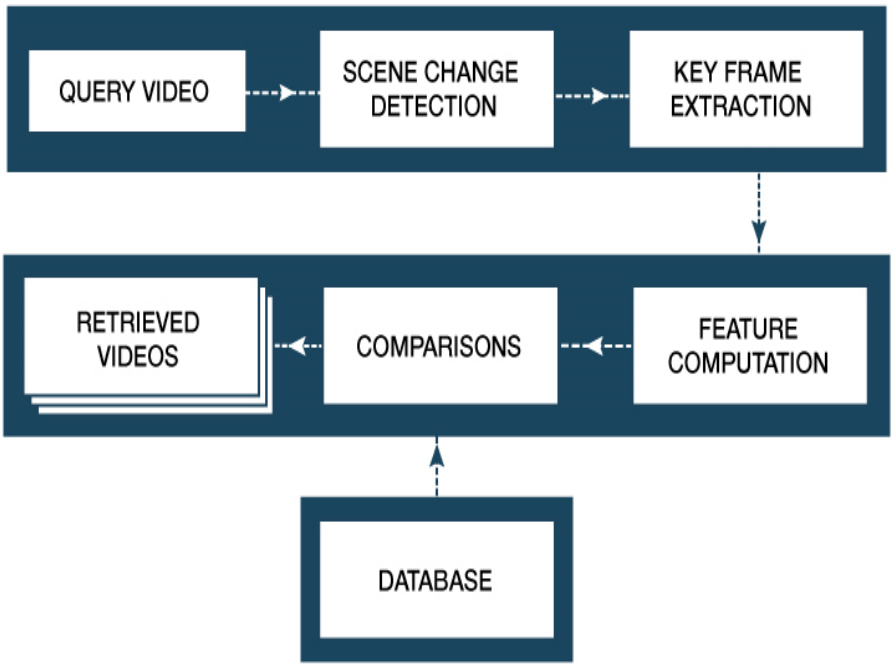


Fig. 1.23: Block diagram of a typical video retrieval system.

parameters of edge detection and edge similarity of the frames between broadcast news and video shot, and we have also implemented the SSIM algorithm [74] for refining our edge similarity algorithm. While structuring our algorithm, we have gone through various research papers by researchers in this field of image processing. Some of the notable work of some researchers based on broadcast news and edge detection algorithms are discussed in the literature [65]-[74].

In [65], CC. Ko and W.M. Xie performed shot boundary detection using two features: (i) Shot transition detection, where they initially initialized normalized color histogram intersection based on HSB (hue, saturation, brightness), and then employed edge exchanging ratio and threshold selection to identify shot transitions. (ii) Anchorperson shot identification and categorization, where they identified anchorpersons based on edges and categorized them into segments.

In [66], P. Melin *et al.*, constructed a method for edge detection based on morphological gradient and type-2 fuzzy logic. Using type-2 fuzzy logic, they differentiated the image based on planes and their mathematical calculations.

In [67], Ganesan P and G. Sajiv thoroughly analyzed different edge detection methods, explaining each algorithm's performance in image processing. They performed edge detection using a Robert edge detector, Sobel edge detector, Prewitt edge detector, Laplacian of Gaussian Edge Detector, Canny edge detector, and wavelet-based edge detection.

In [68], Zhou Wang *et al.*, proposed a clear method based on the Structural Similarity Index (SSIM). They conducted a detailed analysis of image quality assessment, considering error sensitivity and explaining each step of the process from the pre-processing stage to limitations. They then presented the clear use of SSIM and how it helps analyze an image based on its structure. In [69] Divakar Yadav *et al.*, present an approach to image processing in the medical diagnosis of bones. They took edge detection as a significant parameter. They first filtered the image using a linear filter, then applied the Sobel, canny, and log edge detectors and compared the result for the desired output. In [70] A.W.M. Smeulders *et al.*, has used a method of video searching. It uses an algorithm proposed by his team called Media Mill Challenge, where the generic video indexing problem is into a visual-only, textual-only, early fusion, late fusion, and combined analysis experiment. The paper is based on supervised learning, known as the SVM classifier. In [71], A. Vyas, R. Kannao, and V. Bhargava present a commercial and non-commercial video characterization approach. They devised an algorithm based on features of the video like the Text Masked Edge Change Ratio (ECR), Text Masked Frame Difference (FD), Shot Length (SL), and Overlaid Text Distribution (OTD). They also emphasized the audio features of the video, considering parameters like high music content, faster change, and higher volume in a commercial one. They also analyze the contextual analysis in the frames of commercial and non-commercial ones. They rectified the shot boundaries and the Block Level Post Processing in their post-processing part. They experimented with their algorithm on NDTV, TIMES NOW, and CNN-IBN news. In [72], Tamanna Sahoo and Prof. Sandipan Pine put their analysis based on edge detection where they divided into Step edges, Concave slope edges, convex

slope edges, Roof edges, Valley edges, and Staircase edges. They further worked on the edge structure, edge orientation, and environmental noises. Furthermore, they worked on applying the edge detector and finding the most efficient result. In [73], Y. Song, W. Wang, and F. Guo developed an algorithm for news video story segmentation fusing multi features, including audio and visual. They first detected the anchor person shot for marking the beginning of the story and detected topic captions between anchor person shots. They also detected silence clips and voice features of anchor person shots and fused all these parameters for segmenting news stories.

Methods of scene classification proposed in the literature include the use of wavelet features [83], sub-band energy [84], tree structures [85], and discrete cosine transform (DCT) features[86]. Videos are then represented by high-dimensional vectors of the extracted visual features of the frames [87]. High-dimensional indexing has recently received extensive study and remains an open research area. The multi-frame video representation further increases the problem's complexity. In the video retrieval literature, while most work has emphasized accuracy issues [88]-[90], more work must be reported on efficient video indexing and query processing. In the VA-file [91] and OVA-file method [92], data approximations of vectors are used to reduce the complexity. In these two methods [91], [92], the database consists of a vector file and an approximation file. The query is given as a feature, and its approximation is computed. Initially, its approximation is used to filter the vectors, and then the filtered vectors are compared with the original query vector. Finally, similar videos are retrieved. The OVA file creates an ordered approximation file in which the approximation vectors are clustered in groups or slices according to their positions in the data space. First, the clusters are filtered according to their centroids, and the clusters close to the query are visited. In addition, filtration is performed using the approximation file. After two stages of filtration, the remaining vectors are compared with the query vector using a distance metric. The VA-file [91] is the current state of the art but requires much computation. The OVA-file

system [92] requires less computation but offers less accurate video retrieval.

1.5 Research Gap

The literature review shows that most spatial domain techniques considered multiple cues of training data and employed some pre-defined models. In some other works, an anchorperson shot was considered as a cue for story segmentation. Some earlier methods used closed caption text, which is often less readily available. The spatial-temporal features of a news channel are caption texts, overlaid texts, logo position, anchor person, etc., However, the earlier methods do not consider the contents within text boxes, which could have been employed for shot segmentation and categorization. The methods described in the literature can not perform well when the news video has a split-screen configuration.

1.6 Motivation and Objectives

It is important to develop an integrated system for segmenting news video shots, categorizing content, indexing information, segmenting stories, and facilitating retrieval. Various news channels employ distinct spatial and temporal structures for broadcasting news. Therefore, the development of a universal method for shot and subsequent story segmentation holds significance. Different news channels present the same news from various perspectives. As a result, organizing the identical news presented by different news channels in a well-categorized manner becomes an essential task. Though several video processing units are present, one that does the three segmentation functions, categorization, and indexation is absent. Hence, an automatic video segmentation, categorization, and indexation system of news programs is required..

The main objectives of this research is to extract a set of independent news stories from a continuous stream of news videos broadcasted in a TV channel. Important objectives are enumerated as follows:

1. To segment news videos into individual shots.
2. To categorize news videos into commercial and news shots.
3. To combine news shots with similar news contents to generate individual stories in a multi-modal frameworks.
4. To search and retrieve news stories using visual and textual information.
5. To search and retrieve news stories using visual and textual information.

1.7 Outline of the thesis

The research work is classified into six chapters. Chapter 1 describes a brief introduction to the research work, briefly describing feature selection from the spatiotemporal structure of a news frame, discusses the literature review motivation and problem formulation, and finally lists our research objectives. Chapter 2 describes the news video shot segmentation and categorization. For shot segmentation, the YIQ model is used. Categorization is achieved by inspecting certain spatial features, primarily text blocks and color content, in the frames of the news shots. Based on the results of the inspections, the shots are categorized. Chapter 3 proposes a novel indexing method based on the text appearing in a news frame. The indexed shots corresponding to similar news content are combined to form individual videos depicting a single story. Chapter 4 gives a brief overview of news video story segmentation and indexing using a multimodal framework. Chapter 5 discusses the searching and retrieval algorithms. Finally, in Chapter 6, the thesis is concluded with a discussion of the results and future work. .

2

Broadcast News Video Shot Segmentation and Categorization

Contents

2.1 Introduction

The main objective of our research was to take a continuous stream of broadcast news video as input and segregate the commercial and news sections after segmenting

out the entire news video into shots. In this research, we address a combination of different methods to segment a news video into shots and categorize them into news and commercial shots. First, with the help of our shot detection algorithm, we detect all the cuts, fade in/out, dissolve the video, and segment the video based on these boundaries. The segmented shots are then categorized into news and commercial shots with the help of our shot categorization algorithm.

2.2 Shot Segmentation Algorithm

The frames near a shot boundary exhibit certain characteristic features such as illumination change, the information content of a frame, etc., which can be used to determine shot transitions. Shot segmentation mainly involves modeling of these transitions and their identifications to determine the presence of a boundary. Chapter 1 defines different spatial and temporal features used in the shot segmentation algorithm. The shot segmentation algorithm compares the histogram between two consecutive frames based on pre-defined and adaptive thresholds. A shot boundary is detected if the similarity measure is less than the threshold. This section mainly discusses the theoretical aspects required for the shot segmentation in news videos, followed by the mathematical analysis to support the cause of the algorithm.

Based on a set threshold, the shot segmentation algorithm compares the histogram between two consecutive frames. A shot boundary is detected if the similarity measure is less than the threshold. However, this method detects only abrupt changes in scenes, *i.e.,* cuts. It can also detect dissolves when the majority of the content in the frame has dissolved to give way to a new content. However, we need to proceed to a second step to detect a fades. In this second step, we further check the detected shots for fade-ins/fade-outs with the help of the mean luminance curve

2.2.1 Feature description: Histogram

The proper selection of features for the detection of shot boundaries is very important. In this algorithm, we used the histogram of each frame for comparisons. A histogram of a frame is plotted with the y-axis representing the number of pixels and the x-axis representing k^{th} level of intensity. Thus, a particular point in a histogram plot represents the number of pixels having intensity level k.

2.2.2 2D histogram extraction

The histogram of a YIQ image is of a 3D nature. The YIQ model of colour is obtained from the RGB model using the following equation:

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.581 & 0.114 \\ 0.596 & -0.274 & -0.322 \\ 0.211 & -0.523 & 0.321 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{2.1}$$

1. The three matrices viz., Y, I, and Q are extracted to obtain three independent matrices.

2. The maximum value of Y, I, Q is obtained. Let this value be m .

3. Let the total number of bins required in the expected histogram be n. The ceiling values of $Y(x, y)$, $I(x, y)$ and $Q(x, y)$, will convert all the intensity values of the image into numbers of a n base system.

$$Y(x, y) \times \frac{n}{m} = y_i \tag{2.2}$$

$$I(x, y) \times \frac{n}{m} = i_i \tag{2.3}$$

$$Q(x, y) \times \frac{n}{m} = q_i \tag{2.4}$$

4. Let

$$I_i(x, y) = n^2 \times y_i + n \times i_i + q_i \tag{2.5}$$

Where,

$I_i(x, y)$ will give a number in the n base system that is representative of the intensity value at $f(x, y)$.

5. Thus, we will obtain a matrix I, and plotting the histogram of the values in this matrix will give the required $2D$ matrix of the YIQ image.

2.2.3 Developement of a Y-mean curve

The Y-matrix of each frame represents the intensity of the frame *i.e.,* $Y(x, y)$ represents the intensity of the frame at position (x, y). For plotting the Y-mean curve, we compute the average of the Y value for each frame with the help of the following formula:

$$Y^i_{mean} = \frac{1}{M \times N} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} Y(x, y) \tag{2.6}$$

Where,

Y^i_{mean} represents the Y-mean value for the i^{th} frame. $M \times N$ the size of the frame. The Y-curve is plotted with the frame number along the x-axis and the Y-mean value along the y-axis.

2.2.4 Standard for measurement of similarity

If we consider the histogram for each frame as a representative of a vector plane in a feature space, then the closeness of the two planes can be used as magnitude for measurement of similarity. The closeness of the two planes can be computed equal to the cosine of the angle between the planes. If the two frames are similar, then the cosine of the angle will be approximately equal to unity. As the magnitude will decrease, the similarity will decrease. The cosine of the angle between any two vectors A and B can be computed with the help of the following formula:

$$\cos\theta = \frac{A.B}{|A||B|} \tag{2.7}$$

If $\cos\theta = 1$ *i.e.,* $\theta = 0$ then A and B are exactly similar but if $\cos\theta = 0$ *i.e.,* $\theta = 90$ then A and B are completely different.

2.3 Algorithm for Shot Segmentation

We developed an adaptive threshold-based multi-step approach to detect shot boundaries based on a YIQ colour model. The shot detection algorithm based on a YIQ color consists of three steps. The first step aims to extract a $2D$ histogram from the YIQ model of a frame. The shot boundaries are obtained on the differences in YIQ features from one frame to another. The second step is the detection of a cut. We used cosine comparisons of cumulative histograms in an n-base feature space between consecutive frames for cut detection. A similarity is detected if the cosine value exceeds a predefined threshold. The comparison between two histogram plots of two frames is shown in Fig. 2.1. The third step consists of cut detection based on fade in/out, where we detected cuts by identification of negative peaks in the mean luminance curve of the frames. In Fig. 2.2, we showed the Y-mean curve, whose $Y_{mean} \approx 0$. A negative peak identifies the presence of a monochromatic frame and marks the center of the fade in/out. The algorithm assigns all frames just before the monochromatic frame to the preceding shot and all frames after the monochromatic frame to the succeeding shots.

2.3.1 Cut detection algorithm

In our approach, we extract one primary global feature for every frame in the video: the frame histogram. Histograms represent the distribution of pixel intensities in the frame. Whenever there is a drastic change in the content of a pair of frames, the pixel intensity distribution changes significantly. This property of the histogram feature is exploited for detecting shot boundaries.

Various actions of the camera, such as zoom in/out or pan, can induce a significant change in the content of the frames, yet there is no on/off switch for the

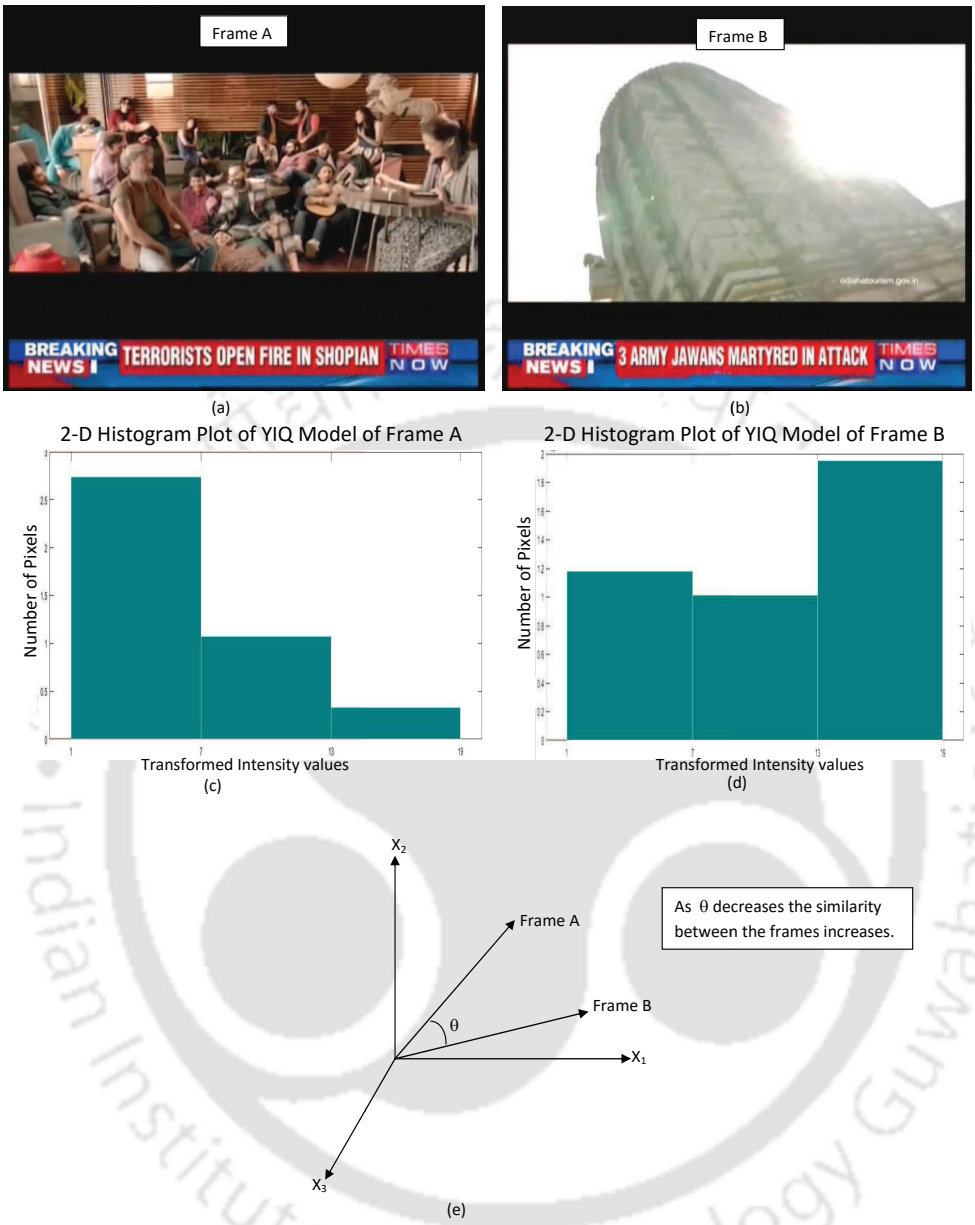


Fig. 2.1: Comparison of histogram plots of two frames. (a) Frame A (b) Frame B (c) Histogram Plot of Frame A (d) Histogram Plot of Frame B (e) Comparison of Frame A and Frame B in 3 dimensional system

camera, *i.e.,* no shot change. Therefore, the histograms of frames involved in zoom in/out or pan undergo a significant change that might cause misdetection of a shot boundary. To prevent such misdetection, a continuous running pattern has been used, and this is discussed in detail in Step 4 of the algorithm. The algorithm is

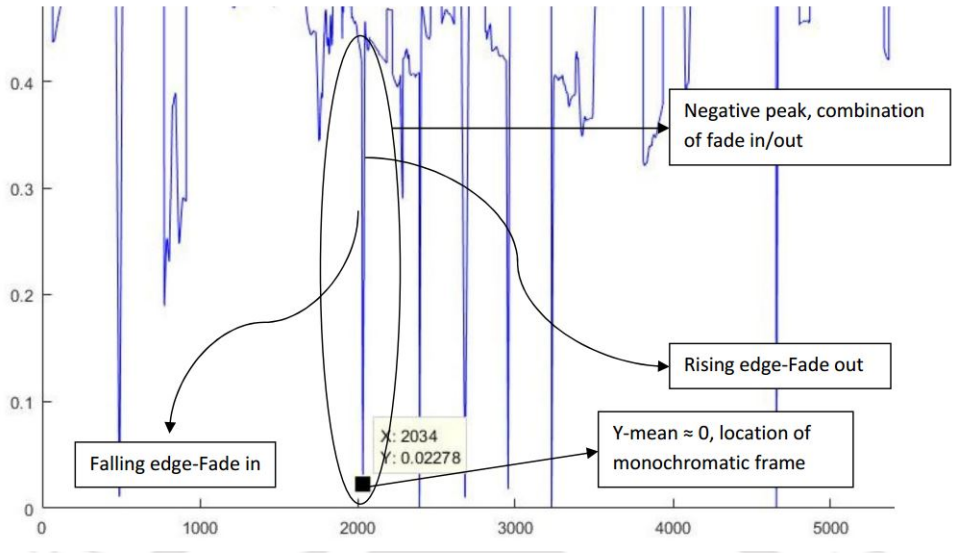


Fig. 2.2: Analysis of Y-mean Curve

discussed as follows.

1. Each video frame is extracted and processed through the steps discussed in Section 2.2.2 to extract the $2D$ histogram.

2. The histogram values of each frame are stored in a matrix A where the columns of A represent the frames of the video, while the rows of A represent the bin values of the histogram for each frame. Thus,

$$A = [a_{ij}] \tag{2.8}$$

   Where,
   i is the i_{th} bin of the histogram of the j_{th} frame

3. The i_{th} column A is compared with the $(i+1)_{th}$ column with the help of dot product. A threshold T is used for comparison such that

$$S = \frac{A(j)_{new}.A(j+1)}{|A(j)_{new}||A(j+1)|} \forall j \in (1, N-1), \tag{2.9}$$

   where,

N is the total number of frames

$A(j)$ and $A(j+1)$ are the respective columns of A and S is the cosine angle between the vectors $A(j)$ and $A(j+1)$

4. If $S \geq T$, the two frames j and $(j+1)$ are similar enough to belong to the same shot or a shot boundary is detected. If $S \geq T$, then the previously obtained histogram is averaged before comparison with the following consecutive frame histogram with the following equation:

$$A(j)_{new} = \frac{A(j-1) + A(j)_{old}}{2} \tag{2.10}$$

By the end of this algorithm, all the cuts in the considered video will be detected.

2.3.2 Detection of fade in/out

For fade detection, we use the YIQ system of colour. The Y component gives the chrominance of each frame. For a transition consisting of Fade in and Fade out, the Y component decreases and increases linearly respectively to or from a monochromatic frame whose $Y_{mean} \approx 0$. It has been observed that when plotting a curve depicting the mean luminance values of all the frames in the video, there are noticeable extreme negative peaks in cases where both fade-in and fade-out effects are applied, as shown in the Fig. 2.2. Thus, our algorithm mainly involves the detection of this frame, and the frames before it are assimilated to the previous shot, and the frames succeeding it are joined to the next shot.

1. The cut detection algorithm classifies all the frames involved in fade-in or fade-out as single or two to three-frame shots because the histogram values change abruptly in fade owing to the change in luminance. Hence, our algorithm detects them as shots containing very few frames. So, we take out all the shots that contain frames less than the frame rate of the video under consideration.

2. The mean of the Y component in each frame is computed accordingly.

3. Plotting the curve of the mean of the Y components, the point of local minima of the curve whose peak height is greater than L are obtained.(Fig. 2.2) This minima point is the location of the monochromatic frame, say, O.

4. All frames preceding point O in the immediately preceding negative gradient of the curve are added following the last frame of the preceding shot where point O is located.

5. Frames following point O in the immediately succeeding positive gradient of the curve are added after the first frame of the succeeding shot in which point O is located.

Thus, our proposed shot segmentation algorithm should successfully detect the presence of all shot boundaries through segmentation at fade-in/out, dissolve, and cut. This also means segmenting the different segments of a news video, *viz.,* commercial and news shots. In the case of split screens, our algorithm detects a shot boundary when the majority of the screen content changes. This is so because when the majority of the screen changes, the histogram plot for the intensity values changes, and the presented algorithm detects this change.

The discussed algorithm, based on changes in YIQ properties, detects a shot boundary. However, our algorithm can only partially detect shots for the following scenarios.

Case I : The visual content should remain the same during one shot or change slowly. However, during a shot transition, an abrupt change is occasionally seen between two consecutive frames due to the change of visual content in the text box. Shot transitions should occur due to the changes in the visual content of the body of the news frame rather than the visual content of the whole frame

To consider this problem, each frame is pre-processed to remove text boxes from the body of the news frame. To remove text blocks, all the rectangles present in the frames are detected. The rectangles having a width approximately equal to that of the input frame and a particular height are obtained. Out of these rectangles, the

frames' top and bottom rectangles are considered text boxes. The result is cross-checked using Optical Character Recognition (OCR)[10]. Detected rectangles in a binary image are shown in Fig. 2.3. These text boxes are then cropped from the input frame. A frame with and without text boxes is shown in Fig. 2.4a and Fig.

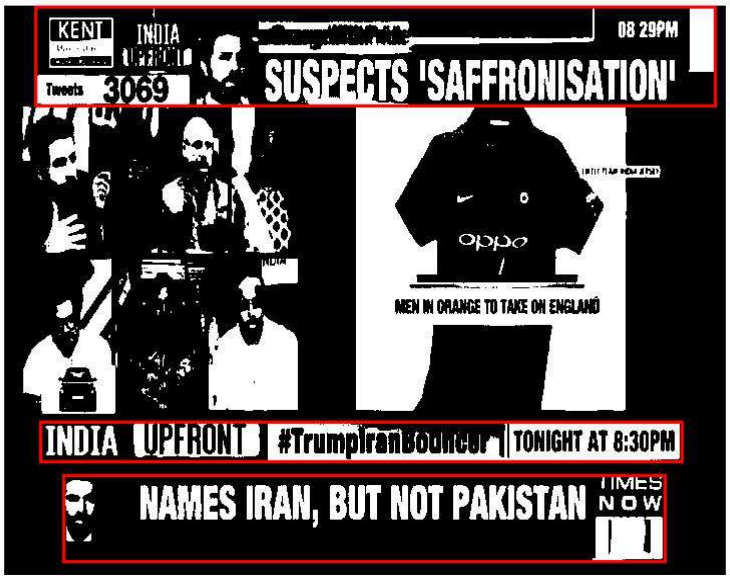


Fig. 2.3: Detected text boxes in a binary image

2.4b, respectively.

Case *II* : In the case of split-screen, as shown in Fig. 2.5, the frame is divided into multiple parts to display the field shots and anchorperson shots together. When a split-screen is used, multiple shots of an incident may be displayed in the field shot portion, while the anchor-person shot remains the same. These shots belong to the same story. Due to such changes in the major part of the frame, some false shot transitions are detected.

To solve this issue, the rectangles within a frame in which different shots are displayed are detected. The text boxes are removed from the frame. The frame is divided into rectangle boundaries if the number of rectangles detected is 2 or 3. If the number of rectangles is less than one or more than three, the screen is

(a) A frame with a text box

(b) A frame without a text box

Fig. 2.4: A splitscreen frame with and without text boxes

devided into two halves. Now, a frame's first and second parts are compared to the corresponding parts of the succeeding frames. A shot boundary is detected only when transitions are detected in both halves/rectangles. Again, the number of rectangles is also used as a parameter for shot transition detection. If the number of rectangles in two consecutive frames does not match, it is detected as a shot boundary without further calculations. This parameter is given the first preference in shot transition detection.

Fig. 2.5: Detected rectangles in a binary image

Fig. 2.6: Divided rectangles of a split-screen

Case III : The successful implementation of the cut detection algorithm depends on the optimum selection of the threshold value for comparisons. However, considering a fixed value as a threshold could be more efficient. This work suggests a new method, which calculates the threshold by itself. In this method, a band is considered a instead of a single value. If the difference in angle between two frames lies in this band, then a cut transition is detected.

$$average1 = (average1 \times (cut\_no - 1) + angle1)/cut\_no \tag{2.11}$$

$$average2 = (average2 \times (cut\_no - 1) + angle2)/cut\_no \tag{2.12}$$

$$threshold = min(average1, average2) \tag{2.13}$$

Here,

cut_no = total number of cut transition up to the current frame

In our approach, up to ten transitions, the average of the angles at which the transitions take place is calculated for each half in the split screen configuration. The minimum of these two is considered as the threshold. From the next frame, a transition is detected if the angle is more significant than the threshold. This threshold is the average of the angles for previous transitions. However, for some of the following transitions, the value of the angle may be slightly less than the calculated threshold. To consider this issue, when the angle is less than the previously calculated threshold, a shot boundary is detected if the difference between the previously calculated threshold and the cosine angle is less than one-third of that threshold. For each half of the frame, if any of the above conditions are satisfied, the shot is segmented. The threshold is then modified by including this angle in 2.13. Fig. 2.7 shows the adaptive threshold band. The threshold value is fixed up to the first ten transitions, as shown in the Fig. 2.7

The complete algorithm for shot segmentation can be classified into six parts, which are an algorithm for removal of the text boxes, finding a $2D$ image histogram from the YIQ image model, split-screen detection, cut transition detection, detection based on a threshold band, and fade in/out transition detection.

2.4 Shot Categorization Algorithm

Commercials constitute a significant chunk of the videos aired on a news channel. These videos are substantially different from the ones containing news, and the inclusion of commercials in between the news videos makes it challenging to segment the stories. So, to overcome such discrepancies, news videos are filtered out, and the remnant mainly consists of commercials. The main aim of the shot categorization algorithm is to categorize the segmented shots into news and commercials.

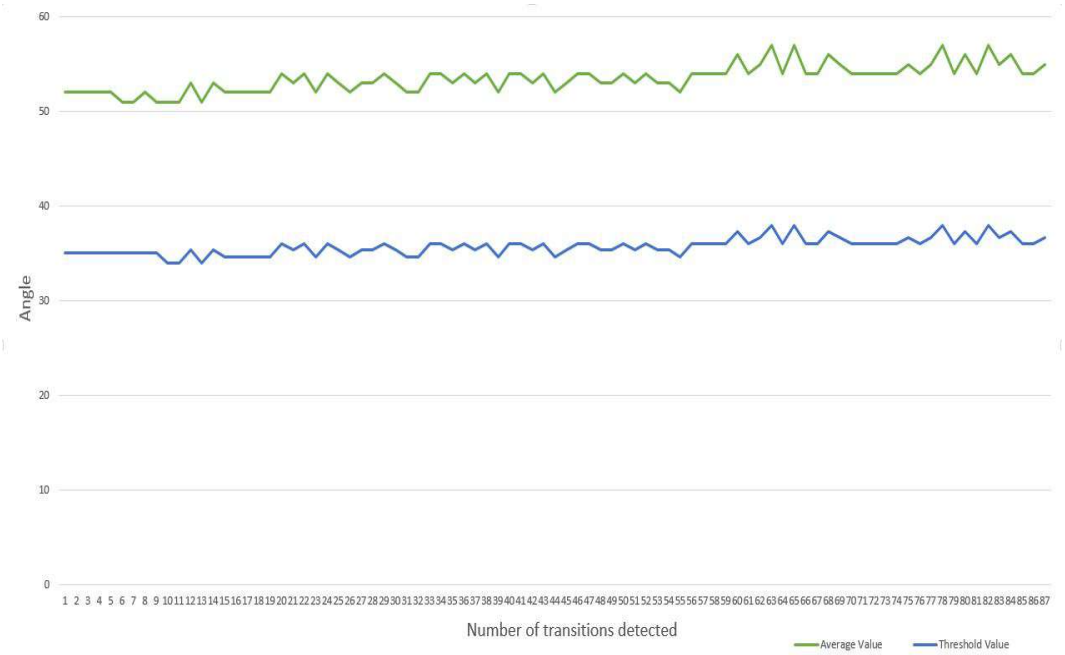


Fig. 2.7: Adaptive threshold band

To achieve this, we utilize the presence of text blocks in the spatial arrangement of the news frame and color content in the news frame selected as a feature for classification.

2.4.1 Feature description: Qualities and criterion

The best feature selection is necessary for the best categorization algorithm. Many features of a news program separate it from the other categories of videos. Some of these features include text blocks, color content in the news frame, logo, Animation shots, etc. However, only some of the features are useful for the discussion. The number of text blocks presented in different frame sections is notable. The number of text blocks in the commercial section is less than in the news section, shown in Fig. 1.3 and Fig. 1.4. It is observed that the total content of red and blue in a news frame is much higher than in a commercial frame. An analysis has been made on the content of these two colours in several randomly chosen news and commercial frames from four different channels. The frames are first converted into

HSV colour model. HSV is a cylindrical colour model where H represents hue *i.e.,* the actual colour in a circle, S represents saturation *i.e.,* the grey colour content, and V represents value *i.e.,* the brightness of a pixel. In the hue circle, 0 degrees represent red, 120 degrees represent green, and 240 degrees represent blue colour.

The percentage of red and blue pixels in a frame is found using (2.15) and (2.16), respectively. The average values of each channel from the analysis are given in Table 2.1.

$$Tag = \begin{cases} red, & \text{If H<10 or H>350, S>0.5 \& V >0.5} \\ blue, & \text{If 230<H<250 , S>0.5 \& V>0.5} \\ other, & \text{Otherwise} \end{cases} \tag{2.14}$$

$$\%\ red\ pixel = \frac{R}{N} \times 100\% \tag{2.15}$$

$$\%\ blue\ pixel = \frac{B}{N} \times 100\% \tag{2.16}$$

where,

R = Total number of red pixels

B = Total number of blue pixels

N = Total number of pixels in the frame

Tab. 2.1: Colour content analysis

News Channel	**News frame**			**Commercial frame**		
	Red(%)	Blue(%)	Total(%)	Red(%)	Blue(%)	Total(%)
Times Now	18.41	23.32	**41.74**	2.79	5.04	**7.83**
CNN-News 18	27.55	2.98	**30.52**	5.24	3.86	**9.16**
India Today	36.82	15.34	**52.16**	5.33	5.14	**10.67**
CNBC-TV18	39.10	11.59	**50.70**	2.27	9.04	**11.31**

2.4.2 Shot categorization algorithm based on text block

The algorithm based on text block first determines the number of text blocks present in a frame with the help of edge detection and Optical Character Recognition [12, 13]. As the number of text blocks and their location varies for different news channels, the use of Sobel Mask and OCR make the algorithm unsupervised *i.e.,* we need not define a region of interest for text block detection. Consequently, based on comparison with a predefined threshold, segmented shots are classified. This algorithm uses a range of features optimized to achieve better accuracy at minimal computational complexcity. These steps have also been used in Section 3.4.7 of Chapter 3.4 with necessary modifications. The steps of the algorithm are discussed in detail:

1. The frames are extracted from the news shot in the first step.

2. The frames are convolved with a Sobel Horizontal Mask (Figure 3.9a) to detect the presence of horizontal lines in the frame. The obtained values of g_x (Equation 3.4) are compared with a set threshold T_e as shown below

$$g_x \in L \iff g_x \geq T_e \tag{2.17}$$

where,

L is a set of horizontal lines. In other words L is a matrix given by the following equation

$$L(x,y) = \begin{cases} 1, \text{ if}(x,y) \text{ lies in a line.} \\ 0, \text{ if}(x,y) \text{ doesn't lie in a line.} \end{cases} \tag{2.18}$$

3. To mark a reference for a comparison, each halved frame (F_{new}) is marked with a perfectly horizontal line at the top and bottom of the frame.

4. In order to prevent the doubly detected lines or to convert slightly slanted

lines to perfectly horizontal lines, the initially detected lines are averaged over its neighborhood. This can be expressed mathematically as:

$$L(i, y) = L(i - h, y) + L(i + h, y) \tag{2.19}$$

In our algorithm we set the value of $h = 10$.

5. All detected lines are considered as edges if their length is greater than 50% of the total length of the frame, *i.e.,*

$$L(x, y) \in E \iff \sum_{x=1}^{N} L(x, y) \geq 0.5 \times N \tag{2.20}$$

where,

N is the width of the frame and E is set of valid edges i.e. $E \subseteq L$

6. The space between two consecutive edges is considered to be a valid text block if the height of the text block is greater than 0.1% and less than 25% of the total height of the frame. To achieve this, it takes the first edge at the top as reference. From the top edge it goes on to check downwards until it finds a valid block height which must lie between 0.1% to 25% of the total frame height. On reaching a valid edge it repeats the same procedure taking the last detected valid block end as reference.

$$E_k(x, y) \in B \iff 0.01 \times M \leq \sum_{y=a}^{b} E(x, y) \leq 0.25 \times \ M \tag{2.21}$$

where,

$E_k(x, y)$ is the k^{th} iteration and a is the block end (edge) of $(k-1)^{th}$ iteration and b is the block end of the k^{th} iteration.

7. All the validated text blocks are run through the OCR algorithm. A text block is accepted for further analysis if the word confidence of the text block is greater than a set threshold.

8. Steps $2-7$ are repeated for all frames of the shot. The number of validated text blocks are stored in a vector TB such that

$$TB_s = [a_{ij}] \tag{2.22}$$

where,

a_{ij} is the number of text blocks in j_{th} frame of shot 's'.

9. Mode of a vector is defined as the most repeated element of the vector. Mode of TB_s will provide the common number of text blocks present in maximum number of frames of the shot. This is the threshold of comparison for categorization. Let this be M_s.

10. If $M_s \leq N_{blocks}$, where N_{blocks} is the number of text blocks present in a particular channel during the news section, then shot 's' is a commercial or else it is a news shot.

11. Steps 1 to 10 are repeated for all the segmented shots of the news program.

12. However in some channels the text blocks are not permanent *i.e.,* the text blocks appear and disappear in the course of the news section and during the times of commercial text blocks are at all absent. In such cases a shot is classified to be a news is classified to be a news shot if a text block appears continuously for more than 50% of the total length of the shot

2.4.3 Shot categorization algorithm based on colour content

The algorithm first calculates the total percentage of the red and blue color for each frame of the news video. This value is then compared with a threshold. The frames with a value higher than the threshold are considered news frames; otherwise, they are considered commercial.

However, this condition may not hold in the case of an anchorperson shot. Some commercials also do not satisfy this condition, leading to errors. To address

this drawback, a rectification algorithm is employed, which verifies the accuracy of the previous detection process.

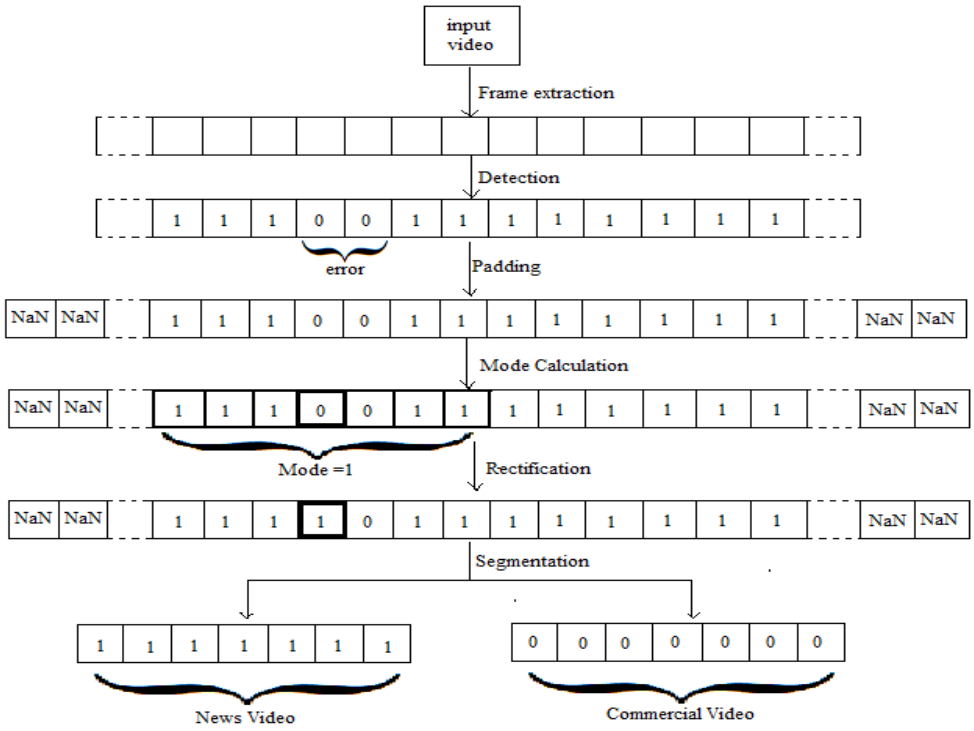


Fig. 2.8: Visual representation of the algorithm for filtering of commercial

First, the detection algorithm is used, and each frame is assigned a 'news' or 'commercial' tag. A news frame is represented by 1, and a commercial frame by0. It is considered that a particular type of frame is displayed continuously for at least 10 seconds *i.e.,* 250 frames for a 25-fps video. The rectification algorithm takes the 125 frames before and after the frame under processing. However, for a few frames at the start of the video, it is only possible to consider 125 frames after it. So, a null value is padded in the matrix. After that, the mode of these 251 frames is calculated. If the mode is 1, it represents that most of the frames in that range as a news frame; otherwise, it is consider a commercial frame. The mode reassigns the current frame. After completing this process, the input video is categorized into two parts: news video, which consists of the frames with the tag '1', and commercial with the tag '0'.The visual representation of the algorithm is shown in Fig. 2.8

Algorithm 1 Algorithm for filtering of commercial based on colour content

Result: Two segmented videos, news and commercial

N← Number of frames in the input video
i ← 1
j ← 126
flag← 1-dimentional array of size N with all element equal to one;
while $i \leq N$ **do**
Extract a fram of the video;
Convert the frame into HSV color model;
Find the percentage of red and blue pixels of the image using(2.15) and (2.16);
R← Percentage of red pixels;
B← Percentage of blue pixels;
if $R + B \leq 15$ **then**
flag(i,1)←0
end
i← i+1
end while
Pad the flag matrix with 125 columns at the front and 125 columns at the end with all elements 'NaN';
while $j \leq N$ **do**
Take the j^{th} element of the padded matrix with 125 elements before and after it;
Find the mode of these 125 elements;
Replace the $(j-25)^{th}$ term of flag matrix with the mode;
$j \leftarrow j + 1$
end while
Declare a video writer for news part;
Declare a video writer for commercial part;
$i \leftarrow 0$;
while $i \leq N$ **do**
if flag(i,l)==1 **then**
Write the i^{th} frame of the input video into the news segment;
else
Write the i^{th} frame of the input video into the commercial segment;
end
i← i+1
end while
Release the two video writer;

2.5 Results and Discussions

We recorded almost two hours of a news program from Times Now and CNN IBN channels comprising all the previously mentioned spatial and temporal contents/features. The news was recorded directly from a TV using an i-ball clairo TV recorder in MPEG-1 format. The video has twenty-five frames per second and a total of 180000 frames. The shot boundaries in the video are determined, and shots are segmented successfully. Three parameters are used for estimating the performance: precision, recall, and F-measure[11].

- True positive (TP): An event is identified as true positive if it has actually taken place as well as identified correctly by the algorithm.
- False positive (FP): An event is identified as false positive if it has not actually taken place but is identified by the algorithm to have taken place.
- False negative (FN): An event is identified as false negative if it has actually taken place but is not identified by the algorithm to have taken place.

Precision, Recall and F-measure is defined as:-

- Precision (P): It is identified as the ratio of the number of events identified correctly by the algorithm to the total number of events identified. It can be determined using (2.23).

$$Precision(P) = TP/(TP + FP) \tag{2.23}$$

- Recall value (R): It is the ratio of the total number of events identified to the total number of true events. The formula for recall is given in (2.24).

$$Recall(R) = TP/(TP + FN) \tag{2.24}$$

Tab. 2.3: Results of Shot Segmentation Algorithm based YIQ model

Parameter	**Times Now**	**CNN IBN**
Number of shots identified correctly	522	983
Total Number of actual shots	695	1231
Total number of detected shots	561	1104
Recall	0.75	0.80
Precision	0.93	0.89
F-measure	0.83	0.84

- F-measure: It is defined as the weighted harmonic mean of precision and recall of the test. It can be determined using (2.25).

$$F - measure = 2PR/(P + R) \tag{2.25}$$

The analysis of the results is shown in Table 2.3

Our proposed shot segmentation algorithms based on the YIQ model should successfully detect the presence of all shot boundaries through segmentation at fade-in/out, dissolve, and cut. After consideration of split screen and threshold band algorithms, it shows improved results.

A comparison of results obtained by different authors on shot segmentation is given in Table 2.4. For the gradual transition range, both precision and recall degrade. Our results are comparable to all these findings, with a recall rate of 80% achieved at 89% precision for CNN IBN news data and 70% at 93% for Times Now news data.

The news categorization algorithm aims to categorize the segmented news shots into news and commercials based on the number of text blocks detected and the color content in a frame. The following two tables 2.5 and 2.6 show the result of the two algorithms, respectively.

Tab. 2.4: Comparison of Results for Shot Segmentation

Paper	**Recall(%)**	**Precision(%)**
Xinbo Gao *et al.*, [18]	96.48	98.07
U Gargi *et al.*, [93]	90	70
M. R. Naphade *et al.*, [94]	94.21	98.5
Y Avrithis *et al.*, [95]	97	95
J Mas *et al.*, [96]	84.7	80.6
P. P. Mahanta *et al.*, [97]	81	87
kr Krishna K. *et al.*, [98]	92.84	93.45
Johan S. *et al.*, [99]	90	82
Our Approach (CNN IBN news data)	80.00	89
Our Approach (Times Now news data)	70.00	93

Our shot categorization algorithm based on text and color features obtained highly satisfactory results.

2.6 Post Processing: Integration of Audio into the News Stories

We need to find the audio sample associated with a frame. This can be determined with the help of the following formula:

$$Number\ of\ audio\ samples\ per\ unit\ frame = \frac{Sampling\ Frequency}{Number\ of\ Frames\ per\ second} \quad (2.26)$$

Therefore audio sample associated with a news story beginning from Frame F_i and ending with F_j is given as:

$$Total\ number\ of\ Audio\ Samples = (j-i) \times Number\ of\ audio\ samples\ per\ unit\ frame \quad (2.27)$$

Tab. 2.5: Results of Shot Categorization Algorithm based on text feature

Parameter	**Commercial Shots**	**Commercial Shots**	**News Shots**	**News Shots**
	CNN IBN	Times Now	CNN IBN	Times Now
Number of Shots categorized correctly	307	379	666	229
Total number of actual shots	410	389	667	301
Total number of detected shots	308	458	769	259
Precision	0.9967	0.83	0.8660	0.88
Recall	0.7487	0.97	0.9985	0.76
F-Measure	0.8550	0.89	0.9343	0.81

Tab. 2.6: Result analysis for filtering of commercials based on color feature

Parameter	**Commercial**	**News**
Total number of actual frames	24777	155223
Total number of detected frames	23116	156884
Total number of frames identified correctly	22509	152955
Recall	90.84%	98.54%
Precision	97.37%	97.49%
F-measure	**93.99%**	**98.01%**

The actual process involves the extraction of the frames and audio samples separately from the news video. Then audio is integrated into the video with the help of Equations 2.26 and 2.27.

2.7 Conclusions

This work is the first step and initial approach toward modeling an unsupervised story segmentation model. As an output, we have modeled an algorithm for the efficient shot segmentation of news videos. A few test cases were input to the model, and the results were satisfactory. The algorithm gives quite satisfactory results. However, in some animations, the size of the rectangles of the split-screen varies as the frames proceed, or the rectangles' boundary is not properly defined due to the dissolve transition in one part of the split-screen. In such cases, the rectangles cannot be adequately detected, which induces an error in detecting the shot boundary. The algorithm also struggles when there is an animation in the background of text boxes; the text box detection is affected as the rectangle's outline is not adequately defined. Animations in between news videos cause an error as the animated frame provides no information, and they cause a series of consecutive shot segmentations. The false segmented videos have very few frames to contain information. The algorithm for fade detection filters out animations with less than five frames, but all the other animations still cause an error in shot segmentation.

3

News Video Indexing and Story Unit Segmentation using Text Cues

Contents

3.1 Introduction

News video indexing and story segmentation are important steps for creating a well-categorized, organized, and indexed video storage system. Such a system allows disciplined storage to cope with an increasing amount of data and easier retrieval for viewing in the future. Keeping this approach in mind, we proposed a news video indexing and story unit segmentation algorithm based on keywords obtained from the text displayed in the news frame during a news program. First, with the help of our shot categorization algorithm, we classify the news shots into commercials and news based on spatial features of a particular news channel. Then, with the help of our shot indexing algorithm, we detect the presence of words in each frame of a shot and index each shot with a keyword. With the help of our shot combination algorithm, we combine the shots with similar keywords to form individual stories. The results obtained in this work were satisfactory.

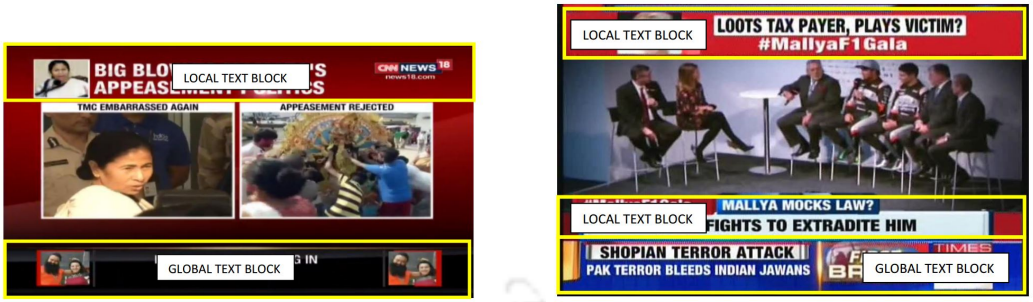
3.2 Feature Selection from Spatial Structure of a News Frame

The main idea of a news program is the maximum dispersal of information in the stipulated duration of the program. It achieves this not by mere news reading and video portrayal but with the help of many editing techniques that incorporate extra features such as text blocks, logos, and background videos into a news frame. This arrangement of different components in a news frame can be described as its spatial structure. One of the characteristic features of news videos is the presence of these text blocks. The text blocks contain text pertaining to various news shown in the TV program. These text blocks satisfy all the basic criteria for selecting a feature. The text blocks are global *i.e,.* they are present in most news channels,

local or international. They are extractable with the help of various techniques. The number of text blocks present varies for sections of the program. This helps us to set a threshold for comparison for different sections of the program. These text blocks can be used to extract text from them and be analyzed to define keywords for a news program. The keywords can be later used for index-based classification of news stories. The text block can be classified into Local text block, Global text block, and Scrolling text block, depending on the context of the text block. The classification and description of the text block have been described in section 1.2. All or at least a few of the mentioned blocks (Global and Local Text Block) are always present in all the news programs of different news channels. However, one or more text blocks are removed during a commercial to provide more screen area to advertise.

A text block's size depends on its content's portrayal. In a news frame, the Global Text block and Local Text block are of maximum size as they focus on the day's most important topics. On identification of the local text block, it can be used for extraction of keywords to tag the current news story being displayed.

There may be one or more local text blocks in a news frame. As observed in most news channels, a local text block lies separate, secluded from the other text blocks. Therefore, our algorithm detects the largest text block in either half of a news frame for identification. It can be classified as a local text block. This has been depicted in Fig. 3.1a and 3.1b. For particular news, the local text block periodically displays a series of two or more sentences. Therefore, each frame can be tagged with the words in the local text block that is present in that frame. After tagging each frame with their respective words, the most repeated words can be designated as keywords for that particular shot of which the frames are a part. However, the contents of the local text block are only sometimes consistent with text. The following section discusses the variation of contents in a local text block.



(a) News frame from CNN News 18 *(b)* News frame from TIMES NOW

Fig. 3.1: Location of Local Text Blocks

3.2.1 Variation of contents in a local text block

The local text block may display a periodic sequence of two or more sentences for a particular news shot. There are two possible cases of textual transition in a local text block of the same shot.

Case i : To and fro transition of same sentence : This case is a description of a scenario when the text before and after the transition remains the same, as shown in Fig. 3.2a, 3.2b and 3.2c. In this case, the frames that are a part of the transition as such in Fig. 3.2b are tagged with the text of either the preceding or subsequent frames.

Case ii : To and fro transition of different sentence This case is a description of a scenario when the text before and after the transition is different, as shown in Fig. 3.3a, 3.3b and 3.3c. In this case, the frames that are a part of the transition as such in Fig. 3.3b are tagged with the text of the subsequent frames.

The above cases are also generally applied when there is no recognizable text in the local text block of a frame. This enables us to tag each frame without any loss of information.

(a) Before Transition

(b) During Transition

(c) After Transition

Fig. 3.2: To and fro transition of same sentence

(a) Before Transition

(b) During Transition

(c) After Transition

Fig. 3.3: To and fro transition of one sentence to another

3.2.2 Indexing a shot

After each frame in a shot has been tagged with the words present in the local text block of that frame, an entire shot can be indexed based on those keywords. In order to achieve this, we need to determine the repeated words. The following formula is used to find out the repetition of a particular word:

$$n_i = f - \sum_{k=1}^{N} R_k \tag{3.1}$$

where,

n_i is the number of times of repetition of the i^{th} word

f is the number of frames in which the particular word occurs.

N is the total number of frames

R_k is the number of times i^{th} word is repeated

in a particular sentence of a local text block in the k^{th} frame

A particular word repeated for n times is an index for the shot if it satisfies the following condition.

$$n_i \geq 0.6 \times N \tag{3.2}$$

Equation 3.2 means that for a particular word i to be considered as an index, it must occur in more than 60 % of the frames. In other words, i^{th} word must appear in the local text block of the news shot for more than 60 % of the total length of the shot. At the end of this process, each shot will be tagged with i words, each of which will have an occurrence in more than 60% of the total number of frames.

3.3 Feature Selection from Temporal Structure of a News Program

In this section, the temporal structure of the news section of a news video will be exclusively discussed. The discussion will follow in context with the text variation

in a local text block as the news story changes. A news story can be defined as a collection of news shots that focus on the same particular news. Shots are obtained based on visual content change, while stories are based on semantic content change. Therefore, to obtain news stories, we need to analyze the news content of a news shot. News shots with similar content will be combined to form news stories. The news content of a news shot will be obtained based on the theoretical groundwork discussed in Chapter 1.2.1. The following section will briefly review the temporal structure of a news program in general.

3.3.1 Variation of content in news section

A news program broadcasts news from different genres and different places. This leads to frequent variations of both visual and semantic content in a news program. In order to achieve good classification, this variation needs to be understood in detail. This will help to define a proper relationship with a story boundary between two consecutive news stories. The variation in a news section can be classified into two broad classes, as discussed below:

1. *Variation in Visual Content* The visual content in a news video changes abruptly when there is a change in the location from where the news is being broadcast. This usually happens during a transition between a Studio or Anchorperson shot to a field shot. A change in visual content might also occur if an animation shot is induced in between. However, the vital thing to be noted here is that the change in visual content does not necessarily determine a story boundary. However, the visual content will always change in case of a story boundary.

2. *Variation in News Content* The news content in a news video changes when there is a transition from one news to another. Along with this, the visual content also changes in accordance with the news. The content in the local text block particularly changes when the news story changes. This is an inherent property of local text block to focus on the current news being broadcast. This feature

has been explicitly used to determine the location of a news story boundary.

Thus, it is evident from the above discussion that it is very important to determine the nature of variation between two shots to establish a valid story boundary. To identify a story boundary, we need to verify whether there is a variation in the news content. This can be achieved through the matching of keywords between two consecutive shots. The keywords for a shot can be obtained as described in Section 3.2.2.

3.3.2 Determination of variation in news content

In order to determine the variation in news content, we need to determine the news currently being broadcast. This can be done by processing the audio content in the news video or through visual cues being displayed. Audio processing induces a lot of complexity and unwanted noise. So, the efficient way is to analyze the visual cues in the frames of news video. One of the strongest cues is posed by the textual content in the local text block of a news video. As the news changes, the sentence in the local text block also changes. The change in sentence will be reflected in the shot index obtained for a shot as described in Section 3.2.2 of Chapter 1.2.1. Thus it can be said with great accuracy that two distinct news stories will have no key terms (shot indices) in common. Therefore, the shot indices can be used to compare variations in news stories. Hence, a story boundary can be defined in a position where two consecutive shots have no key terms in common.

3.3.3 Variation of indices of shots

The variation of content in a local text block discussed in Section 3.2.1 of Chapter 1.2.1 was in the context of a particular shot. This section will discuss the content variation in a local text block between two consecutive shots. This refers to the variation of indices between two shots. There may be two cases of variation of indices between two consecutive shots:

Case i: When there is at least one common keyword (index) between two consecutive shots, it can be said that the two shots belong to the same news story. This case has been depicted in Fig. 3.4 where A, B, C ... are representative keywords.

Case ii: When there is no common keyword (index) between two consecutive shots, then it cannot be surely said whether they belong to the same news story. In this case, there arises a discrepancy regarding the origin of the news shot. So, the algorithm compares the key terms of that particular shot with the keywords of the next five consecutive shots. Suppose there lies at least one common keyword within the next five shots. Then, the shot up to which the common word has been found will be incorporated into the same news story. This scenario has been depicted in Fig. 3.5. If there lies no common keyword up to the fifth shot, then a story boundary is marked at the point of the first shot. This case has been shown in Fig. 3.6.

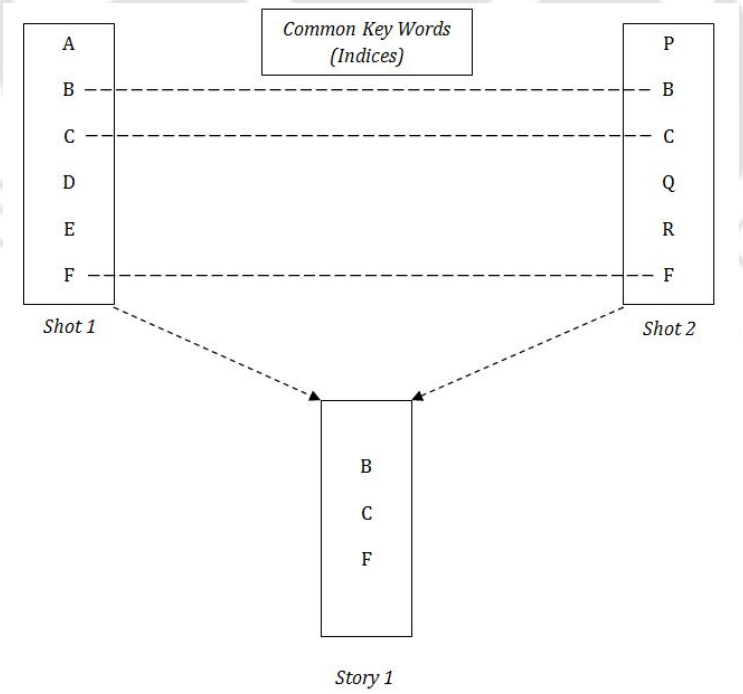


Fig. 3.4: When two consecutive shots have at least one common key word (index)

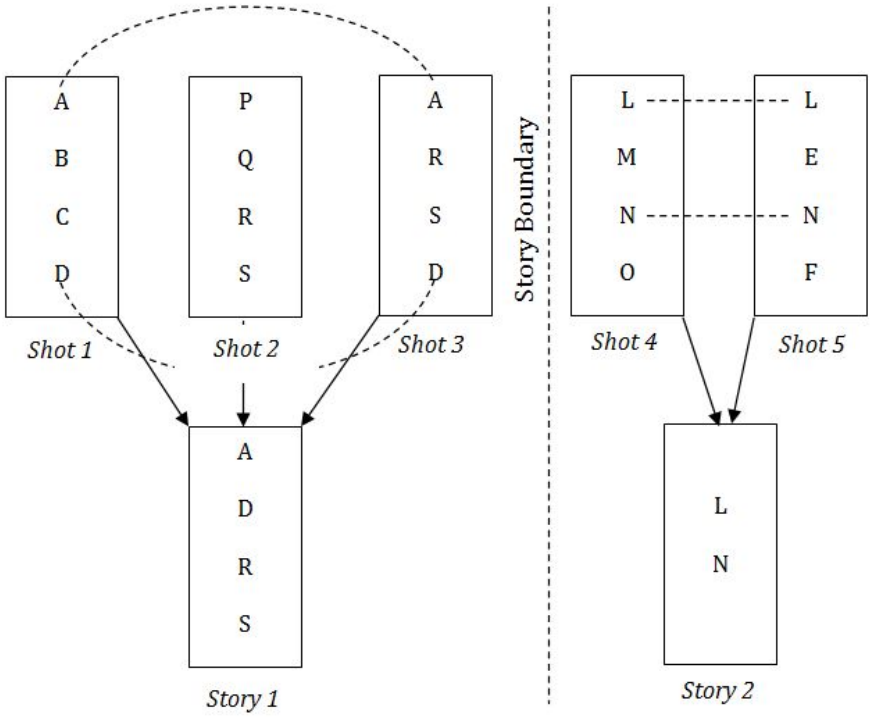


Fig. 3.5: When there is a common key word with other than the next consecutive shot

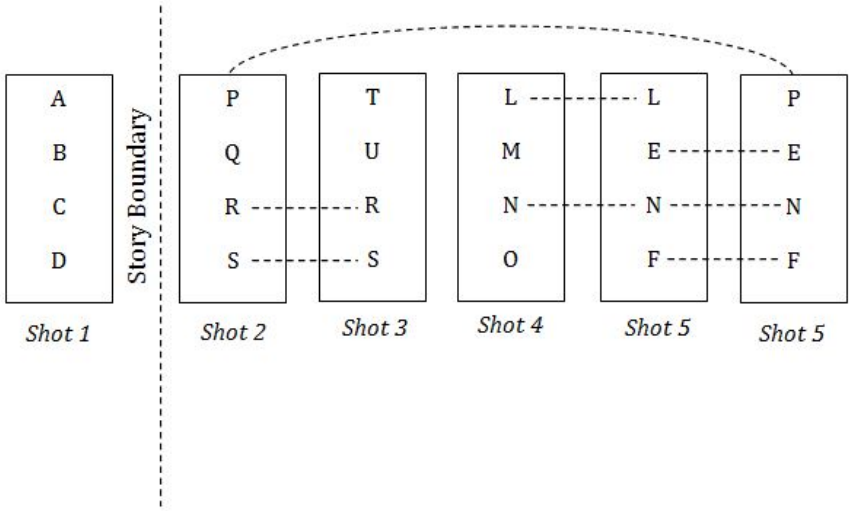


Fig. 3.6: When there is no common key word with next five consecutive shots

3.3.4 Formation of a news story

A news story is formed by combining shots with similar keywords. The similarity or, in other words, the variation between keywords is obtained as described in Section 3.3.3. A shot x is incorporated into a story X if it satisfies the following condition:

$$x \in X \iff A_x \cap A_n \neq \phi \text{ where } n = x+1, ..., x+5 \tag{3.3}$$

where,

A_x is the set of indices for shot x.

All shots x satisfying statement 3.3 will be incorporated into shot X. The point where Statement 3.3 will not be satisfied, a shot boundary will be marked in that position. After the combination of shots forms a story, that particular story will be tagged with the keywords that will satisfy Equation 3.2 where N is the number of shots in the story X.

3.4 Algorithm for Indexation of Shot based on Keywords

In the Indexing algorithm, we index news with words. The words will be based on sentences obtained from the local text block of each shot frame. But before proceeding to that, a discussion on the extraction of the largest text block and a threshold-based word-matching algorithm will be stressed. These two processes are necessary to obtain the indices of shots ultimately. These algorithms are found to be successfully working. The theoretical groundwork for this chapter has been discussed in Chapter 1.2.1. Hence, the content of this chapter has been laid upon more from the viewpoint of computational processing.

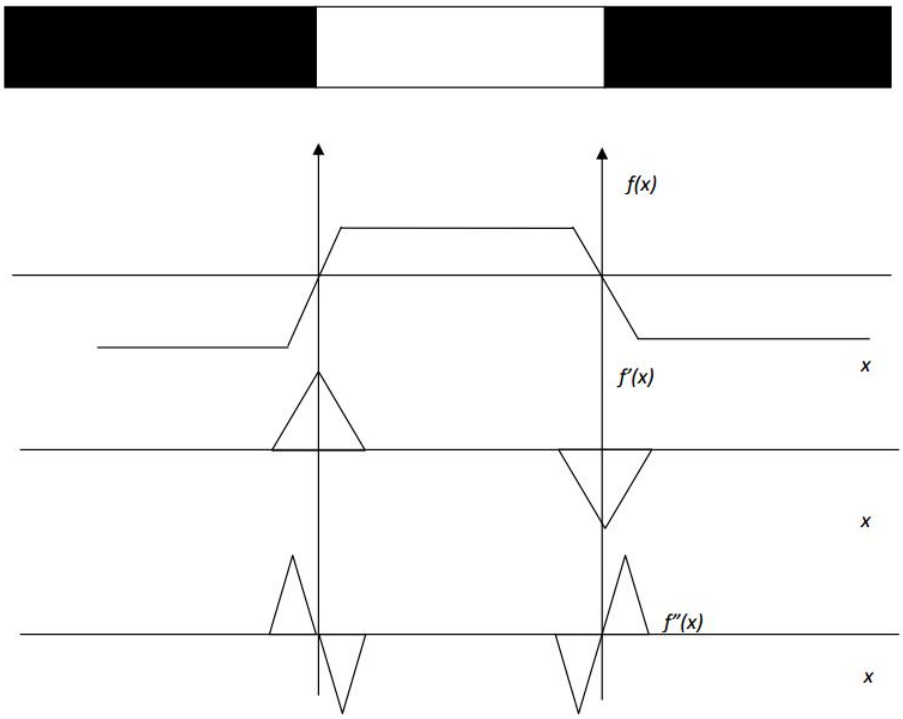


Fig. 3.7: Response of First order and second order differential operator on a black and white edge.

3.4.1 Edge detection of text blocks

The text blocks in a news video are brightly coloured in nature. This makes the text inscribed within it easily attractive to the viewers. Thus, the best possible way to extract the text blocks is to detect their edges. However, to detect an edge, it is necessary to define an edge. An edge can be defined as a change in the intensity level. It can be easily detected with the help of first and second-order differential (Laplacian) operators. The response of these operators changes upon the change in the colour level. Fig. 3.7 shows the first-order and second-order differential operator response on a black-and-white edge. Here $f(x)$ is a function of the intensity level at x.

The basic fundamental concept of edge detection is determining the change in colour level. This can be obtained by computing the first-order and second-order partial derivates. In this work, we have used first-order derivates for edge finding.

Z_1	Z_2	Z_3
Z_4	Z_5	Z_6
Z_7	Z_8	Z_9

Fig. 3.8: A 3×3 image with intensity level $z_i, i \in [1, 9]$

The first-order derivative is also called the gradient operator. The gradient operator in a discrete form about a point for a 3×3 region can be defined as:

$$g_x = \frac{\partial f}{\partial x} = (z_7 + 2 \times z_8 + z_9) - (z_1 + 2 \times z_2 + z_3) \tag{3.4}$$

$$g_y = \frac{\partial f}{\partial y} = (z_3 + 2 \times z_6 + z_9) - (z_1 + 2 \times z_4 + z_7) \tag{3.5}$$

where,

$z_i, i \in [1, 9]$ is the intensity level of a 3×3 region as shown in the Fig. 3.8

However, for finding edges in region of dimensions more than 3×3, equations 3.4 and 3.5 can be computed to form masks called Sobel Masks as shown in Fig. 3.9a and 3.9b. The Sobel mask is convolved with the destined image to compute the edges. In this work, we have used the Sobel Horizontal Mask for detection of horizontal edges.

-1	-2	-1
0	0	0
1	2	1

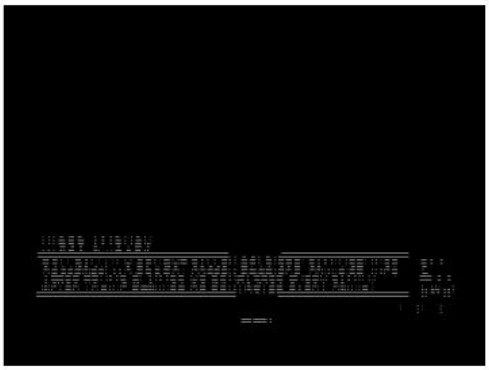
(a) Horizontal Sobel Mask

-1	0	1
-2	0	2
-1	0	1

(b) Vertical Sobel Mask

Fig. 3.9: Sobel Masks for edge detection

(a) Original Image (a news frame from CNN)



(b) Image after edge detection

Fig. 3.10: Sobel Masks for edge detection

After the execution of the Sobel Masks on an image (Fig. 3.10a) we obtain a binary image as shown in Fig. 3.10b where the '1's(white lines) represent the detected edges. After the detection of all the edges, the main challenge involves finding the largest horizontal edges. As a text block runs thoroughly across the screen, the largest edges represent the edges of a text block. This can be achieved through threshold comparison. A detailed discussion is provided in Section 3.4.7.

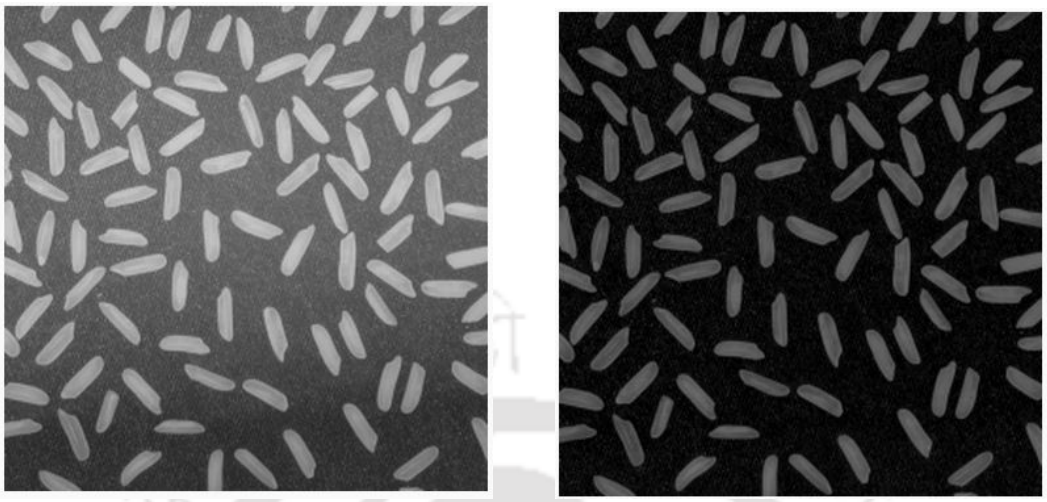
3.4.2 Identification of local text block

Section 3.4.1 discussed the theoretical background for identifying the largest horizontal edges in a frame. After detecting horizontal edges, the next step involves identifying text blocks. A text block is obtained when the vertical distance between two consecutive horizontal edges is within a specified threshold. If the vertical distance is too low (falsified double edges) or the vertical distance is too high(news display area), then that particular gap is neglected. The largest text block in either half of the screen will be a local text block.

After identification of the local text block, the portion of the frame that contains only the largest text block is retained. This portion of the frame is processed through Top Hat filtering to sharpen the edges of letters of the words in Local Text Block. The next step involves recognition of the text within the local text block. This can be achieved through the Optical Character Recognition feature of Matlab 2016a.

3.4.3 Top hat filtering

Top-hat transform is an operation that extracts small elements and details from given images. Top-hat filtering computes the morphological opening of the image and then subtracts the result from the original image. Opening of an image refers to erosion of the image followed by dilation. Dilation and erosion are morphological image processing techniques that are used to remove noises. Erosion erodes the edges of an image, and dilation widens the edges. For opening an image, a structural



(a) Before Top Hat Filtering *(b)* After Top Hat Filtering

Fig. 3.11: Top Hat Filtering

element needs to be defined. The structural element is a mask that is convolved with the original image. Based on the shape of the structural element, the results of the morphological operation are obtained. Top hat filtering, $T_w(f)$ is defined with the help of the following equation:

$$T_w(f) = f - f \circ b(x) \tag{3.6}$$

Where,

f is the original image and $f \circ b(x)$ refers to opening of the image f with structural element $b(x)$. Opening can be defined with the help of the following equation:

$$f \circ b(x) = (f \ominus b(x)) \oplus b(x) \tag{3.7}$$

Where,

$\ominus$ refers to erosion and $\oplus$ refers to dilation. An example of Top hat filtering is shown in Fig. 3.11a and Fig. 3.11b

3.4.4 Optical Character Recognition (OCR)

Optical Character Recognition (R Mithe, 2013) involves the process for detection of text in an image and recognizing its semantic meaning. In the case of OCR, the input is an image captured through a camera or a scanned document. The text in the document is converted to a machine-readable format. This is usually achieved with the help of an OCR engine. One of the most commonly used OCR engines is Tesseract. In an OCR engine, the image has to go through the following important steps:

- The image is converted into a binary image. The binary image is segmented to identify the individual characters in the image.
- The second step is recognition. It involves converting these images to character streams representing letters of recognized words.
- The final step is storing the recognized characters for semantic analysis.

3.4.5 Word confidence feature of OCR

Mostly, in the case of camera-captured images, a lot of noise is involved. In such a scenario, the OCR engine wrongly detects many words. However, while doing so, it provides a word confidence feature to the detected words. This feature has been used in our shot categorization algorithm to check the validity of the detected edge blocks. This feature has been used to analyze the surety of the detected words in our algorithm.

3.4.6 Threshold based word comparison

This section especially and independently focuses on a novel threshold-based word comparison algorithm. This algorithm is used both for the Indexing of Shots and the Formation of a Story by combining shots. As such, this algorithm deserves special attention and has been discussed independently. The main idea of the word

matching algorithm is to permit threshold-based comparison rather than obtaining a mere accurate result based on complete congruency and false result otherwise. The word matching algorithm will give a true value when there is 90% similarity and a false value otherwise. A threshold-based comparison is necessary because OCR may recognize the same word with a difference of one or two letters. Thus, this threshold-based comparison is necessary to compare similarity rather than complete congruency. Let us consider the comparison between two words, 'X' and 'Y'.

1. At the first step it will compute the length of each word. If the lengths are not equal then the output will be declared as false *i.e.* the words are not similar.

2. If the words are of equal length then the first letter of X will be compared with the first letter of Y, the second letter of X will be compared with the second letter of Y and so on. Each time the letters are same a count 'n' starting from zero is increased or if the letters are different the count 'n' remains same.

3. The two words 'X' and 'Y' will be declared similar if it satisfies the following condition

$$n \geq x\% \times N \tag{3.8}$$

where,

N is the length of the words and $x\%$ is the threshold for similarity.

This value has been set to 90% in both the algorithms.

3.4.7 Algorithm for indexation of shot based on keywords

The main objective of this algorithm is to identify a set of keywords for each shot. For this, it identifies the local text block in each frame. With the help of OCR it extracts the words which satisfy the threshold of word confidence. The extracted words are matched using a word-matching algorithm based on a user-set threshold. All those words obtained after the processes above are selected as keywords for the shot.

3.4.8 Steps for algorithm for indexation of shot based on keywords

This algorithm uses a range of features optimized to achieve better accuracy at minimal computational complicacies. The steps of the algorithm are at this moment discussed in detail:

1. The frames are extracted from the news shot in the first step.

2. Each frame is halved and only the upper half is considered i.e. if F is the original frame with dimension $M \times N$ then F_{new} is obtained from F by the following equation

$$F_{new}(x, y) = \{x | x \in [0, \frac{M}{2}] \text{ and } y | y \in [0, N]\} \tag{3.9}$$

3. The frames are convolved with a Sobel Horizontal Mask (Fig. 3.9a) to detect the presence of horizontal lines in the frame. The obtained values of g_x (Equation 3.4) are compared with a set threshold T_e as shown below

$$g_x \in L \iff g_x \geq T_e \tag{3.10}$$

where,

L is a set of horizontal lines. In other words L is a matrix given by the following equation

$$L(x, y) = \begin{cases} 1, \text{ if}(x, y) \text{ lies in a line.} \\ 0, \text{ if}(x, y) \text{ doesn't lie in a line.} \end{cases} \tag{3.11}$$

4. To mark a reference for comparison each halved frame (F_{new}) is marked with a perfectly horizontal line at the top and bottom of the frame.

5. In order to prevent the doubly detected lines or to convert slightly slanted lines to perfectly horizontal lines, the initially detected lines are averaged over

its neighborhood. This can be expressed mathematically as

$$L(i, y) = L(i - h, y) + L(i + h, y) \tag{3.12}$$

In our algorithm we have set the value of $h = 10$.

6. All detected lines are considered as edges if their length is greater than 50% of the total length of the frame *i.e.*

$$L(x, y) \in E \iff \sum_{x=1}^{N} L(x, y) \geq 0.5 \times N \tag{3.13}$$

where,

N is the width of the frame and E is set of valid edges *i.e,.* $E \subseteq L$

7. The space between two consecutive edges is considered to be a valid text block if the height of the text block is greater than 0.1% and less than 25% of the total height of the frame. To achieve this, it takes the first edge at the top as a reference. From the top edge, it goes on to check downwards until it finds a valid block height, which must lie between 0.1% to 25% of the total frame height. On reaching a valid edge, it repeats the same procedure, taking the last detected valid block end as a reference.

$$E_k(x, y) \in B \iff 0.01 \times \frac{M}{2} \leq \sum_{y=a}^{b} E(x, y) \leq 0.25 \times \frac{M}{2} \tag{3.14}$$

where,

$E_k(x, y)$ is the k^{th} iteration and a is the block end (edge) of $(k-1)^{th}$ iteration and b is the block end of the k^{th} iteration.

8. After detecting all valid text blocks, the largest text block is found out. This is done by calculating the height of each valid text block, and the text block having the largest height among them is retained. Thus, at the end of this step, the image is of size $T(h, N)$, where h is the height of the largest text

block.

9. Top Hat filtering operation is performed on $T(h, N)$. This is done to highlight the edge features of letters in the local text block. It helps the OCR function of Matlab to recognize letters more efficiently.

$$T_p(T(h, N)) = T(h, N) - T(h, N) \circ b(x) \tag{3.15}$$

Here,

$b(x)$ is a disk shaped structural element with a radius of 15 units.

10. The image T_p is further processed using the following equation:

$$I_{final}(h, N) = \begin{cases} 1, \text{ if } T_p(x, y) \geq T_h \\ 0, \text{otherwise} \end{cases} \tag{3.16}$$

where,

T_h is a threshold determined using Ostu's method

11. The final image I_{final} is given as an input to the OCR function in Matlab. The output comprises of a mixture of both wanted and unwanted words. The unwanted words are present due to misrecognition by the OCR function. These words have a very low value of word confidence. Thus to avoid these words we set a threshold value of word confidence at 0.90. All those words whose word confidence is below 0.90 are discarded and the rest are kept.

12. All the words remaining after Step 11 are converted to character. However after their conversion to character format the accessibility is increased to each letter of the word. This is very necessary for finding out the similarity between words.

13. All the unwanted spaces and special characters between words are removed. Most of the detected special characters and spaces are falsified classification

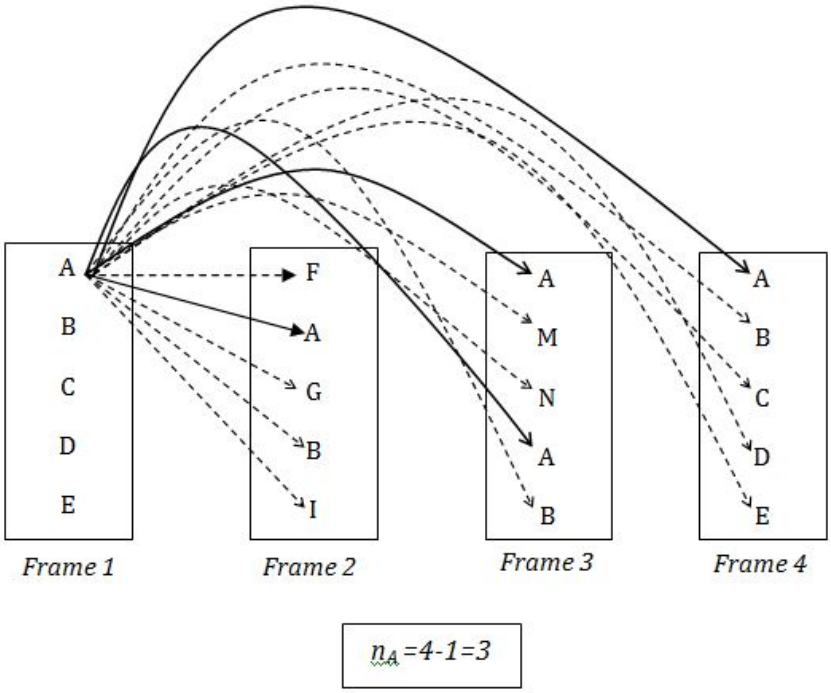


Fig. 3.12: Comparison pattern of a word between frames

and has to be manually trimmed to increase the accuracy in further processing.

14. Step 1 to Step 13 are repeated for all the frames of a shot.

15. The word matching algorithm is computed with all the other words of other frames except its own frame. The comparison pattern is shown in Fig. 3.12 where each letter is representative of a word. The number of repetitionS of each word is calculated using Equation 3.1

16. The number of words that have had a similarity match are removed to increase the computational efficiency.

17. Step 14 is repeated for all the words of all frames.

18. Finally the words of the shot which satisfy condition 3.2 are considered as indices of the shot.

19. Steps 1 to Step 18 are repeated for all shots of the news section.

	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1	'TAX'	'PLAYS'	'__'	'_'	'_'	'_'	'__'	'__'	'__'	'__'	'_'	'VICTIM?'	'_'	'_'
2	'PAYER,'	'VICTIM?'	'PAYER,'	'VICTIM?'	'TAX'	'TAX'	'TAX'	'TAX'	'_'	'LODTS'	'TAX'	'PLAYS'	'LODTS'	'TAX'
3	'PLAYS'	_'	'PLAYS'	'PLAYS'	'PLAYS'	'PAYER,'	'PLAYS'	'PLAYS'	'TAX'	'PAYEB,'	'PLAYS'	'TAX'	'TAX'	'PLAYS'
4	'VICTIM?'	'TAX'	'VICTIM?'	','	'VICTIM?'	'PLAYS'	'VICTIM?'	'VICTIM?'	'PLAYS'	'PLAYS'	'VICTIM?'	'lOOTS'	'PLAYS'	'VICTIM?'
5	'_'	'.'	'LODTS'	'LODTS'	'LODTS'	'VICTIM?'	'.'	'LODTS'	'VICTIM?'	'VICTIM?'	'LODTS'	[]	'VICTIM?'	'.'
6	'LODTS'	[]	'TAX'	'TAX'	'PAYER,'	'LODTS'	'_'	'.'	'.'	'_'	'.'	[]	'PAYER,'	'PAYER,'
7	'LOOTS'	[]	'_'	[]	'.'	'.'	'PAYER,'	'_'	'PAYER,'	'TAX'	[]	[]	[]	'LODTS'
8	[]	[]	'PAYEB,'	[]	[]	[]	'LODTS'	[]	'LODTS'	'.'	[]	[]	[]	[]

Fig. 3.13: Sreenshot of results for Indexation of Shots based on Keywords

After the execution of the aforementioned steps on the shots of a News Section, each shot will be indexed with its key terms. Fig. 3.13 shows a screenshot of the results, where each column represents a shot. Thus, the result will be a structure $S(p, q)$ where each cell will be a word and each column represents a shot. This structure will be the direct input for the Algorithm for Story Formation by Shot Combination.

3.5 Algorithm for Story Formation by Shot Combinations

The main objective of the story formation algorithm is to combine similar shots to form individual News Stories. The similarity between shots will be measured based on the keywords with which the shots have been indexed using the algorithm discussed in Section 3.4.7 of Chapter 3.4. The Algorithm for Shot Formation is based on the theoretical aspects discussed in Chapter 3.3. The Keyword Matching Algorithm (Section 3.4.6) is also used here to find the similarity between indices of shots. While designing this algorithm, the main focus has been kept on minimizing the computational time so that it could be later used for Real-Time Processing. The output of this algorithm will be individual stories; each indexed with a set of keywords that will reflect the news story's content.

3.5.1 Algorithm for shot combination

The algorithm's output for the indexation of shots based on Keywords, which is a structure $S(p, q)$, can be directly fed as input to the Algorithm for Story Formation using Shot combination without any preprocessing. Each keyword of a shot will be compared in a very particular pattern, which has been optimized to minimize computational processing time. The comparison is done using the Key Word Matching Algorithm. Shots having similar news content will be combined to form a news story. The news story will be again tagged using keywords.

Steps of the shot combination algorithm

1. $S(1, 1)$ which is the first keyword of the first shot will be converted to an array and will be considered as the base word. The base word will be compared with $S(x, 2)$ where $x \in [1, p]$ using the Keyword Matching Algorithm (Section3.4.6)

2. In case a similarity is found in the second column (i.e. second shot) then the second shot will be incorporated into the first shot.

3. If no similarity is found for $S(1, 1)$ then the new base word will be $S(2, 1)$ and steps 1 and 2 will be repeated for $S(2, 1)$. If no similarity is found then the new base word will be $S(3, 1)$ and so on it will continue until the last word $S(p, 1)$ is reached.

4. If a similarity is found Step 1 to Step 3 will be again repeated but this time it begins from $S(1, 2)$ and compared with $S(x, 3)$ where $x \in [1, p]$. And if a similarity is found the third shot will be sequentially incorporated into the second and first shot. This can be further generalized as

5. Step 4 holds valid if a similarity exists in the immediate next column. If no similarity is found then i of Algorithm 2 is varied upto $i + 5$. Algorithm 2 can be further modified as shown below: From the algorithm it is clear that if no similar key word lie in the next immediate shot then algorithm will continue

Algorithm 2

while $i \leq q$ **do**

if $S(n,i) \cap S(x,i+1) \neq \phi$ for $x \leftarrow [0,p]$ **then**

$i \leftarrow i+1$

else $n \leftarrow n+1$

Algorithm 3

Initialize $5k = 0$

while $i \leq q$ **do**

while $S(n,i) \cap S(x,i+1) = \phi$ for $x \leftarrow [0,p]$ and $n \leftarrow [0,p]$ **do**

$k \leftarrow k+1$

$i \leftarrow i+k$

if $S(n,i) \cap S(x,i+1) \neq \phi$ **then**

Add i^{th} shot to l^{th}story

$k \leftarrow 0$

Break

if $k > 5$ **then**

End story at i^{th} shot and begin $(l+1)^{th}$ story from $(i+1)^{th}$ shot

$k \leftarrow 0$

Break

the comparison upto the next five shots. If there lies no similarity upto the next five shot then a story boundary is located in that particular shot. If a similar keyword lies in the third or fourth shot then story will continue upto the third shot. After this comparison will continue from the fourth shot onwards.

6. Once a similar keyword of a particular shot is located it is removed so that repetitive comparison doesn't occur. Besides this once a similar keyword is located in a shot no other keyword of that shot will be compared.

7. When Step 1 to Step 6 is repeated for the entire range of shots then the output will be a set of stories. In actual practice the output is a set of videos, each video representing a news story

8. After a news story is obtained the key words of the shots from which it is obtained are fed as input to the Shot indexation Algorithm. The input in this case is key words of individual shots rather than words from local text blocks of news frames. The output will be the most repeated key-words from the set of inputs. Those key words are used to tag the story. Thus each individual story will be tagged with a set of key words that reflects the content of the story.

3.6 Results and Analysis

The Shot Indexation Algorithm was tried on the **721 news shots** obtained from the Shot Categorization Algorithm (Section 2.4.2). Each shot was successfully indexed with the keywords. Fig. 3.13 shows a screenshot of the results. The indexed shots were recombined to form **8 stories** using the Algorithm for Story formation by Shot Combination. The database was obtained from 1 hour of 7 pm Prime Time News from Times Now. The algorithms were coded on the Matlab 2016*a* platform. The algorithms acted by expectations. However, certain errors can be justified as

follows:

One of the main errors of the Shot Indexation algorithm was the misrecognition of words from the local text block of a news frame. This indicated further preprocessing was required, as well as improvement of the OCR feature of Matlab. However, the algorithm made provisions to remove mis-detected characters, especially single symbols, and unwanted spaces. Besides this, the local text block of certain channels features logos and thumbnail pictures that doped the results obtained from the OCR function. Thus, one of the significant improvements of this algorithm lies in introducing preprocessing techniques that help the words in local text blocks to be recognized more easily.

The errors of the shot indexation algorithm get forwarded to this algorithm. This results in over-classification or under-classification of news shots. This usually can be corrected by varying the comparison parameter to more or less than five consecutive shots. Another source of error lies in the misdetection of the local text block of the Shot indexation algorithm. A misidentified local text block will contribute to the wrong set of keywords, which will cause a hindrance to this algorithm. The scope of improvement of this algorithm lies in developing an automatic threshold based on the keywords obtained.

3.7 Conclusions

This chapter primarily focuses on the spatial description of a news frame in a news video. It also provides a detailed discussion of how the spatial and temporal structures have been exploited to define a feature. The extracted text block was processed using Optical Character Recognition to identify the text within the block. The word identified for each frame has been used to tag the corresponding shot with a set threshold based on how many times the words are repeated. For shots to be combined into the story, they should have at least one keyword in common for every five shots. This threshold was determined based on practical observation. However,

it can be adjusted based on the user's specific needs. Finally, we developed an algorithm for indexing a shot with its key terms. The key terms were obtained from the word-matching algorithm, and another algorithm was developed to form the story by combining news shots. The main focus of this algorithm was to reduce computational processing time while keeping the desired objectives in mind. The output of this algorithm will be a set of news stories that have been combined from shots. Each story will be tagged with a set of keywords that describe the content of the news story. This algorithm was tested on Matlab 2016a and was found to work successfully.

4

Story Segmentation and Indexing of Broadcast News Videos in a Multi-modal Framework

Contents

4.1 Introduction

We proposed a novel technique for story segmentation of news videos. The visual similarity, silence in the audio, and the text in text boxes of a news video are used as parameters to create an index, and these three indices are fed to a probabilistic multimodel algorithm, which then predicts the story breaks. The multimodal

algorithm takes account of the previous state of the indices and predicts the present state. Then, It is compared with the actual present indices, and the story breaks are determined. The segmented stories are then indexed for retrieval of the stories.

4.2 Algorithm for Story Segmentation

Story segmentation is the process of segmenting news videos in accordance with the story in the video. The input video is pre-processed to remove commercials, and three features are extracted from the video: visuals, text, and audio. These features give substantial information about the present frame and its relation with a few previous frames. We have exploited these features to determine the story breaks in a news video. These features and the method of determination are discussed in the following subsections.

4.2.1 Visual index

The visual index refers to how much a frame resembles the previous one based on the visual content. A shot is a segment of video collected between consecutive on and off of the camera. During one shot, the visual content remains roughly the same or changes linearly. However, an abrupt change is seen between two consecutive frames during a shot transition. These transitions are only applied to the body of the news rather than the whole frame. Thus, each frame is pre-processed to remove the text boxes and filter out the body of the news frame. As the text boxes are rectangular, the rectangles at the top and bottom of the frame having a width equal to the input frame are considered text boxes. These text boxes are then cropped from the input frame.

In a news video, split screens are used to pass as much information as possible in less time. In a split-screen, the frame is divided into multiple parts so that the field and anchorperson shots can be displayed together. The field shot generally occupies more space in the frame than the anchorperson shot. When a

split-screen is used, multiple shots of the field shot may be displayed in one part of the frame while the anchorperson shot remains the same. These shots come under the same story. Due to such changes in the significant part of the frame, the value of the index will be high, which may indicate an undesirable story change. To overcome this, the boxes in which a shot is displayed are detected. In this case, the rectangles with a height almost equal to the height of the frame after removal of the text boxes are considered as split-screen boxes. If the number of rectangles detected for split-screen is two, the frame is divided at the rectangle boundary. Otherwise, the frame is divided into two equal parts for ease of computation.

To detect transitions, the pixels of each half are compared to those of the corresponding previous half. The index's value is determined based on the dissimilarity between the pixels. To compare frames, one of the requirements is to convert the three-dimensional frames into two-dimensional ones. For this, the values of the three channels of a pixel must be represented by a single value. This is achieved using (4.1).

$$pixel = \sum_{n=1}^{3} \frac{value\ of\ the\ i^{th}\ channel}{10^i} \tag{4.1}$$

These values are stored in another 2-dimensional matrix and histogram of each half of a frame is obtained. The histogram matrices represent the frame as a point in an n-dimensional coordinate system, where n is the size of the matrix. The difference in the angle made by the position vectors is calculated using (4.2).

$$Angle = \cos^{-1} \frac{im1.im2}{|im1||im2|} \tag{4.2}$$

Where,

$im1$ = the histogram matrix of the 1^{st} frame

$im2$ = the histogram matrix of the 2^{nd} frame

This angle is calculated separately for each half of the frame. The maximum

possible angle between two position vectors is 180 degrees. So the visual index is obtained using (4.3).

$$index_{visual} = \frac{min(angle_1, angle_2)}{180} \tag{4.3}$$

Where,

$angle_1$ = Angle made by the first half of the frame with respective previous half.

$angle_2$ = Angle made by the second half of the frame with respective previous half.

4.2.2 Audio index

The audio feature of the video is also used as an index for story boundary detection. Whenever a story changes, the duration of silence is more than that in continuous news reading. The algorithm calculates the duration of silence whenever there is a silent snippet. At first, the audio framing is done so that frames per second of the audio and video are the same. The sample rate of the audio gives the number of samples per second. So, the number of samples in an audio frame is calculated using (4.4).

$$Number\ of\ samples = \frac{Sample\ rate}{Frame\ rate\ of\ the\ video} \tag{4.4}$$

A silence frame is detected using Short-Term Energy(STE) of that frame [59]. Short term energy of a signal in a frame is calculated as (4.5).

$$STE = \sum_{n=0}^{N-1} x(n)^2 \tag{4.5}$$

Where,

N = Number of samples.

When the short-term energy of a frame is less than a threshold, it is considered as a silent frame. The time span of the period can be obtained by finding the number of such continuous frames. Since the frames per second of the audio and

the video is same, the duration of silence in seconds is given by (5.1).

$$duration = \frac{Number\ of\ silent\ frames}{Frame\ rate\ of\ the\ video} \tag{4.6}$$

We have considered maximum duration of silence in a news video as 4 seconds *i.e,*. 100 frames for 25 fps video. So the audio index for all the frames that comes under a duration is given by (4.7).

$$index_{audio} = \frac{Number\ of\ continuous\ silent\ frames}{100} \tag{4.7}$$

If the value of index exceeds 1, it is reassigned as 1.

4.2.3 Text index

The text boxes in a news video can be classified into local and global text boxes. The local text boxes show news related to the news being telecast on the screen, whereas the news in global text boxes may or may not be related to the current news display. The first procedure to extract meaningful information from the local text boxes (to segment the news into stories) is the identification of the local text boxes. The local text boxes are generally placed near the news screen area and are constantly changing. To extract as much information as possible, we cropped out the local text boxes and filtered them so that the boundaries of the letters in the text box became sharp and easily distinguishable.

After the pre-processing, Optical Character Recognition (OCR) is applied to the processed image to extract as much data as possible. The OCR algorithm could be better and miss out on a few data or misinterpret letters as symbols, so there are instances when a complete word is not formed, which induces error while calculating the index. To overcome this issue, we have tried to check each detected letter with all 26 alphabets. A 1×26 row matrix is formed for each of the frames in the video such that the 1st column contains the number A in the local text box, the

2nd column contains the number B, the 3rd column contains the number C, and so on.

The similarity of text boxes in subsequent frames can be determined by the similarity of the 1×26 column matrix of subsequent frames. A text similarity index is created by counting the number of similar elements of the two subsequent frames in 1×26 matrix. The maximum of the similarity index is 26 when the numbers of letters are the same in the previous frame, and the minimum is 0 when the count of each letter is unique to the previous frame. So, this similarity index gives us the frames at which the letters in the text box drastically change beyond a tolerance value, and we can say that the text in the text box has changed.

4.2.4 Multi-modal approach for story segmentation

The three indices are combined using a multimodal algorithm. This algorithm builds a trust factor depending on the variance of each factor from its previous values. The importance given to a particular index depends on the trust factor associated with it. From the third frame onwards, the present frame is estimated (predicted) based on the previous two frames. The predicted frame represents the linear variation of indices. The difference between the predicted frame and the actual frame (or the error) forms the basis of the trust factor, and it is significant when there is an abrupt change in the index. Combined with the abrupt change and the percentage of the trust factor, the algorithm segments the video into stories if the combined threshold of indices is above the normal value or if there is a sudden spike in the value of the coefficient. The predicted index of the current frame is given by (4.8).

$$V_{ip} = V_{i-1} + (V_{i-1} - V_{i-2}) \tag{4.8}$$

where,

V_{ip} = predicted index of present frame

V_i = actual index of present frame

This predicted value is calculated for each one of the indices and the error between the predicted and the actual value of indices is calculated using (4.9).

$$V_{err} = abs(V_i - V_{ip}) \tag{4.9}$$

where,

V_{err} = error of the predicted and actual value of indices

$$Den = V_{err} + A_{err} + T_{err} \tag{4.10}$$

$$V_{coeff}(i) = \frac{V_{err}}{Den} \tag{4.11}$$

$$A_{coeff}(i) = \frac{A_{err}}{Den} \tag{4.12}$$

$$T_{coeff}(i) = \frac{T_{err}}{Den} \tag{4.13}$$

where,

V_{err} = error of the visual index predicted and the actual value.

B_{err} = error of the audio index predicted and the actual value.

C_{err} = error of the text index predicted and the actual value.

The error is calculated for each of the three indices and the trust factor(coefficient) for each of the indices is formed as (4.10).

The combined coefficent using multimodal algorithm is calculated as (4.14).

$$combi_i = (V_{coeff} \times (max(V_i, V_{i-1}))) + (A_{coeff} \times (max(A_i, A_{i-1}))) + (T_{coeff} \times (max(T_i, T_{i-1}))) \tag{4.14}$$

Where,

A_{coeff} = trust factor for audio

V_{coeff} = trust factor for video

T_{coeff} = trust factor for text

$combi_{coeff}$ = combined multimodal coefficient

These trust factors are then multiplied with their respective maximum of present and last index to form a general coefficient (or multi-modal coefficient) of the story for each frame. The video can be segmented on a particular frame based on the coefficient of each story frame. So, the trust factor allows us to manipulate the importance of each of the indices. The trust factor assigns greater importance to a changing value, ensuring that story segmentation depends more on a significantly changing value rather than a minor change in any one index.

4.3 Indexing of Segmented Stories

The story segmentation algorithm successfully segments the stories of the news video. However, these stories need to be indexed or labeled correctly for ease of retrieval. Indexing refers to assigning keywords to the segmented stories to make the searching process more convenient.

Keywords are extracted from the local text boxes of the frames in a story. The words from the text boxes are obtained using Optical Character Recognition (OCR). OCR also provides word confidence for each detected word. Word confidence refers to the extent to which the OCR algorithm is confident about a detected word. From our observation, if the word confidence is greater than 85%, it can be assumed that the word is correctly detected. But even for some correctly detected words, the word confidence lies in the 75-85% range along with other incorrectly detected words. Thus, the results obtained from OCR are only partially reliable.

To avoid this discrepancy, we checked the spelling of each detected word. The spell check algorithm opens MS Word for each word and uses its built-in spell checker. If the word is present in the database of MS Word, the algorithm returns the result to be correct and otherwise incorrect [60]. However, opening another application for each of the detected words is time-consuming. A text box will appear on the screen for at least 1 a second. So, instead of considering all the story frames, the frames with an interval of 25 frames are considered. The words for which the

word confidence is above 85% are considered correctly detected words, and spell check is applied only for those words for which the word confidence lies in the $75-85\%$ to reduce the time further. In this way, the incorrectly detected words of OCR are discarded.

Thus, raw keywords data is obtained, including punctuation and words like 'a', 'are', 'above,' etc. These punctuation are to be removed to reduce the redundancies in detected keywords. Text Analytics Toolbox [61] in Matlab contains a database of all punctuation and 225 stop words. The words included in that database are removed from the raw data, and a clean data of words is obtained. Now, a cell array of two columns is used where each word is stored only once, along with its count. From that array, the percentage frequency of each word is calculated using (5).

$$\%frequency = \frac{count\ of\ a\ word}{\sum_{i=1}^{N} count\ of\ ith\ word} \times 100\% \tag{4.14}$$

where,

N= Number of words.

The words for which the percentage frequency is above 60%, are assigned as keywords to the story. Thus, the stories are indexed with appropriate keywords.

4.4 Results

The algorithm is used in a two-hour news video from the 'Times Now' channel. The video has 25 frames per second and 180000 frames. The video is directly recorded using the iball claro TV T18 TV tuner card [62]. The software used for image processing is MATLAB along with its Image Processing Toolbox [63], Text Analytics Toolbox [61], and Parallel Computing Toolbox [64]. The frames of the video are extracted from the video, and each frame is treated as an image and analyzed in the image processing engine. After removing commercials, the three indices are calculated parallelly using the Parallel Computing Toolbox. The three

Algorithm 4 Algorithm for indexing of segmented stories

```
Result: A story indexed with keywords
    N← Number of frames in the story
    i, k, a ← 1
    raw_data← Empty array to store words
    while i ≤ N do
        if mod(i, 25)==0 then
            Extract the frame
            Detect the local text box
            Apply OCR in the text box
            M← Number of words detected
            j← 1
            while j ≤ M do
                Find the word confidence of the word
                if word_confidence≥ 0.85 then
                    raw_data(k) ← word
                    k← k+1
                else
                    if 0.75≤ word_confidence ≤ 0.85 then
                        Check the spelling of the word
                        if the spelling is correct then
                            raw_data(k)←word
                            k← k+1
                            end
                        end
                    end
                    j← j+1
                end
        i← i+1
        end
    L← Number of words in raw_data
    while a ≤ L do
        if raw_data(a) is a punctuation word then
            Discard that word from the array
        else
            if raw_data(a) is a stop word then
                Discard that word from the array
                end
            end
        a← a+1
        end
    Count the number of each word remaining in raw_data
    Find the % frequency of each word using (5)
    if % frequency≥ 60 then
        That word is assigned as keyword
        end
```

indices are then fed to the multi-modal system that finally segments the video. Precision, recall, and F-measure parameters are used to estimate the performance. Precision(P) is identified as the ratio of the number of events identified correctly by the algorithm to the total number of events identified. Recall(R) is the ratio of total number of events identified correctly to the total number of actual events. The F-measure is calculated as the weighted harmonic mean of precision and recall of the test. It can be determined using (4.4).

$$F - measure = \frac{2PR}{(P+R)} \tag{4.14}$$

4.4.1 Filtering of commercial

To calculate the results for the commercial filtering algorithm, it is assumed that a commercial starts at the first disclaimer or animation frame and ends at the first anchorperson frame. The results are presented in Table 2.5 in Chapter 2. It can be observed that some commercial frames are even misdetected as news frames, leading to an overestimation of the total detected news frames compared to the actual ones, as determined using (4.4).

4.4.2 Story segmentation

Fig.4.2 represents the actual variation of stories, and Fig.4.1 represents the multimodal coefficient for frames. The actual variation of stories is calculated by assigning constant trust factors obtained from manual supervision. It can be seen that the unsupervised multi-modal algorithm determines the story boundaries quite accurately. Though a small quantity of noise creeps in, the spikes in the graph that represent the story change remain the same. The analysis of the results is shown in Table 4.1.

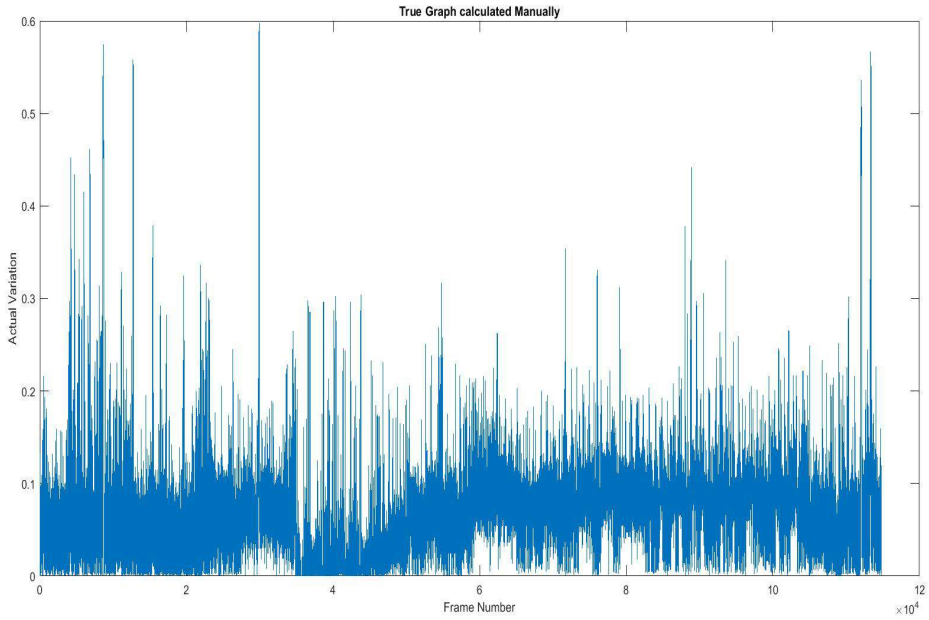


Fig. 4.1: Actual variation of stories

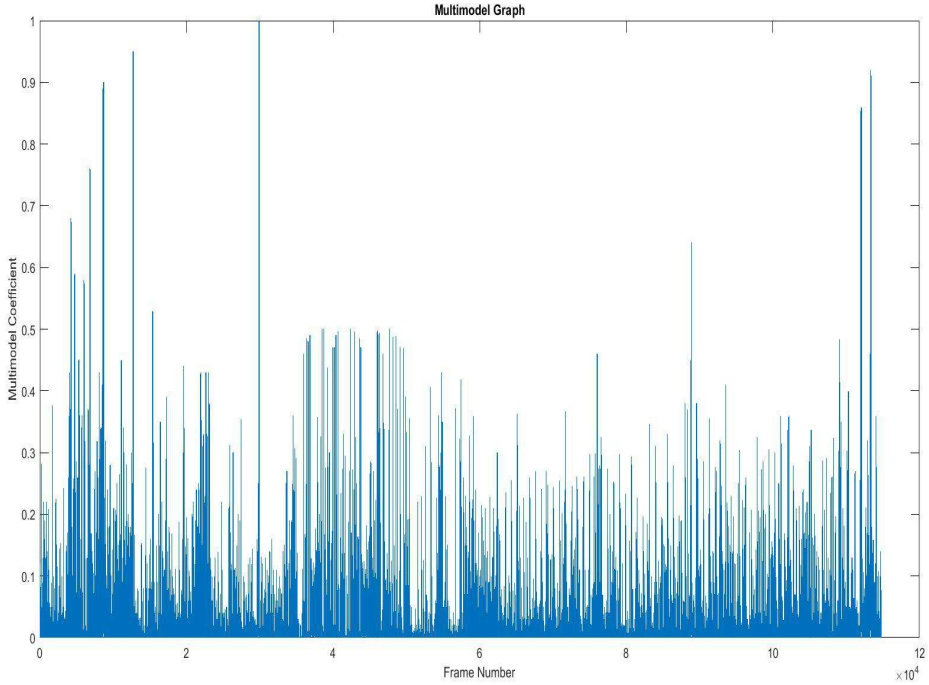


Fig. 4.2: Multimodal coefficient of frames

Tab. 4.1: Result analysis for story segmentation

Parameter	**Value**
Total number of actual stories	6
Total number of detected stories	5
Total number stories identified correctly	5
Recall	83.33%
Precision	100%
F-Measure	**90.91%**

4.4.3 Indexing of segmented stories

The algorithm is applied in four segmented stories. After watching the story, some keywords are predicted that a user may use to search for that particular story. Table 4.2 shows the predicted and detected keywords. The result analysis for indexing is shown in Table 4.3.

A comparison of results obtained by different authors on story segmentation is given in Table 4.4.

4.5 Conclusions

The proposed system can filter out the stories from a news video in an effective manner, and the input video is segmented into different stories. The multi-modal approach makes the system more sensitive to each parameter's change. The weightage given to all the parameters is equal when there is no change in the system output. The stories segmented are indexed based on the news displayed and the story linked with those videos. Any constituent news can be found in the output folder tagged with the indexing keywords. The algorithm also outputs commercials as a different concatenated video. This can be very helpful for the advertising industry to analyze the advertisements displayed on a particular channel and the frequency of each

Tab. 4.2: Analysis of predicted and detected keywords

Topic	**Predicted keywords**	**Detected keywords**
Discussion on change of the color of the jersey of Indian cricket team from blue to orange.	Orange, Color, Jersey, Cricket	Orange, Courage, Cong, Jersey, Saffron, Suspect, Object
Discussion on the violence of an MLA, son of a highly powerful politician against a government official at a public place.	BJP, Son, Father, Defend, MLA, Power, Bureaucrat, Violence, Politician	BJP, Son, Father, Defend, Hurl, Media, Aukaat, Naamdar, Culture, Drunk, Power, Bash, Bureaucrat, Bat
Discussion on suspecting the Chief Minister of Haryana for trying to help a rapist.	Land, Rapist, CM, Khattar, Parole, Haryana	Land, Rapist, Reveal, Parole, Justification, Colluding, Help, Investigation, CM, Khattar, Expose
Discussion on the speech of the Prime Minister at the parliament on the vision of new India.	Vision, India, Modi, PM, Congress, Parliament, Development	India, Modi, Development, Congress, Counter, Taunt, PM, Vision, Underline, Ghotala

Tab. 4.3: Result analysis for indexing of segmented stories

Parameter	**Value**
Total number of Predicted keywords	26
Total number of predicted keywords that are actually identified	20
Recall	76.92%

Tab. 4.4: Comparison of Results for Story Segmentation

Paper	**Recall(%)**	**Precision(%)**
Song *et al.*, [40]	94.1	82.1
Wang *et al.*, [45]	93	89
Dumont *et al.*, [47]	89.3	76.7
Chaisorn *et al.*, [33]	74.9	80.2
Chaisorn *et al.*, [49]	74.9	80.2
Feng *et al.*, [50]	95.3	97.3
O'Connor *et al.*, [51]	84	84
Zhai *et al.*, [54]	64.21	73.81
Tapu *et al.*, [55]	90	90
Zedan *et al.*, [56]	95.96	94.06
Our Approach	83.33	100

advertisement. The system yields a satisfactory F measure of 90.91%

The algorithm is computation-heavy and requires suitable hardware to process it. We plan to use speech recognition techniques to extract more keywords for indexing the segmented stories. We will also analyze and work on detected commercials to find the frequency of each commercial and make a comprehensive analysis that may be helpful for the advertisement firms.

5

Image and Video Clip Searching and Retrieval in Broadcast News Videos.

Contents

5.1 Introduction

The study of image-based retrieval becomes challenging when the database consists of videos. This variation of visual search is important for a broad range of applications that require indexing video databases based on their visual contents. The main agenda of this work is to retrieve the data of an image or a particular shot/story of a video, compare it with a part of broadcast news, and check its occurrence in the broadcast news video. Outnumbering different algorithms, we have devised an efficient way of retrieving the data. In this work, we extracted the frames for a particular video segment and performed image/video-based searching. In image-based searching, the query image is converted into a gray-scale image, and its pixel values are mapped and matched with those in the broadcast news database. The video frames are converted into binary form in video-based searching, and an edge detection algorithm is applied. The similarity between the detected edge is calculated for the data extracted from the query video shot and broadcast news video. High-dimensional indexing has recently received extensive study and remains an open research area. The multi-frame video representation further increases the problem's complexity. The second part proposes a novel scheme for video searching and retrieval using scene classification.

5.2 Pixel Mapping-Based Image Searching and Retrieval in Broadcast News Videos

In this algorithm, the data is extracted from the image to be searched in the broadcast news database, and a list of operations is performed on the image and the broadcast news database. The search results are being obtained based on the results

of all the operations. The list of operations that are performed for image-based video searching are listed below.

A. Removal of text boxes of image/frame.

B. Conversion of image/frame in RGB to Grayscale.

C. Mapping of pixel values of the grayscale image.

D. Matching of the image matrix after mapping.

E. Identification of the image based on percentage match.

5.2.1 Removal of text boxes of image/frame

The text box keeps scrolling in a news video, so at the same images, different text boxes may occur, so it becomes essential to remove the text boxes to check the similarity between the frames.

5.2.2 Conversion of image/frame in RGB to grayscale

The RGB color model, where the image consists of three matrices, makes it very difficult to perform computations. Hence, the image matrix is converted into a single matrix by converting the image from an RGB colour image to a gray-scale image, resulting in reduced computation in the image being processed.

5.2.3 Mapping of pixel values of the grayscale image

Gray scale pixel values of an image lie within the range of intensity values from 0 to 255. First, all the pixel values(*i.e.,* $0 - 255$) are grouped into five groups, dividing the pixel values ranging from 0 to 51, 52 to 102, 103 to 153, 153 to 204, and 204 to 255, as shown in the Fig. 5.1. For a particular image, it is checked for all the pixel values in the image, in which of the five groups the pixel value lies. After checking on the specific group in which the pixel value lies, the pixel value of the

image under process is mapped to the smallest value in the group. This is done for all the pixels in the image.

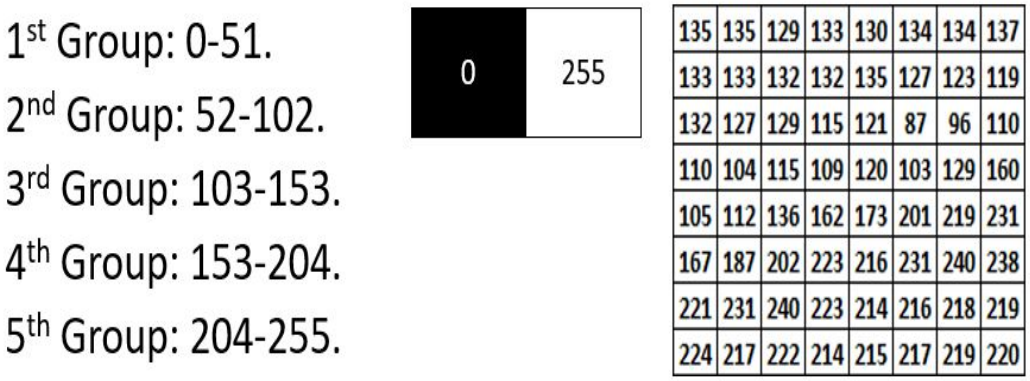


Fig. 5.1: Mapping of pixel values of the grayscale image

5.2.4 Matching of the image matrix after mapping

The processed query image matrix, i.e., the matrix with the mapped pixel values of the image being searched, is matched for similarity with the images in the database of the broadcast news video. The percentage of similarity between them is then calculated. To check the similarity between the query image and the images in the broadcast news, the similarity in the elements of the matrix is verified. The percentage of the match is calculated by the following formula.

$$Percentage\ of\ Match = \frac{Number\ of\ similar\ pixels}{Total\ Number\ of\ pixels} \times 100 \tag{5.1}$$

5.2.5 Identification of the image based on percentage match

After calculating the percentage of the match for the query image being matched with the broadcast news database, a particular threshold is set to find the images with maximum similarity, a Plot of the frame no vs. percentage of match gas is shown in Fig. 5.2. For this, a threshold value of 97 % is set so that images greater than 97 % similarity gets selected.

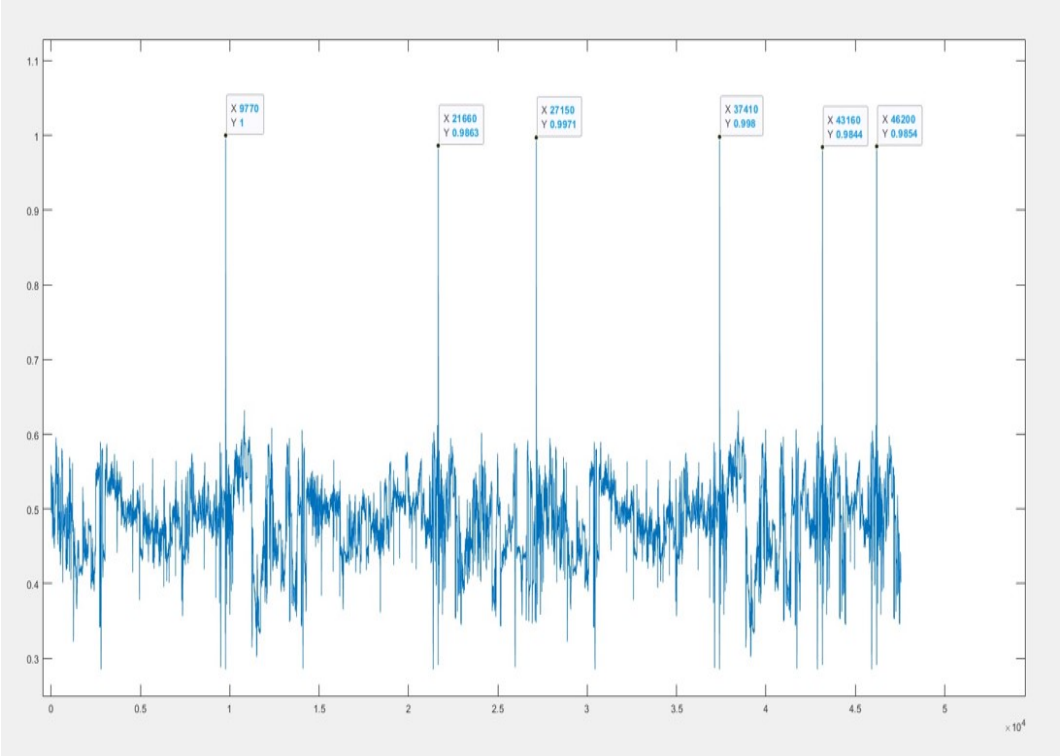


Fig. 5.2: Plot of frame no vs percentage of match

5.3 Edge-Based Video Clip Searching and Retrieval in Broadcast News Videos

In the Edge-based video clip searching and retrieval technique, we first extracted the frames for a particular segment of video and converted each frame to binary form, detected the edges of each frame using the edge detection method, and stored the no of edges detected into arrays for the query video shot as well as the broadcast news. The no of edges detected may have some discrepancies despite the similar frames. To overcome this discrepancy, we have developed an algorithm for checking the similarity between the number of detected edges of the query video shot and the broadcast news. The algorithm we devised and tested in the video clip based technique has been divided into sub-sections.

5.3.1 Detection of edges of the image/frames

To detect edges within frames/images using the RGB color model, the initial step involves converting them into binary form to facilitate the implementation of the edge detection algorithm. In our scenario, our focus has been on segments of broadcast news and video shots. The primary challenge revolves around accomplishing the conversion process and subsequently applying the edge detection algorithm.

Edge detection is an image processing technique for finding the boundaries of an object in a given image. Every image/frame has its unique boundaries and values through which we can identify the image. Roberts, Prewitt, Sobel, and Canny are the methods through which edge detection can be performed. So, in our case, we have applied all four methods above separately for the broadcast news and video shot. Every method was tested, but it has been found that there are several glitches at some point in the detection, and we have obtained the most desirable output in the Canny edge detection method.

The values that were given by the canny edge detection are stored separately by assigning two arrays for broadcast news and query video shots. The real difference lies in the fact that the values of each edge of an image/frame are different.

$$ARRAY1[] = [(ES1), (ES2), ..., (ESn)] \tag{5.2}$$

$$ARRAY2[] = [(EB1), (EB2), ..., (EBn)] \tag{5.3}$$

n = frame no of the video being processed

ES = no of edges detected of the query news/commercial shot

EB = no of edges detected of the broadcast news

5.3.2 Edge similarity algorithm

The no of detected edges of the query news/commercial shot and the similar shot present in the broadcast news theoretically seem equal, but this is not the

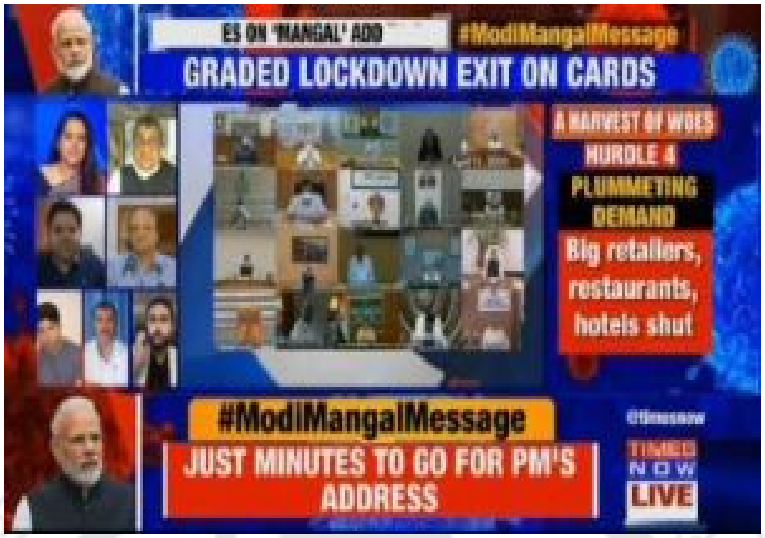


Fig. 5.3: Extracted news frame.

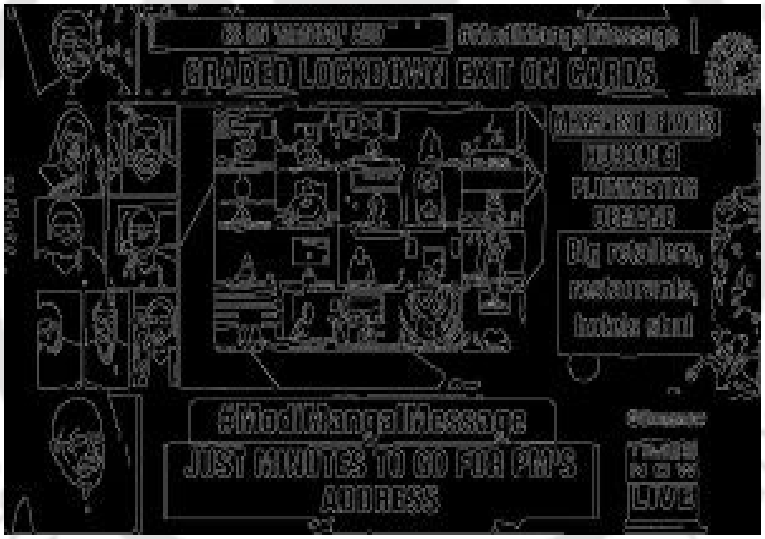


Fig. 5.4: Edges detected in the news frame.

scenario in a practical approach. The no of edges detected may have some discrepancies (*i.e.,* there may be a slight error in the no of edges) despite the frames being similar. To overcome this discrepancy, we have developed an algorithm for checking the similarity between the number of detected edges of the query video shot and the broadcast news.

In our algorithm, we calculated the difference between the number of detected edges of the query video shot and the broadcast news for each frame and

subsequently increased the frame number of the broadcast news. The sum of the n (the no of frames in the query video shot) differences is then computed and stored in an array. The minimum values in the array(*i.e.,* edge similarity) up to a certain threshold mark the point of occurrence of the news/commercial shot in the broadcast news video. The index value of the array gives the frame number of the point of occurrence. This frame number can be used to calculate the exact time at which the news/commercial shot had been broadcast in the broadcast news.

$$Sum[i] = \sum Mi - Ni, \sum Mi + 1 - Ni, \sum Mi + 2 - Ni... \tag{5.4}$$

where,

Mi = No of edges detected in the broadcast news

Ni = No of edges detected in the query video shot

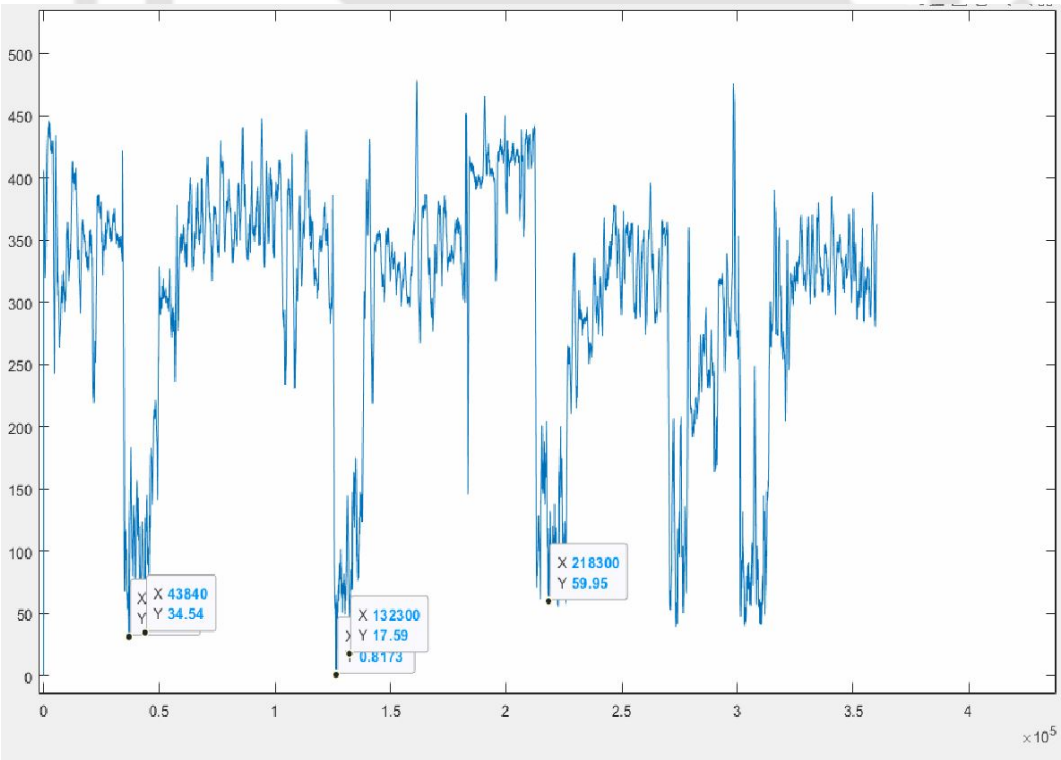


Fig. 5.5: Plot of frame no vs the sum of the differences

5.3.3 Cross checking algorithm for detected frames

After applying the aforementioned rectification algorithm to a broadcast news video, it was observed that anomalous frames were occasionally being detected, and certain desired frames were not being detected at all. We employed the SSIM (Structural Similarity) algorithm to eliminate these anomalies from the output. SSIM serves as a metric for measuring structural similarity by comparing local patterns of pixel intensities that have been normalized for luminance and contrast.

To implement the SSIM algorithm effectively, we adjusted the threshold of our edge similarity algorithm to prevent any desired frames from being overlooked by the edge similarity algorithm. However, this threshold adjustment resulted in the identification of an increased number of anomalous frames. This is where the SSIM algorithm came into play.

The first frame of the video shot is taken as a reference for implementing the SSIM algorithm. Out of the frames detected by the edge similarity algorithm, only the first frame is taken for every shot (*i.e.,* a series of continuous frames being detected), and ssim is performed with the reference. The SSIM algorithm then returns a measure of the similarity based on their structure between the input frames in the range of 0 to 1. Based on this similarity measure, anomalous frames can easily be removed from the frames detected by our edge similarity algorithm by assigning a suitable threshold.

5.4 Video Searching and Retrieval using Scene Classification in Broadcast News Videos.

Image and video indexing techniques are essential in content-based searching in multimedia databases. A novel scheme is proposed for video searching and retrieval using scene classification. For scene classification [86], DCT features of the key frame are extracted to train the classifier, and the feature-length is reduced to a

low dimensional feature. An indexing method with scene classification is proposed using this low-dimensional feature. Subsequently, a novel method for video retrieval based on query entropy is proposed. Supervised classification with low-dimensional features is employed to enhance retrieval accuracy and reduce complexity instead of clustering vector approximations. A modified OVA-file method is then introduced using scene classification techniques, proving to be more accurate and less complex than existing methods.

5.4.1 Proposed method.

The overall block diagram of our proposed method is shown in Fig.5.6. In our proposed method, an image or video can be given as a query to this system. Initially, all (or some fixed frames) the keyframes of the videos in the database are used to train the classifier using the discrete cosine transform feature. The DCT feature for the given query image is calculated, and subsequently, its low dimensional form is used to classify the scene by using a classifier. Then, a k nearest neighbor (kNN)-based query algorithm filters the vast majority of vectors on the low-dimensional vectors of the same class. The next phase identifies the retrieved results by computing the exact distances of all the remaining vectors with the query vector in the high dimensional form.

An efficient video clip retrieval method is proposed for a typical query by video clip, which consists of multiple frames based on scene classification and the entropy of the image. In this, temporal information of the video shot is also considered. A novel query by video retrieval method is proposed, which uses the entropy of the image for searching. An extensive performance study on several data sets confirms the effectiveness of our method compared to that of the OVA-file and VA-file-based methods.

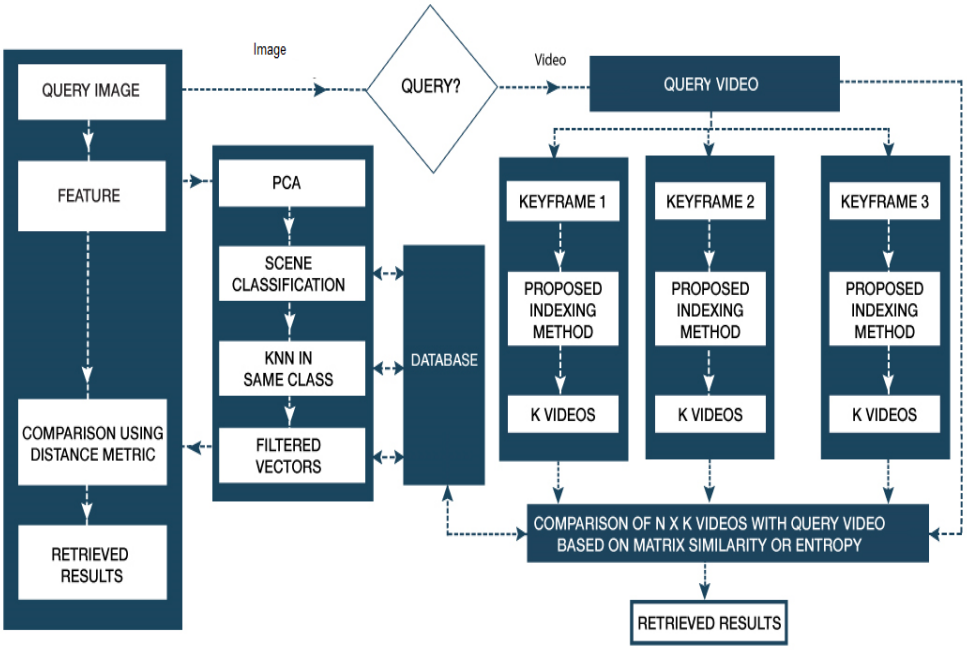


Fig. 5.6: Overall block diagram of the proposed video retrieval system.

5.4.2 Scene classification using DCT feature

The information obtained in the frequency domain has been successfully exploited to holistically encode the content of natural scenes to obtain a vigorous representation for scene classification[49]. The authors exploit an all-encompassing representation of the scene within the discrete cosine transform space utterly congruous with the JPEG format. In [49], it was experimentally observed that the natural scenes spectrum is quite isotropic, whereas the artificial scenes spectrum has a strong horizontal and vertical axis. An information set containing distinctive fundamental classes of scene categories (*i.e.,* city, mountain, forest, and building) may be considered in this respect.

Additionally, test results in [49] illustrated that there exists a connection between the spatial envelopes of structures within scenes (*i.e.,* edges) and the degree of extension, roughness, and openness. Specifically, the spatial envelope of the structures within scenes is highly valuable for distinguishing between indoor and

outdoor scenes. In some particular cases, the differentiation between natural and artificial scenes is based on the observation that straight and vertical lines dominate man-made structures, whereas most natural scenes exhibit textured areas and undulating contours.

The features, dominant orientation of the block, and strength of the orientation, which are extracted directly from the compressed 8×8 DCT block, are used to represent each 8×8 spatial block in the image. The proportion between the sum of the DCT coefficients compared to the horizontal and vertical frequencies is used to represent the tangent of the local dominant orientation (LDO) angle of a 8×8 spatial block. The strength of the local dominant orientation (LDO) of a 8×8 spatial block is related to the overall AC energy of each block. The DCT coefficients $H_k(u, v)$ of each block $I_k(x, y)$ in the original image can be generated by the following equation:

$$H_k(u,v) = \alpha(u)\alpha(v) \sum_{x=0}^{7} \ldots \sum_{y=0}^{7} I_k(x,y) \cos\left(\frac{\pi(2x+1)u}{16}\right) \times \cos\left(\frac{\pi(2x+1)u}{16}\right) \tag{5.5}$$

where,

$$\alpha(f) = \begin{cases} \frac{1}{\sqrt{8}}, & f = 0 \\ \frac{1}{\sqrt{4}}, & 1 \leq f \leq 7 \end{cases}$$

From this representation, the following equation can be used to obtain the edge orientation of $B_k(x, y)$

$$\tan(\theta_k) = \frac{\sum_{u=1;u=u+2}^{7} H_k(u,0) I_k(x,y)}{\sum_{v=1;v=v+2}^{7} H_k(0,v)} \tag{5.6}$$

The strength of the edge is represent by the local variance(the AC energy) of each DCT block. Orientation of the block can be evaluated by (5.6). The strength

can be used to weight each of the edges by its importance. The following equation is used to evaluated each block strength:

$$A_k = \sum_{u=1}^{7} {H_k}^2(u,0) + \sum_{v=1}^{7} {H_k}^2(v,0) + \sum_{v=1}^{7}\sum_{u=1}^{7} {H_k}^2(u,v) \tag{5.7}$$

By analyzing the distribution of the LDO weights, cosidering their corresponding strengths, a holistic representation of the scene can be built. Let a gray scale image I coded with L blocks in 8×8 DCT domain. The L numbers of LDOs $\{\theta_1, \theta_2, ...\theta_L\}$ are extracted by using (5.6) and The L numbers of AC energies $\{A_1, A_2, ..., A_L\}$ are extracted by using (5.7).

The d-dimensional feature vector $LDO(DCT_{8\times8}(I)) = \left[f_{\hat{\theta}_1}, f_{\hat{\theta}_2}, ., f_{\hat{\theta}_d}\right]^T$ of the whole image I is obtained as follows:

$$f_{\hat{\theta}_i} = \frac{N(\theta_i)}{SN}, \quad \forall i \in \{1, ., d\} \tag{5.8}$$

where, $N(\hat{\theta}_i) = \sum_{A_k \in \Theta_i} \log(A_k)$

$$\hat{\theta}_i \in [-90, 90]\,,\ \hat{\theta}_1 = -90,\ \hat{\theta}_{i+1} = \theta_i + \frac{180}{d}, \hat{\theta}_{d+1} = 90$$

$$\Theta_i = \left\{ A_k | \hat{\theta}_i < \hat{\theta}_k \leq \hat{\theta}_{i+1}, A_k > \xi, k = 1, 2, ...L \right\}$$

$SN = \sum_{n=1}^{d} N(\hat{\theta}_n)$: normalization constant.
d : Orientation bins number.
ζ : Marginal orientations discard threshold .
Here, $d = 32$, $z = 10\%$ of the maximal A_k extracted from the image I and $L = 3$. The similarity measure is extracted based on Bhattacharyya coefficient, .

5.4.3 Proposed indexing method with scene classification

For the OVA-file method, clustering the vector approximations and visiting only some clusters according to the given query substantially reduce the computations

compared with the VA-file and speed up the overall retrieval process. However, the accuracy of the retrieved results will depend on the clusters visited. To improve the efficiency of this method, a supervised classification technique is proposed in place of clustering. The proposed method uses the feature vector described in the previous section. The block diagram of the proposed technique is given in Fig.5.7.

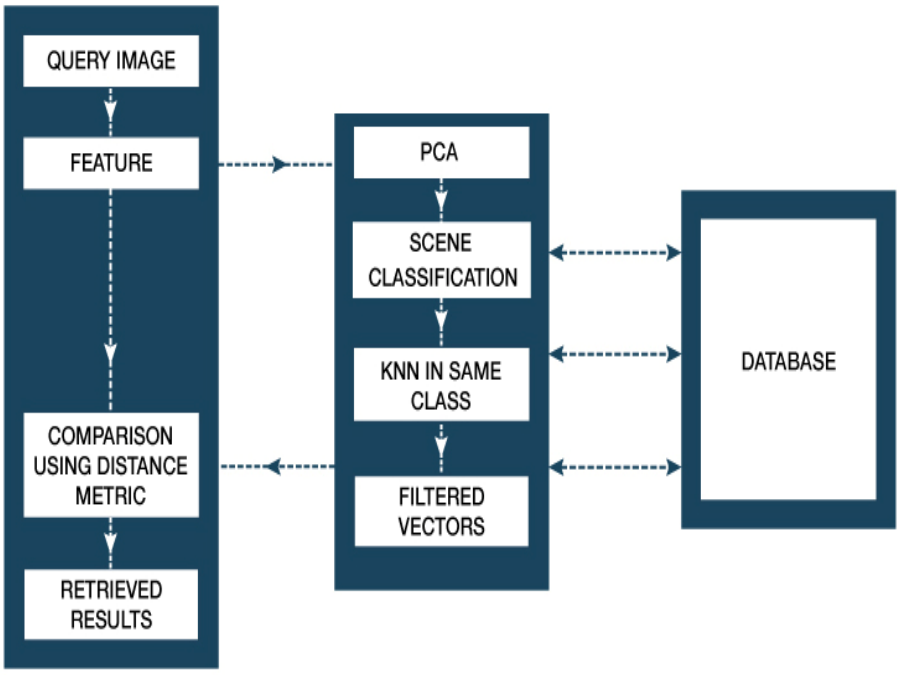


Fig. 5.7: Block diagram of the proposed indexing method with scene classification.

In this method, the images or key frames in the database are used to train the classifier by their feature vector as described in the above section. When an image is given as query, its feature vector is calculated, and then it is placed in one of the classes by the trained classifier. Next, N neighborhood vectors from that classified section are found. Then, the original vector file for these N vectors is visited and subsequently k nearest vectors are found by using the distance metric. Finally, the corresponding similar videos are retrieved. Algorithm 1 illustrates the proposed indexing method.

Algorithm 5 INDEXING METHOD WITH SCENE CLASSIFICATION

Input: q : Query image

k : Maximum number of results to return

Output: h : list of similar videos $(V_1, V_2, ...V_k)$.

STEPS:

1. Extract the feature vector for query image q
2. Project the features into a low dimensional space by PCA.
3. Classify image into one of the classes by using a SVM.
4. Perform kNN in the same class and find the nearest vectors using a Euclidean distance metric.

5.Perform kNN on these vectors, with their high-dimensional form, and find the nearest vectors.

6.Push videos corresponding to these results into

Return h***:***

5.4.4 Searching algorithm based on entropy

A novel searching algorithm for video clip retrieval by incorporating temporal information is proposed, which is entirely based on entropy. Entropy is a measure of randomness. Thus, maximum entropy means maximum information. The entropy of image is calculated by using its color histogram as given in (5.9).

$$E(f_i) = -\sum_{k=1}^{n} h_k \log\, h_k \tag{5.9}$$

Where, h_k represents the proportion that the pixel of value k accounts for the total number of image pixels. Image entropy is obtained by calculating H, S, V components of the image respectively by using (5.9). Thus, maximum entropy of an image implies maximum information in the image. The block diagram describing the proposed method is shown in Fig.5.8. Algorithm 6 also highlights the proposed method.

Algorithm 6 PROPOSED VIDEO SEARCHING ALGORITHM (Q, k)

Input: (1) Query video $Q = \{q_1, q_2, ., q_n\}$ where,

$q_1, q_2, ., q_n$ are key frames of the video.

(2) k Maximum number of retrieved results

Output: h : list of similar videos $(V_1, V_2, ...V_k)$.

STEPS:

1. Perform proposed indexing method on all key frames individually and find $h = \{V_1, V_2, ., V_{n\times k}\}$ i.e., $n \times k$ videos from the database.

2. Find key frame with maximum entropy q_{max} from Q

3. Compare Q with a video V_i in h , i.e., q_{max} is compared with all the key frames of V_i by using a distance metric and a nearest v_{imax} is found.

4. Compare frame left of q_{max} , i.e., q_{max-1} with the set of key frames of V_i which are left from $v_{i,max}$. Then, $v_{i,max-1}$ i.e, the frame with maximum similarity is found. Similarly, the matches are found for the frames $q_{max-2}, .., ..1$.

5. Compare the frames right of q_{max} with the key frames that are located right to $v_{i,max}$, and the most similar key frames of V_i corresponding to Q are found.

6. Calculate similarity between $Q = \{q_1, ., q_{max}, ., q_l\}$ with $V_i = \{v_{i,1}, ., v_{i,max}, ., v_{i,l}\}$ for $l \leq n$ by

$$e^{-d(q_1,v_{i,1})} \times diff(1,2) + e^{-d(q_2,v_{i,2})} \times diff(1,2)...$$
$$\times diff(2,3) + \ + e^{-d(q_n,v_{i,n})} \times diff(n-1,n)$$

where

$$diff(t, t+1) =$$

$$1 - \left(\frac{(q_{t+1}time - q_t time) - (v_{i,t+1}.time - v_{i,t}time)}{(q_{t+1}time - q_t time) + (v_{i,t+1}.time - v_{i,t}time)} \right)^2 ;$$

for $t = 1, 2, ., l-1$

$d(q_n, v_{i,n})$ = distance between q_n and $v_{i,n}$.

$q_n \cdot time$ and $v_{i,n} \cdot time$ are the indices of the key frames.

Here, q_n is in the video Q and $v_{i,n}$ is in the video V_i .

7. Find similarity with all the videos in h and the most k similar videos are determined.

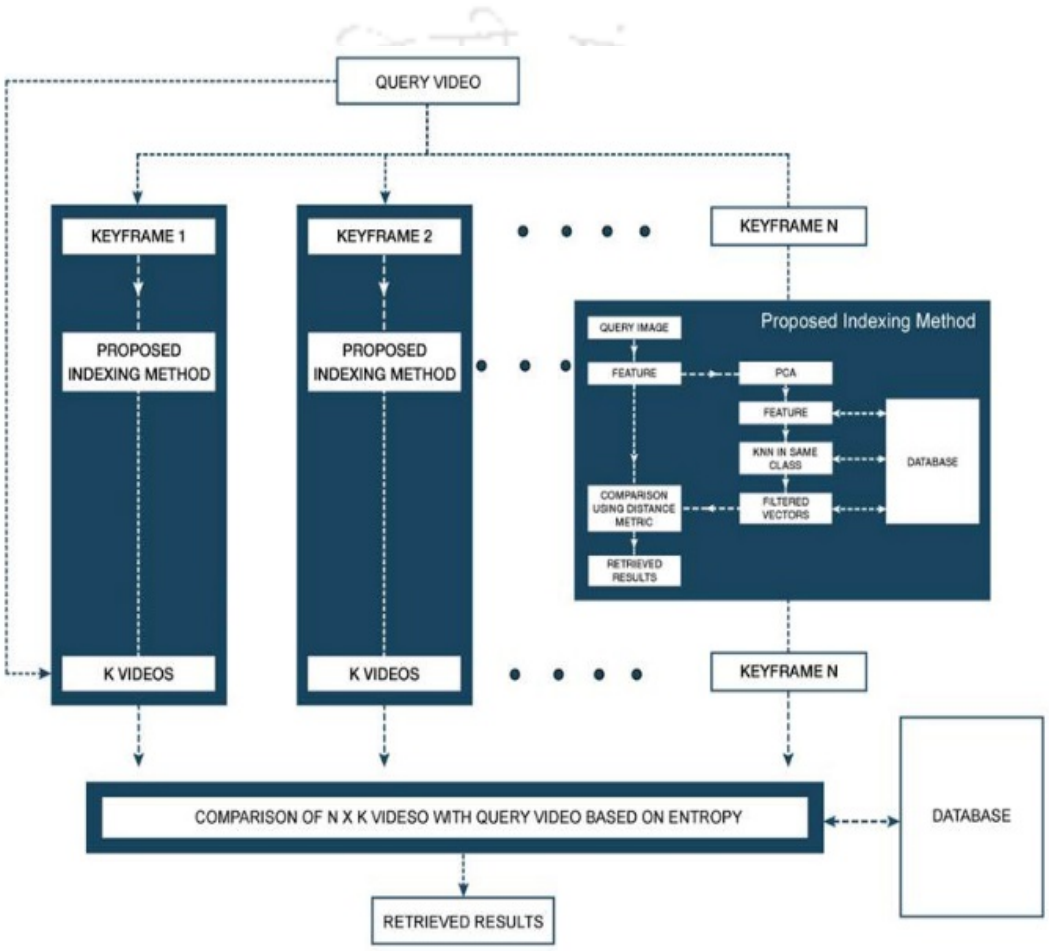


Fig. 5.8: Block diagram of the proposed searching scheme.

5.5 Performance Evaluation

5.5.1 Performance evaluation: Pixel mapping based image and edge based video clip searching and retrieval in broadcast news video

We have tested our algorithm on a 4−hour video of the Times Now news channel. The frame rate of the video was 25 fps with a total no of 360000 frames. The results have been calculated for varying lengths of video shots present in the broadcast news. Accordingly, the results have been tabulated for the different algorithms being employed for the search of the video shot in the broadcast news video. The overall performance of our algorithm can be calculated based on a few factors termed Precision, Recall, and F-measure. The results obtained in the image and video-based searching are tabulated in Table 5.1

Tab. 5.1: Results of Image based searching on a video of 48000 frames

Parameters	**Number of frames**
No of frames searched	6
No of frames detected i.e. TP	6
No of false positive frames i.e. FP	0
No of false negative frames i.e. FN	0
Recall	1
Precision	1
F-measure	1

The results obtained by the edge similarity algorithm is tabulated in Table 5.2. The efficiency of our Edge Similarity algorithm increases with the increasing number of frames in the video shot being searched. But as the shots may be of small duration too. The SSIM algorithm is being employed by adjusting the threshold to

Tab. 5.2: Results of Edge similarity algorithm

Parameters	**Number of frames**
No of frames searched	750
No of frames detected	4*750
No of false positive frames	0
No of false negative frames	0
Recall	1
Precision	1
F-measure	1

minimize false-negative frames, and the results obtained are shown in the tablet 5.3 and Table 5.4.

Tab. 5.3: Results of Edge similarity algorithm

Parameters	**Number of frames**
No of frames searched	235
No of frames detected i.e. TP	5*235
No of false positive frames i.e. FP	6*235
No of false negative frames i.e. FN	0
Recall	1
Precision	0.45
F-measure	0.62

With adjusted threshold for application of SSIM

Tab. 5.4: Results of SSIM algorithm

Parameters	**Number of frames**
No of frames searched	235
No of frames detected i.e. TP	5*235
No of false positive frames i.e. FP	0
No of false negative frames i.e. FN	0
Recall	1
Precision	1
F-measure	1

Application of SSIM on the results of table III

5.5.2 Performance Evaluation: Video searching and retrieval using scene classification in broadcast news videos

A performance evaluation is performed using an image database containing 200 images of each class, viz., indoor, outdoor, natural, and artificial scenes. So, in total, 800 images are considered. Finally, 50 images from each class are used for training, and the 150 images are used for testing. The accuracy of the proposed scene classification method is computed with different feature lengths for classifying Natural vs. Artificial and Indoor vs. Outdoor scenes by using a support vector machine (SVM).

The accuracy of the proposed method for different feature lengths is calculated in Table 5.5. From the sample images database, 50 images are used for training from each class, and the remaining 150 images are given as queries for the VA-file method, OVA-file method, and the proposed method.

Euclidean distance is used as a metric to find the most similar key frames of the videos in the database. Then the accuracy and the average time is taken to retrieve the videos for these three methods are calculated (Table 5.7).

The results show that the OVA-file method is the fastest one, but it lacks accuracy as the comparison is made with only the selected group of clusters in the database. The clusters were formed on the basis of the approximation of data vectors. As expected, the time taken by the VA-file method is greater than the other two methods because comparisons are made with all the vectors in the database. Additionally, its accuracy is higher than that of the OVA-file method. The proposed method is more accurate than the other two approaches and less computationally complex than the VA-file method. For the VA-file method, all the vectors in the database must be compared with the query. By contrast, for the proposed method, the classification is performed initially, and then the comparison is only made with the respective class vectors. Thus, it significantly reduces the complexity.

Fig. 5.9: Retrieval results for a query video using searching algorithm based on entropy

As shown in Fig.5.9, a video with four keyframes is given as input for searching from the database and the retrieved videos are found using the searching algorithm based on entropy. The entropy of the second keyframe of the query is higher than the other keyframes. Hence, the frames similar to this frame are computed first in the retrieved videos. Subsequently, the remaining frames are compared according to temporal information. Similarly, if we consider the second retrieved result, i.e., video 2, the most similar frame with respect to the maximum entropy query frame

Tab. 5.5: FEATURE LENGTH VS ACCURACY

Feature length	**Accuracy (in %)**	
	Natural vs. Artificial	**Indoor vs. Outdoor**
32	94.40	94.04
16	94.10	93.26
8	92.24	92.18
2	92.01	92.08

Tab. 5.7: COMPARISON OF THE PROPOSED METHOD WITH VA-FILE AND OVA-FILE METHODS.

Method	**Avg. time taken**	**Accuracy**
VA-File	$457ms$	81.0%
OVA-File	$225ms$	12 76.4%
Proposed Method	$345ms$	92.3%

is the third keyframe.

Hence, the key frame which is located left to the maximum entropy frame (the first key frame of the query) is compared only with the first and the second key frames of video 2 and a very similar frame is found. The same technique is also applied to the key frames that are located right to the maximum entropy frame. Thus, in the proposed method, similarity is measured using the temporal information. So, the retrieved results are more accurate or visually similar to the given query.

5.6 Conclusions

While working on the broadcast news and video shot of that news, several problems have been faced as the algorithm lined by us is being tested on various

parameters. We have taken 'Times Now' news as the reference for our system. As our algorithm is based upon edge detection and similarity between edge values and further for an absolute result, we have used the structural similarity index algorithm(ssim).

In the initial stage of edge detection, we tested the Canny, Sobel, and Prewitt edge detection methods, but finally, the desired output was achieved through the canny edge detection method. However, there are some discrepancies in the edge value obtained, as at some point, when the anchorperson and its background change, there is a sudden change in edge value. In the testing phase, we also tried to crop the mainframe, removing the text boxes, but we also found a major problem during the split-screen, which required us to analyze the whole frame. But this applied technique solves most of our problems, and it works almost with all news considering a few factors where it fails, like in the split-screen.

After our problems in edge detection were resolved, we moved towards rectifying the edge values obtained in the broadcast news and video shot. We have taken out the value using the difference in the edge value between the video shot and broadcast news frames. But this was not as simple as we had thought. As we stated, there is an abrupt rise in edge values, and we also found a problem in the transition of fade out/in; we needed to drop the idea of the periodicity curve, but the difference in the frames based on similarity is kept on the algorithm. Now, after going through various research papers, we found a solution to our problem by using the structural similarity index. Using SSIM solved many of the problems previously faced by other researchers in their research on news videos. Thus, the algorithm we proposed worked in a decent position, paving the way for a sharp improvement in future research.

The problem of video searching and retrieval is currently very interesting to the research community. Our work aims to build a video searching and retrieval engine to search a video from a large video database. In this paper, the technique of scene classification using a DCT feature is significantly improved by applying a

dimensionality reduction method. A video-searching algorithm with reduced computational complexity is proposed. The performance of the proposed scheme is evaluated by comparing it with other existing data partitioning methods. The proposed system gives better retrieval accuracy compared to these algorithms.

Finally, a method for high dimensional indexing of videos using its keyframes is proposed. The proposed method can retrieve similar videos from a video database without losing the temporal information. The proposed search algorithm entirely depends on the entropy of an image. The performance evaluation in large database query tests shows that this method is efficient and effective for searching for a similar video. The proposed method can easily be used to retrieve similar videos from a database based on the content of the video, and the proposed system may also be used with the existing text-searching methods.

Using high-level rather than low-level features reduces the semantic gap and, hence, improves the system's accuracy. This possible improvement remains for future work.

6

Conclusions

Contents

6.1 Conclusions

The discussed algorithms in this report segmented, categorized, indexed, and combined news shots to form news stories. After segmentation, we discussed a novel retrieval algorithm. The complete work can be summarized below.

- Developed a Shot Segmentation Algorithm, which segments the entire news video into several shots based on content change, i.e., a shot boundary is detected when the content in the news video changes. This is achieved through histogram transformation and comparing two consecutive frames (expressed in YIQ model). If the two frames are similar, they are included under the same shot, and if they are dissimilar, a shot boundary is detected between them. For the measurement of similarity and dissimilarity, cosine comparison is used.

- The segmented shots are categorized as news videos or commercials with the help of our developed Shot Categorization Algorithm. Generally, news pro-

grams display textual information by embedding text within text blocks during the course of the broadcast news. The number of text blocks displayed depends on the section of the news program displayed. For example, commercials or advertisements contain no or at the max one text block, but news broadcasts contain more than two to three text blocks. The shot categorization algorithm first checks for the presence of valid text blocks, and depending on the number of text blocks present, the algorithm classifies the segmented shots accordingly into news shots or commercial shots.

- In the Shot Indexation Algorithm, each shot is processed to extract its frames. All the frames are further processed to extract the local text block and store it after recognizing its text using optical character recognition. The stored words are used to tag the corresponding frame. Those words appearing in more than 50% of the frames will be considered keywords for that shot. After indexing each shot with their respective keywords, each shot was compared based on their keywords in the Story Formation by shot Combination algorithm. Suppose a particular word is repeated in more than five consecutive shots. In that case, all the shots will belong to the same story, or on the contrary, if a particular word does not appear in the next five consecutive shots, a story boundary is marked.

- We have presented a novel technique for story segmentation of news videos. The visual similarity, silence in the audio, and the text in the text boxes of a news video are parameters to define the story boundaries. Each of these parameters is used to create an index, and these three indices are fed to a probabilistic multi-modal algorithm which then predicts the story breaks. The multimodal algorithm takes account of the previous state of the indices and predicts the present state. Then, It is compared with the actual present indices, and the story breaks are determined. The segmented stories are then indexed for easy retrieval the stories.

- Our retrieval algorithm is based on edge detection and similarity between edge values; we have used the Structural Similarity Index algorithm (SSIM) for an absolute result. In the initial stage of edge detection, we tested the canny, Sobel, and Prewitt edge detection methods, but finally, the desired output was achieved through the canny edge detection method. Though there were some discrepancies in the edge value obtained, when the anchorperson and its background changed at some point, there was a sudden change in edge value. In the testing phase, We also tried to crop the mainframe removing the text boxes, but we found a significant problem during the split-screen, so we needed to analyze the whole frame. However, this applied technique solves most of our problems and works with almost all news considering a few factors where it fails, like in the split-screen. After resolving the edge detection problems, we moved towards rectifying the edge values obtained in the broadcast news and video shot. We have taken out the value using the difference in the edge value between the video shot and broadcast news frames. However, this was more complex than we had thought as we stated, there is an abrupt rise in edge values, and also we found a problem in the transition of fade out/in. So, we needed to drop the idea of the periodicity curve, but the difference in the frames based on similarity is kept in the algorithm. After going through various research papers, we solved our problem using the structural similarity index. Using SSIM solved many of the previous problems faced by other researchers in their research on news videos. Thus, our proposed algorithm worked to a decent position, paving a sharp improvement in future research.

- Finally, a method for high dimensional indexing of videos using its keyframes is proposed. The proposed method can retrieve similar videos from a video database without losing the temporal information. The proposed search algorithm entirely depends on the entropy of an image. The proposed method can easily be used to retrieve similar videos from a database based on the content

of a video.

6.2 Future Works

Mere segmentation of shots at the shot boundaries of the news videos is not very helpful in creating a sorted video library or database. Future scope of the this research lies in an approach towards a genre-based classification. This can be achieved based on training datasets from the keywords obtained in this works. Besides this the algorithm can be restructured so that they could be used in real time processing embedded on a set top box of television network.

The development of an adaptive threshold for the Edge Similarity algorithm resulting in alleviating the need of adjusting the threshold for the application of SSIM. Further leading to the development of an efficient algorithm without the need for performing SSIM on the output obtained from the Edge Similarity Algorithm.

Using high level features instead low-level features reduces the semantic gap and, hence improvement remain for the future work.

List of Publications

International Journals

- Pranabjyoti Haloi, M.K. Bhuyan, "Unsupervised story segmentation and indexing of broadcast news video," *Multimedia Tools and Applications, Springer*, 16 Septamber, 2021, https://doi.org/10.1007/s11042-021-11490-y

International Conferences

- Pranabjyoti Haloi, M.K. Bhuyan and Arnab Kisor Bordoloi, "Unsupervised broadcast news video shot segmentation and classification," *2nd (IEEE) International Conference on Innovations in Electronics, Signal Processing and Communication (IESC)*, pp.1-5, March 1-2, 2019.

- Pranabjyoti Haloi, Prathik Gadde, and M.K. Bhuyan, "News video indexing and story unit segmentation using text cue," *(IEEE) International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET 2019)*, pp.1-7, March 21-23, 2019.

- Pranabjyoti Haloi, M.K. Bhuyan, Pooja Rani Borah and Dibyajyoti Chatterjee, "A novel algorithm for shot boundary detection and segmentation of broadcast news videos ," *(IEEE)International Conference on Computational Performance Evaluation (ComPE)*, North-Eastern Hill University, Shillong, Meghalaya, India. July 2–4, 2020

- Pranabjyoti Haloi, M.K. Bhuyan "Video searching and retrieval using scene classification in multimedia databases," *2nd (IEEE) International Conference for Emerging Technology (INCET)*, pp. 1-7, 2021, doi:10.1109/INCET51464.2021.9456317.

- P. Haloi, M. K. Bhuyan, K. Gautam and A. Ali, “Edge based video clip searching and retrieval in broadcast news videos," *International Conference for Advancement in Technology (ICONAT)*, Goa, India, 2022, pp. 1-5, doi: 10.1109/ICONAT53423.2022.9725985.

Bibliography

[1] D. Swanberg, C.-F. Shu, and R. C. Jain, "Knowledgeguided parsing in video databases", *Storage and Retrieval for Image and Video Databases* 1993. (Cited on page 13.)

[2] L. Meng, Y. Cai, M. Wang, and Y. Li, "TV commercial detection based on shot change and text extraction," in Image and Signal Processing, 2009. CISP '09. *2nd International Congress on, pp. 1–5, Oct 2009.* (Cited on page 15.)

[3] N. Ozay and B. Sankur, "Automatic TV logo detection and classification in broadcast videos," in *17th European Signal Processing Conference(EUSIPCO)*, Glasgow, Scotland, pp. 839-843, 2009 (Cited on page 15.)

[4] L. Y. Duan *et al.*, "Segmentation, categorization, and identification of commercial clips from TV streams using multimodal analysis," in *Proceedings of the 14th Annual ACM International Conference on Multimedia, (New York, NY, USA)*, pp. 201–210, ACM, 2006. (Cited on page 15.)

[5] N. Liu *et al.*, "Exploiting visualaudio-textual characteristics for automatic TV commercial block detection and segmentation," *Multimedia, IEEE Transactions*, vol. 13, pp. 961–973, Oct 2011. (Cited on pages 15 and 16.)

[6] A. Jindal, A. Tiwari and H. Ghoosh, "Efficient and language independent news story segmentation for telecast news videos," *Multimedia(ISM), IEEE International Symposium* on, pp. 458-463, Dec 2011. (Cited on pages 15 and 18.)

[7] L. Agnihotri *et al.*, "Evolvable visual commercial detector," in *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, vol. 2, pp. II–79–II–84 vol.2, June 2003. (Cited on page 15.)

[8] P. Duygulu, J.-Y. Pan, and D. A. Forsyth, "Towards autodocumentary: Tracking the evolution of news stories," in Proceedings of the *12th Annual ACM International Conference on Multi-media, MULTIMEDIA' 04*, (New York, NY, USA), pp. 820–827, ACM, 2004. (Cited on page 15.)

[9] L. Xie, A. Natsev, and J. Tesic, "Dynamic multimodal fusion in video search," in *IEEE International Conference on Multimedia and Expo, 2007*, pp. 1499–1502, July 2007. (Cited on page 16.)

[10] A. G. Chifu and S. Fournier, "Segchainw2v: Towards a generic automatic video segmentation framework, based on lexical chains of audio transcriptions and word embeddings," *Procedia Computer Science*, Volume 96, 2016, Pages 1371-1380, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2016.08.182. (Cited on page 16.)

[11] W. Hu *et al.*, "A survey on visual content-based video indexing and retrieval," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 41, pp. 797–819, Nov 2011. (Cited on page 2.)

[12] A. G. Hauptmann and M. A. Smith, "Text, speech and vision for video segmentation : The informedia project," in *AAAI Fall Symposium, Computational Models for Integrating Language and Vision*, pp. 10–12, 1995. (Cited on page 50.)

[13] X. Chen and H. Zhang, "Text area detection from video frames," *Advances in Multimedia Information Processing — PCM 2001*, Volume 2195, ISBN : 978-3-540-42680-62001. (Cited on page 50.)

[14] A. Hanjalic, R. Lagendijk, and J. Biemond, "Automated high-level movie segmentation for advanced videoretrieval systems," *Circuits and Systems for Video Technology*, IEEE Transactions, vol. 9, pp. 580–588, Jun 1999. (Cited on pages 4 and 15.)

[15] Brezeale, D. J. Cook, and S. Member, "Automatic video classification A survey of the literature," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 38, no. 3, pp. 416-430, May 2008, doi: 10.1109/TSMCC.2008.919173. (Cited on page 4.)

[16] H. Zhang *et al.*, "Automatic parsing of news video," *Proceedings of the International Conference on Multimedia Computing and Systems*, 1994, pp. 45–54, May 1994. (Cited on pages xvii, 4, 13, 14, 15, 16 and 17.)

[17] B. Gunsel, A. Mufit Ferman, and A. Tekalp, "Video indexing through integration of syntactic and semantic features,"*In Proceedings 3rd IEEE Workshop on Applications of Computer Vision*, 1996. WACV 96., pp. 90–95, Dec 1996. (Cited on page 14.)

[18] X. Gao and X. Tang, "Unsupervised video-shot segmentation and model-free anchorperson detection for news video story parsing,"*IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, pp. 765–776, Sep 2002. (Cited on pages 15, 22 and 57.)

[19] W. Hu *et al.*, "A survey on visual contentbased video indexing and retrieval," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 41, pp. 797–819, Nov 2011. doi: 10.1109/TSMCC.2011.2109710. (Cited on pages 16, 17, 18 and 28.)

[20] M. Cooper, T. Liu, and E. Rieffel, "Video segmentation via temporal pattern classification," *IEEE Transactions on Multi-media*, vol. 9, pp. 610–618, April 2007. (Cited on page 16.)

[21] J. Yuan *et al.*, "A formal study of shot boundary detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, pp. 168–186, Feb 2007. (Cited on page 16.)

[22] A. F. Smeaton, P. Over, and A. R. Doherty, "Video shot boundary detection: Seven years of TRECVid activity," *Computer Vision and Image Understanding*, vol. 114, no. 4, pp. 411 – 418, 2010. (Cited on page 16.)

[23] T. Zlitni, B. Bouaziz, and W. Mahdi, "Automatic topics segmentation for tv news video using prior knowledge," *Multimedia Tools and Applications*, vol. 75, no. 10, pp. 5645–5672, 2016. (Cited on pages 17 and 18.)

[24] F. Colace, P. Foggia, and G. Percannella, "A probabilistic framework for tvnews stories detection and classification," in *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on. IEEE,* 2005, pp. 1350–1353. (Cited on page 21.)

[25] G.-J. Poulisse *et al.*, "News story segmentation in multiple modalities," *Multimedia Tools and Applications*, vol. 48, no. 1, pp. 3–22, 2010. (Cited on page 22.)

[26] L. Chaisorn, T.-S. Chua, and C.-H. Lee, "A multi-modal approach to story segmentation for news video," World Wide Web, vol. 6, no. 2, pp. 187–208, 2003. (Cited on page 22.)

[27] H. Misra *et al.*, "TV news story segmentation based on semantic coherence and content similarity," in *International Conference on Multimedia Modeling. Springer*, 2010, pp. 347– 357. (Cited on page 23.)

[28] C. Ma *et al.*, "A detection-based approach to broadcast news video story segmentation," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on. IEEE,* 2009, pp. 1957–1960. (Cited on page 23.)

[29] N. Dimitrova *et al.*, "Applications of video-content analysis and retrieval," *MULTIMEDIA, IEEE*, vol. 9, pp. 42–55, Jul 2002. (Cited on pages 17 and 18.)

[30] V. Mezaris *et al.*, "Real-time compressed-domain spatiotemporal segmentation and ontologies for video indexing and retrieval," *Circuits and Systems for Video Technology, IEEE Transactions* on, vol. 14, pp. 606–621, May 2004 (Cited on pages 17 and 18.)

[31] I. Mironica *et al.*, "Fisher kernel based relevance feedback for multimodal video retrieval," in *Proceedings of the 3rd ACM Conference on International Conference on Multimedia Retrieval*, ICMR '13, (New York, NY, USA), pp. 65–72, ACM, 2013. (Cited on pages 17 and 18.)

[32] B. T. Truong and S. Venkatesh, "Video abstraction: A systematic review and classification," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 3, Feb. 2007. (Cited on pages 17 and 18.)

[33] L. Chaisorn *et al.*, "A hierarchical approach to story segmentation of large broadcast news video corpus," in *Multimedia and Expo, 2004. ICME '04. 2004 IEEE International Conference* on, vol. 2, pp. 1095– 1098 Vol.2, June 2004. (Cited on pages 17, 19, 25 and 105.)

[34] B. Feng *et al.*, "Multi-modal information fusion for news story segmentation in broadcast video," in *Acoustics, Speech and Signal Processing (ICASSP), IEEE International Conference* on, pp. 1417– 1420, March 2012 (Cited on page 20.)

[35] T.-S. Chua *et al.*, "Story boundary detection in large broadcast news video archives: Techniques, experience and trends," in *Proceedings of the 12th Annual ACM International Conference on Multimedia, MULTIMEDIA '04*, (New York, NY, USA), pp. 656–659, ACM, 2004. (Cited on pages xvii, 20 and 21.)

[36] L. Chaisorn *et al.*, "A two-level multi-modal approach for story segmentation of large video corpus," in *Proceedings of TRECVID workshop 2003,*. (Cited on pages xvii and 21.)

[37] W. Hsu *et al.*, "Discovery and fusion of salient multi-modal features towards news story segmentation," in *Proc. SPIE 5307, Storage and Retrieval Methods and Applications for Multimedia 2004*, 2023, (Cited on pages xvii and 21.)

[38] M. Naphade *et al.*, "Large-scale concept ontology for multimedia," *MULTIMEDIA, IEEE*, vol. 13, pp. 86–91, July 2006. (Cited on page 19.)

[39] A. Goyal *et al.*, "Split and merge based story segmentation in news videos". *Lecture Notes in Computer Science*, 5478, pp. 766-770, 2009. (Cited on pages 22 and 23.)

[40] Y. Song *et al.*, "News story segmentation based on audio-visual features fusion", in *4th International Conference on Computer Science and Education, Nanning*, pp. 1065-1068, 2009. (Cited on pages 23 and 105.)

[41] L. Huayong, Z. Hui, "The Segmentation of News Video into Story Units", in *Advances in Web-Age Information Management, WAIM 2005. Lecture Notes in Computer Science*, vol 3739. Springer, Berlin, Heidelberg. (Cited on page 24.)

[42] H. Liu , T. He, "Content-Based Story Segmentation of News Video by Multimodal Analysis", in *6th International Conference on Fuzzy Systems and Knowledge Discovery, Tianjin.* Vol 7, pp. 423-426. 2009. (Cited on page 24.)

[43] A. Nagasaka , Y. Tanaka "Automatic Video Indexing and Full-Video Search for Object Appearances". *Journal of Information Processing*, vol. 15 (2), p. 316, 1992. (Cited on page 24.)

[44] A. Rosenberg and J. Hirschberg (2006) "Story segmentation of broadcast news in english, mandarin and arabic". In: *Human Language Technology Conference of the NAACL, Companion*, pp. 125-128. (Cited on page 24.)

[45] J. Wang *et al.*, "A multimodal scheme for programsSegmentation and representation in broadcast video streams". *IEEE Transactions on Multimedia*, vol. 10 (3), pp. 393-408. 2008. (Cited on pages 24 and 105.)

[46] Y. Park , Y. Li, "Extracting salient keywords from instructional videos using joint text, audio and visual cues". in *Human Language Technology Conference of the NAACL, Companion,* pp. 109- 112, 2006. (Cited on page 24.)

[47] É. Dumont, G. Quénot" Automatic story segmentation for TV news video using multiple modalities". *International Journal of Digital Multimedia Broadcasting*, vol. 2012, Article ID 732514. (Cited on pages 22, 25 and 105.)

[48] L. Chaisorn , T. Chua *et. al.*,"A two-level multi-modal approach for story segmentation of large news video corpus, TRECVID", 2003. (Cited on page 25.)

[49] AG. Hauptmann , MJ. Witbrock, "Story segmentation and detection of commercials in broadcast news video", in *IEEE International Forum on Research and Technology Advances in Digital Libraries -ADL'98*, Santa Barbara, CA, USA, pp. 168-179, 1998. (Cited on pages 25, 26, 105 and 117.)

[50] H. Feng *et al.*, "Story segmentation in news video". in *International Conference on Neural Networks and Brain, Beijing*, pp. 831-835, 2005. (Cited on pages 25 and 105.)

[51] N. O'Connor *et al.*, "News story segmentation in the fischlar video indexing system". in *International Conference on Image Processing* , vol. 3, pp. 418-421, 2001. (Cited on pages 25, 27 and 105.)

[52] Jen-Hao Yeh *et al.*, "TV commercial detection in news program videos". In:*IEEE International Symposium on Circuits and Systems (ISCAS)* Kobe, Japan, 2005, pp. 4594-4597 Vol. 5, 2005. (Cited on pages 25 and 26.)

[53] PY. Hui *et al.*, "Automatic story segmentation for spoken document retrieval", in*10th IEEE International Conference on Fuzzy Systems.* (Cat. No.01CH37297), Melbourne, Victoria, Australia, vol.2, pp. 1319-1322, 2001 (Cited on pages 25 and 26.)

[54] Y. Zhai , A. Yilmaz , M. Shah, "Story segmentation in news videos using visual and text cues", in *Image and Video Retrieval: 4th International Conference, CIVR 2005, Singapore*, vol 3568. pp. 92-102, 2005. (Cited on pages 26 and 105.)

[55] R. Tapu , B. Mocanu , T. Zaharia "TV news retrieval based on story segmentation and concept association", in *12th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, pp. 327-334, 2016 (Cited on pages 26 and 105.)

[56] Zedan *et al.*, "News Videos Segmentation Using Dominant Colors Representation", in *Advances in Soft Computing and Machine Learning in Image Processing. Studies in Computational Intelligence*, vol 730, pp. 89-109, 2018. (Cited on pages 27 and 105.)

[57] R. Kannao and P. Guha, "Story segmentation in TV news broadcast", in *23rd International Conference on Pattern Recognition (ICPR)*, pp. 2948-2953, 2016. (Cited on pages 27 and 28.)

[58] R. Kannao and P. Guha, "Segmenting with style: detecting program and story boundaries in TV news broadcast videos". *Multimedia Tools and Applications*, 31925–31957, 2019. (Cited on pages 27 and 28.)

[59] M. Jalil, A. Butt, A. Malik, "Short-time energy, magnitude, zero crossing rate and autocorrelation measurement for discriminating voiced and unvoiced segments of speech signals", in *The International Conference on Technological Advances in Electrical, Electronics and Computer Engineering (TAEECE)*, Konya, pp. 208-212, 2013. (Cited on page 94.)

[60] Fahad Al Mahmood, *Dictionary (https:// www.mathworks.com/ matlabcentral/ fileexchange/ 5408-dictionary), MATLAB central file exchange. Retrieved* September 13, 2020. (Cited on page 98.)

[61] "Text analytics toolbox user's guide (2020)". *https://in.mathworks.com /help/pdf_ doc/textanalytics/textanalytics_ ug.pdf. Retrieved* September 13, 2020. (Cited on page 99.)

[62] "Claro TV - T18 ". *https://www.iball.co.in/core/File/ProductPdf/ClaroTV-T18.pdf. Retrieved*, June 9, 2021. (Cited on page 99.)

[63] "Image processing toolbox user's guide", *https://in.mathworks.com /help /pdf_ doc/images/images_ ug.pdf. Retrieved* June 9, 2021. (Cited on page 99.)

[64] "Parallel computing toolbox user's guide", *https://www.mathworks.com/help/ pdf_ doc/parallel-computing/parallel-computing.pdf. Retrieved*, June 9, 2021. (Cited on page 99.)

[65] Chien-Chuan Ko and Wen-Ming Xie, "News video segmentation and categorization techniques for content demand browsing, "*Congress on Image and Signal Processing*, vol. 2, pp. 530-534, 2008. (Cited on page 29.)

[66] P. Melin, *et al.*, "Edge detection method for image processing based on generalized type-2 fuzzy logic" *IEEE Transactions on Fuzzy Systems*, vol. 22, pp. 1515-1525, 2014. (Cited on page 29.)

[67] P. Ganesan and G. Sajiv, "A comprehensive study of edge detection for image processing applications" *2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, pp. 1-6 2017. (Cited on page 30.)

[68] Z. Wang *et al.*, "Image quality assessment: from error visibility to structural similarity" *IEEE Transactions on Image Processing*, vol. 13, pp. 600-612, 2004. (Cited on page 30.)

[69] Yadav *et al.*, "Implementing edge detection for medical diagnosis of a bone in Matlab", *5th International Conference and Computational Intelligence and Communication Networks*, pp. 270-274, 2013. (Cited on page 30.)

[70] A.W.M. Smeulders *et al.*, "Semantic video searching", *14th International Conference on Image Analysis and Processing,*, pp. 51-58, 2007 (Cited on page 30.)

[71] A. Vyas *et al.*, "Commercial block detection in broadcast news videos", *the Indian Conference*, pp. 1-7, 2014. (Cited on page 30.)

[72] T. Sahoo and S. Pine,"Design and simulation Of various edge detection techniques using Matlab simulink".*"2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, pp. 1224-1228, 2016. (Cited on page 30.)

[73] Y. Song, W. Wang and F. Guo, "News story segmentation based on audio-visual features fusion", *"Proceedings of 2009 4th International Conference on Computer Science and Education*, pp. 1065-1068, 2009 (Cited on page 31.)

[74] Z.Wang, A.C. Bovik, H.R.Sheikh,"Image quality assessment: from error visibility to structural similarity," *IEEE Tansactions on Image Processing*, vol. 13, pp. 600-612,2004. (Cited on page 29.)

[75] Z. Cao, and M. Zhu, "An efficient video similarity search algorithm," *IEEE Transactions on Consumer Electronics.*, pp. 751-755, vol. 56, no.2, 2010. (Cited on page 28.)

[76] A. Lee, R.-W. Hong, and M.-F. Chang, "An approach to content-based video retrieval." *International Conference on Advances in Pattern Recognition*, vol. 1, pp. 273–276, 2004. (Not cited.)

[77] A. Dyana, M. Subramanian, and S. Das, "Combining features for shape and motion trajectory of video objects for efficient content based video retrieval" *International Conference on Advances in Pattern Recognition*, pp. 113–116, 2009. (Not cited.)

[78] A. Jain, A. Vailaya, and W. Xiong, "Query by video clip," *International Conference(IEEE) on Pattern Recognition*, vol. 1, pp. 909-911, 1998.(Cited on page 28.)

[79] K. O. Y. Tonomura, A. Akutsu, and T. Sadakkata,"Video map and video spacelcon: tools for anatomizing video content" *INTERCHI '93 Conference Proceedings*, pp. 131—141, 1993. (Cited on page 28.)

[80] T. Liu, H.J. Zhang, and F. Qi, "A novel video Key-frame-extraction algorithm based on perceived motion energy model," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 10, pp. 1006—1013, 2003. (Cited on page 28.)

[81] W. Wolf, "Key-frame selection by motion analysis," *International Conference on Acoustics, Speech and Signal Processing(IEEE)*, vol. 2, pp. 1228—1231, 1996. (Cited on page 28.)

[82] R. Pan, Y. Tian, and Z. Wang, "Key-frame extraction based on clustering," *International Conference(IEEE) on Progress in Informatics and Computing*, vol. 2, pp. 867—871, 2010. (Cited on page 28.)

[83] H. Yuan, and X. P. Zhang, "Statistical modeling in the wavelet domain for compact feature extraction and similarity measure of images," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 3, pp. 439-445, 2010. (Cited on pages 28 and 31.)

[84] Z. W. Jianhua Wu, and Y. Chang, "Color and texture feature for content based image retrieval," *International Journal of Digital Content Technology and Its Applications*, vol. 4, 2010. (Cited on page 31.)

[85] T. Chang, and C. Kuo, "A wavelet transform approach to texture analysis," *International Conference(IEEE) on Acoustics, Speech, and Signal Processing*, vol. 4, pp. 661–664, 1992. (Cited on pages 28 and 31.)

[86] G.M. Farinella, and S. Battiato, "Scene classification in compressed and constrained domain," *Computer Vision, IET*, vol. 5, no. 5, pp. 320-334, 2011. (Cited on pages 31 and 115.)

[87] N. Koudas *et al.* "LDC : Enabling search by partial distance in a hyper-dimensional space," *Proceedings of the International Conference on Data Engineering*, pp. 6–17, 2004. (Cited on page 31.)

[88] X. Chen *et al.*, "A human-centered multiple instance learning framework for semantic video retrieval," *IEEE Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews*, vol. 39, no. 2, pp. 228–233, 2009. (Cited on page 31.)

[89] Y. Song *et al.*, "Semi-automatic video annotation based on active learning with multiple complementary predictors" *ACM International Workshop on Multimedia Information Retrieval*, pp. 97-104, 2005. (Not cited.)

[90] Y. Song, and H. Zhang, "Efficient semantic annotation method for indexing large personal video database," *ACM International Workshop on Multimedia Information Retrieval*, pp. 289-296, 2006. (Cited on page 31.)

[91] R. Weber *et al.*, "A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces" *International Conference on Very Large Data Bases*, pp. 194-205, 1998. (Cited on page 31.)

[92] H. Lu *et al.*, "Hierarchical indexing structure for efficient similarity search in video retrieval," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 11, pp. 1544–1559, 2006. (Cited on pages 31 and 32.)

[93] U. Gargi *et al.*, "Performance characterization of video-shot-change detection method," *IEEE transactions on circuits and systems for video technology*, vol. 10, pp. 1–13, 2000. (Cited on page 57.)

[94] MR. Naphade *et al.*, "A high-performance shot boundary detection algorithm using multiple cues," *Proceedings 1998 International Conference on Image Processing. ICIP98 (Cat. No. 98CB36269)*, vol. 1, pp. 884–887, 1998. (Cited on page 57.)

[95] Y. Avrithis *et al.* "Broadcast news parsing using visual cues: A robust face detection approach," *2000 IEEE International Conference on Multimedia and Expo. ICME2000. Proceedings. Latest Advances in the Fast Changing World of Multimedia (Cat. No. 00TH8532)* vol. 3, 2000, pp. 1469–1472. (Cited on page 57.)

[96] J. Mas and G. Fernandez, "Video shot boundary detection based on color histogram," *Notebook Papers TRECVID2003,* 2003. (Cited on page 57.)

[97] Partha Pratim Mohanta *et al.*, "A model-based shot boundary detection technique using frame transition parameters," *IEEE Transactions on multimedia*, vol. 14, no. 1, February. (Cited on page 57.)

[98] Krishna K. Warhadea, *et al.*, "Performance evaluation of shot boundary detection metrics in the presence of object and camera motion," *IETE Journal of Research,* vol. 57, pp. 461-466, 2011. (Cited on page 57.)

[99] John S. Boreczhy *et al.*, "A hidden Markov model framework for video segmentation using audio and image features," *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'98 (Cat. No. 98CH36181)*, vol. 6, pp. 3741-3744, 1998.
(Cited on page 57.)