Video Index Point Detection and Extraction Framework using Custom YoloV4 Darknet Object Detection Model

MEHUL MAHRISHI¹ (Senior Member, IEEE), SUDHA MORWAL², ABDUL WAHAB MUZAFFAR³, SURBHI BHATIA ⁴, PANKAJ DADHEECH ⁵, MOHAMMAD KHALID IMAM RAHMANI ⁶ (SENIOR MEMBER, IEEE)

¹Department of Information Technology, Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur, Rajasthan, India (e-mail: mehul@skit.ac.in)
²Department of Computer Science, Banasthali Vidyapith, Niwai, Rajasthan, India (e-mail: sudhamorwal45@gmail.com)
³College of Computing and Informatics, Saudi Electronic University, Riyadh 11673, Saudi Arabia (email: a.muzaffar@seu.edu.sa)
⁴College of Computer science and information technology, King Faisal University, Saudi Arabia. (email : sbhatia@kfuf.edu.sa)
⁵Department of Computer Science & Engineering,Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur, Rajasthan, India (email : pankajdadheech777@gmail.com)
⁶College of Computing and Informatics, Saudi Electronic University, Riyadh 11673, Saudi Arabia (email : m.rahmani@seu.edu.sa)

Corresponding author: Dr. Surbhi Bhatia (e-mail: sbhatia@kfuf.edu.sa).

ABSTRACT The trend of learning from videos instead of documents has increased. There could be hundreds and thousands of videos on a single topic, with varying degrees of context, content, and depth of the topic. The literature claims that learners are nowadays less interested in viewing a complete video but prefers the topic of their interests. This develops the need for indexing of video lectures. Manual annotation or topic-wise indexing is not new in the case of videos. However, manual indexing is time-consuming due to the length of a typical video lecture and intricate storylines. Automatic indexing and annotation is, therefore, a better and efficient solution. This research aims to identify the need for automatic video indexing for better information retrieval and ease users navigating topics inside a video. The automatically identified topics are referred to as “Index Points.” 137-layer YoloV4 Darknet Neural Network creates a custom object detector model. The model is trained on approximately 6000 video frames and then tested on a suite of 50 videos of around 20 hours of run time. Shot Boundary detection is performed using Structural Similarity fused with a Binary Search Pattern algorithm which outperformed the state-of-the-art SSIM technique by reducing the processing time to approximately 21% and providing around 96% accuracy. Generation of accurate index points in terms of true positives and false negatives is detected through precision, recall, and F1 score, which varies between 60-80% for every video. The results show that the proposed algorithm successfully generates a digital index with reasonable accuracy in topic detection.

INDEX TERMS Binary Search Pattern, Keyframe Identification, Neural Network, Optical Character Recognition, Structural Similarity, Shot Boundary Detection, Video Frames, YoloV4.

I. INTRODUCTION

The automatic generation of multimedia content from a captured video lecture is not new. Once stored, videos create an extensive knowledge repository. They are, in itself, a disruptive source of education apart from textbooks and classroom study [1]. One can easily see the considerable attention paid to educational videos amongst all other video sources. The findings show that up to 65 percent of Berkeley students utilize videos to improve their understanding of subjects they missed in class [2]. The Cambridge International Global Education Census Survey 2019 states that approximately 41% of school-goers globally have taken an online course that was not part of their curriculum in the past 12 months [3]. The detailed literature survey in Section II clearly states that there is a paradigm shift in the mode of online learning. Learners are increasingly turning to online video lectures; however, they are less interested in watching the entire video and prefer to focus on specific topics of interest. Therefore, indexing of video lecture is the need of the hour. Mathematically, video indexing and annotation is the probability of discovering a topic and its associated...
text descriptor (keywords) in a video. However, in long video lectures, manually performing this task can be time-consuming. Additionally, new multimedia content demands comprehensive indexing and restructuring of information for a deeper understanding of a video. Therefore, automatic indexing and annotation is a better and effective solution. The literature also claims that deep learning is quite restrictive for such cases even though much work is done in video analysis and segmentation.

A video can be analyzed in two different ways. On the one hand, a video can be examined based on underlying multimedia elements, i.e., speech and images. On the other hand, text inspection and keyword identification open a whole new dimension. This research intends to develop a technique for identifying automatic index points from a video lecture. These index points are nothing more than excerpts from a video’s main headings and discussion topics. The current research involves studying all materials, including variable replay duration, classroom lectures, online tutorials, open courseware, and other comparable resources. A 137-layer YoloV4 Darknet Neural Network is used to create a custom object detector model. The model is trained on approximately 6000 video frames and then tested on a suit of 50 educational videos of around 20 hours of runtime. Structural Similarity Index Matrix algorithm fused with Binary Search Pattern algorithm is used for improving the overall performance of shot boundary detection.

A. RESEARCH CONTRIBUTIONS

The motivation of this research comes from the fact that accessing the range of interest in a lecture video is not easy because there is no table of contents or index sections. The temporal pace of the sequences only affords the user to explore the video linearly. Through this research technique, a video lecture is automatically partitioned into segments based on image and text analysis. This process is called “Index Point Detection”, and the identified keywords are termed “Index Points”. Each ‘Index Point’ will also contain the timestamp of the relative video time instance. The textual content of the videos will be obtained by using Google’s Tesseract OCR Engine. The research is limited to educational videos that contain presentation slides in the English language only. Using these indices, users can easily search topics to find a particular segment of a video file or create a digital database.

Learning blended with Natural Language Processing is undoubtedly the most gripping scientific frontiers that persuade the researchers to explore novel dimensions in indexing and partitioning, thereby producing a coherent video document structure. Thus, the main contribution of this research are:

- To study the existing methods of automatic video indexing and annotation to analyze the outcomes and gaps.
- To develop a methodology through which a video lecture is automatically partitioned into segments based on image and text analysis.
- To identify and implement an accurate shot boundary detection technique for abrupt transitions.
- To automatically detect the index points within a video by validating the trained YoloV4 Custom Object Detector.
- To apply accuracy measures and provide a comparative benchmark for further research in this area and produce a significant impact.
- To provide a method for intelligent information retrieval, researchers can find the applicability of this technique in videos of other domains.
- To support and enhance the research community knowledge.

B. ARTICLE ORGANIZATION

This research work focuses on developing a technique for identifying automatic index points for a video lecture. Section II takes a close look at the theories and related research about video indexing to grasp detailed literature. Section III discusses the technical perspective, research methodology and the technique to facilitate the algorithm’s development based on research gaps identified in the literature and the objectives of the investigation. Since we have used a subjective criterion to evaluate the success of our heading detection algorithm, the hypothesis and ground truth are discussed in Section IV. After successfully implementing the algorithm, it is then tested, and the output is displayed graphically in Section V. Finally, the complete work is concluded in Section VI, and future work is suggested in Section VII.

II. INVESTIGATIONS IN AUTOMATIC VIDEO INDEXING

In a video, words have various meanings depending upon the context. As the saying goes, “sometimes, they are important, while others are just a recollection of a former viewpoint.” When looking at the appearance of an index point in a video, it is essential to understand its importance [4]. The literature review tries to uncover the critical characteristics to decide a topic change in a video lecture. A categorical literature survey is done based on the techniques intended to be performed during the research and tabulate the performance, relative merits, and limitations of existing approaches.

A. INDEXING OF VIDEOS

The repository of educational videos is immense, and it is impossible for a learner to copiously go through it and narrow down the content of interest. Rosalind Picard proposed Affective video indexing in 1990 while defining practical computing. As per the definition given by her, the need for indexing a video can be summarised as “Although affective annotations, like content annotation, will not be universal, they will still help reduce time searching for the right scene”. Automatic video indexing is indispensable to make a video interactive and autodidact. The purpose of indexing is to divide a lecture video into parts that reflect distinct
sub-topics. A successful index for a video lecture can only be created when the contents are automatically analyzed to extract relevant metadata. The first step is to identify all the timestamps in the video when the scene changes considerably. This will allow for segment-wise browsing of the video and content analysis explicitly targeted to the category. Furthermore, a subset of these locations is chosen as index points, representing the start of a sub-topic, which subsequently serves as a part of the index for the video. The specific problem this research will cater is to extract the keywords from the video frames based on which a lecture can be segmented. Each such keyword is called an index point, which will also contain the timestamp of the relative video time instance.

1) Existing Approaches for Automatic Video Indexing

Many researchers have developed frameworks and systems for video segmentation either for information retrieval or for content-wise indexing. Mottaleb et al. [5] suggested one of the oldest pieces of literature in this field that use a combination of content analysis and physical property data to pick the shots shown to users. Riedl et al. [6] proposes a novel algorithm 'TopicTiling' which is based on the standard 'TextTiling' algorithm for text segmentation where blocks of text are compared via bag-of-words vectors. Their work proves to be significant in terms of complexity and computationally less expensive than other LDA-based segmentation methods. F. Sauli [7] researched hyper videos that introduce hyperlinks within a video. The work focuses on the educational domain and provides clickable access to video components like pictures, web pages, and text. Biswas et al. [8] developed a rank model MmtoC for essential keywords (salient words). Both visual slides and voice transcripts were utilized to create the sentence. Cost functions are derived depending on the terms used in a ranking. The topic-oriented indexing of the movie is generated using the programming optimization method, which accelerates processing. An analysis of the textual content is conducted, and lecture video segmentation is based on the linguistic characteristics of the text. Lin et al. [9] convert lecture videos into text analyses of the resulting text to find dissimilarities and find ways to improve them. To distinguish context-dependent information from content-based information is one of the main challenges for video indexing researchers. Uke and Thool [10] aims to provide a digital index to education videos by converting videos into images and extracting the text directly through the Google Tesseract OCR Engine. Merler et al. [11] is one step ahead. The concept of both types of research is the same, unlike the latter extracts and recognizes the text directly from the video rather than converting the whole video into images. Adcock et al. [12] worked on lecture webcasts and tried to provide a searchable text index so that users could access local material within a video. The technique focuses on videos with presentation slides only and performs keyframe extraction using associativity of the textual content. Hui et al. [13] Proposed a new way to improve the interpretability and manipulability of a video by combined keyframe extraction and object-based video segmentation method. The similarity and redundancy between the frames is removed through the Kullback Leibler distance (AIKLDA) criterion.

There could be hundreds and thousands of videos on a single topic — all of them with different points of view and numerous instances of more profound insight. This sub-section claims that manual annotation or topic-wise indexing is not new in videos; it is sometimes time-consuming due to the length of a typical video lecture. Automatic indexing and annotation is, therefore, a better and efficient solution.

2) Automatic Video Indexing using Deep Learning

Deep Learning for automatic video indexing opens a whole new dimensions. Table 1 summarizes the review of research articles and their proposed idea for deep learning techniques, which are already used for video indexing. A method of automated indexing of videos using a text-based query is suggested by [20]. The users are asked to provide the keywords as a query; the framework returns a list of videos. Jothilakshmi [21] developed a novel method for content identification of news scenes in the NEWS Dataset and suggested it for automated content recognition systems. When making a transcription, this method comes with limitations. TV news sequences can only be processed using a commercially available programme, classifying them into one of the six predetermined categories (National Politics, National News, World, Finance, Society & Culture and Sports). Zhou et al. [22] proposes a technique proposed that includes automated video text panoramas to capture natural scene text information and picture stitching. It has been observed that [23] has obtained a very substantial 95.3% Recall and 98.6% Precision using CNN with Cosine Similarity. Lu et al. [24] indicate that deep convolutional neural networks combined with OCR Tools are also used nowadays to detect and recognize video text.

Through the extensive literature survey, a conclusion can be drawn that there is a need for automatic video indexing in educational videos. To reduce the computational complexity, shot boundary detection and keyframe identification techniques are also indispensable. Even though much work is done in video analysis and segmentation, Deep Learning for such cases is still restrictive.

B. A REVIEW OF VIDEO STRUCTURE ANALYSIS AND SHOT BOUNDARY DETECTION TECHNIQUES

During the past decade, shot boundary detection techniques have been a sphere of influence on researchers worldwide working in video analysis and Image processing. A large portion of the research community has been devoted to shot boundary detection using edges, color, motion cues, object correlation, singly, or a combination. Numerous techniques have been developed, and several comprehensive
TABLE1: Comparison of various Deep Learning Models in Video Analysis

| S. No. | Paper Title                                           | Author & Year | Deep Learning Model                                                                 | Conclusion                                                                                     |
|--------|-------------------------------------------------------|---------------|-------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------|
| 1.     | An Innovative Video Searching Approach using Video Indexing | [14] 2021     | Recurrent neural network (RNN), ResNet-152 Bi-LSTM (Bidirectional LSTMs)              | Video searching through indexing is proposed. VIST-Story Telling dataset is used to generate    |
|        |                                                       |               |                                                                                      | Video captions for preparing indices. RNN model is used for audio to text conversion and speech |
|        |                                                       |               |                                                                                      | recognition. OCR technique is used for Text extraction from pre-processed frames.                |
| 2.     | Automatic Image and Video Caption Generation With Deep Learning: A Concise Review and Algorithmic Overlap | [15] 2020     | CNN +RNN/LSTM, GAN, CNN+LSTM and LSTM+GAN                                           | A comprehensive review is presented for both image captioning and video captioning methodologies  |
|        |                                                       |               |                                                                                      | based on deep learning.                                                                          |
| 3.     | A Browsing and Retrieval System for Broadcast Videos using Scene Detection and Automatic Annotation | [16] 2019     | Convolutional Network (CNN)                                                           | Video browsing and retrieval system comprises of scene Detection module, a concept detection     |
|        |                                                       |               |                                                                                      | algorithm and a retrieval algorithm with which users can search for scenes inside a video        |
|        |                                                       |               |                                                                                      | collection. Siamese Neural network, a CNN based network whose penultimate layer is concatenated   |
|        |                                                       |               |                                                                                      | with features extracted from the transcript of the video.                                       |
| 4.     | Deep Learning Based Semantic Video Indexing and Retrieval | [17] 2018     | GoogLeNet network structure, Convolutional Neural Network (CNN)                      | Content based Video retrieval system is proposed in which features are extracted by CNN model.   |
| 5.     | Beyond Short Snippets: Deep Networks for Video Classification | [18] 2015     | Convolutional Network (CNN)                                                          | Proposes a CNN architectures for obtaining global                                                   |
|        |                                                       |               | Recursive neural network derived from Long Short Term Memory (LSTM)                   | video-level descriptors and a framework for video and action classification tasks.                |
| 6.     | Large-scale Video Classification with Convolutional Neural Networks | [19] 2014     | Convolutional Neural Network (CNN)                                                  | CNN is used for obtaining global video-level descriptors, video classification and retrieval       |

surveys have been presented to summarise them. It can be observed that although almost all the approaches detect Cut transitions efficiently, some recent approaches focus on detecting Gradual transitions.

Table 2 compares the existing research approaches that are efficient in detecting either Gradual Transition, Cut transition, or both. Although almost all the approaches detect Cut transitions efficiently, some recent approaches focus on detecting Gradual transitions. Several methods have been implemented to detect shot boundaries, producing highly accurate and acceptable results. We can infer from the literature that Color histograms are the most used global features for video shot boundary detection. Histograms provide a good trade-off between accuracy and computational time. Color Histograms (CBH) uses RGB space for the computation while Hue Saturation Value (HSV) Histograms are computed by HSV Color Space [43]. The hue, saturation, and value (HSV) color space are more intuitive and alternate for the RGB color space. It also uses three dimensions to describe a color, which produces sturdy detection results [44]. Tuna et al. [45] discusses a detailed literature survey of Shot boundary detection techniques that detects Abrupt Transitions and Gradual Transitions. Ma et al. [46] detected gradual and abrupt transitions using a dual detection model by performing pre-detection. An uneven blocked mechanism is used based on the human visual system, histogram, and pixel difference for pre-detection. The false detection was removed in the detection phase by employing the Scale Invariant Feature Transform (SIFT).

Amiri and Fathy [47] developed a noise-robust algorithm using Generalized Eigenvalue Decomposition (GED) and obtained a distance function. Cut transitions were realized when abrupt changes were noticed, and gradual transitions

| Year | Reference | CT Detection | GT Detection |
|------|-----------|--------------|--------------|
| 2021 | [25]      | Y            | N            |
| 2021 | [26]      | Y            | Y            |
| 2021 | [27]      | Y            | Y            |
| 2020 | [28]      | N            | Y            |
| 2020 | [29]      | Y            | Y            |
| 2019 | [30]      | Y            | Y            |
| 2018 | [31]      | Y            | Y            |
| 2018 | [32]      | Y            | Y            |
| 2017 | [33]      | Y            | Y            |
| 2017 | [34]      | Y            | Y            |
| 2017 | [35]      | Y            | Y            |
| 2016 | [36]      | Y            | Y            |
| 2016 | [37]      | Y            | N            |
| 2016 | [38]      | Y            | Y            |
| 2016 | [39]      | Y            | Y            |
| 2015 | [40]      | Y            | Y            |
| 2013 | [41]      | Y            | Y            |
| 2009 | [42]      | Y            | Y            |
were obtained when semi-Gaussian behaviour was reflected in the distance function. Multilevel Difference of Color Histograms and the voting mechanism [38], and [48] used Convolution Neural Networks (CNN) to determine cut and gradual transitions. Hue Saturation Value is used for detecting abrupt shots, and a 3-dimensional convolution layer in CNN is used to obtain gradual transitions. Furthermore, CNN avoids the disturbance caused by abrupt transitions [30]. In [35], the Frobenius norm with double threshold and Singular Value Detection updating was used for detecting Cut and Global Transitions, respectively.

A Modified Artificial Bee Colony algorithm is used to identify candidate boundaries. A hybrid of Fast Accelerated Segment Test and fuzzy histograms is further used to extract local and global features to verify the obtained boundaries [49]. Symmetric Local Binary Pattern with histogram features was implemented on six TV series to handle illuminating changes and detect hard cuts [2]. Temporal features were extracted from movies and trailers, followed by Dynamic Mode Decomposition to extract temporal background and foreground modes to detect hard cuts, Fade, and dissolve [5]. The high-Level Fuzzy Petri Net model with keypoint matching is used on Commercial videos, movies, TV shows, and dramas to detect gradual transition by removing false shots [32]. Audio and optical features were extracted from talk shows obtained using Daily Motion and YouTube websites using clustering-based algorithms to show that the number of frames is limited to two while the actor movement is more [50]. Candidate segments are obtained by comparing Oriented FAST and Rotated BRIEF (ORB) features, Cut Transitions are extracted by comparing Structural Similarity, and Gradual Transitions are detected in the gradual transition model in 106049 test frames from Open-video project, YouTube and YOUKU [51].

### III. TECHNICAL PERSPECTIVE OF THE RESEARCH

Semantic characteristics can now be extracted from pictures and video sequences. There are still problems with video lectures because of their lengthy running duration, non-uniform material, and complex narratives. Automatic video indexing is a technique that presents a potential solution to these problems. This research is carried out in 3-phases, each of which has its significance and contribution towards better accuracy and less computational complexity as depicted in Figure 1.

1) **Phase-I: Frame Extraction and Pre-processing:** The important steps in this phase are identification of appropriate dataset for the research, extraction of frames and embedding of timestamp with frames.

2) **Phase-II: Candidate Segment Extraction using Shot Boundary Detection:** The significant steps in this phase are an amalgamation of Binary Search Pattern Algorithm with Structural Similarity Index and then the comparison of the results with the state-of-the-art algorithm SSIM.

3) **Phase-III: Automatic Index Point Detection:** This phase incorporates training of YoloV4 neural Network over 6000 image frames for heading detection. The model is then tested upon a tailor-made suit of 50 videos. Finally, the results are expressed in Precision, Recall, and F1 Score using the Mean Opinion Score.

When a video is split into frames, numerous boundary and non-boundary frames are generated. The number of non-boundary frames is comparatively higher than that of boundary frames. Since the non-boundary frames are redundant, we need to eliminate them to improve the computational complexity. For this purpose, Shot Boundary Detection is used. When Shot Boundaries are successfully identified, there is a single keyframe that represents a shot. We call that keyframe as ‘Candidate Segment’. We can conclude that in a non-boundary segment, the first frame and the last frame are highly correlated [52].

The significant phases in the proposed work are gathering custom data set, shot boundary detection and candidate segment identification, customizing configuration files necessary to train the Neural Network Model, Validating training data for accuracy and finally testing the custom object detector to see the model’s accuracy in real-time.

### A. FRAMES EXTRACTION AND PROCESSING

This study is based on a dataset of video lectures, and the model is trained on videos ranging in length from 30-45 minutes, for a total run time of 22 hours. Lecture videos often feature a frame rate of 30, which means that 30 frames are produced every second [53]. As a result, the average number of frames per video we have worked on is 35,000 - 40,000 frames, with just 25-30 boundary frames. Therefore, Frame reduction is a significant challenge: the fewer the frames, the faster and more effective the video processing is. Henceforth, initially, to reduce the number of frames, We assumed that since a topic will be debated for at least 10 seconds in a video lecture, a delay of the same can be added in frame generation, and subsequent redundant frames can be reduced. For example, the total amount of frames initially produced in our dataset video entitled “What is JWT JSON” was 38,680. The frames are reduced to 19,015 after a 10-second delay, which is almost half the number.

Another critical aspect of frame generation is keeping track of the frame’s arrival time w.r.t. video playback. This frame timestamp indicates the exact point in time when that frame appears in the video. Since the purpose of the research is to provide automatic video indexing, timestamps plays a very vital role in video browsing and navigation.

### B. BINARY SEARCH PATTERN ALGORITHM (BSPA)

Binary Search Pattern Algorithm (BSPA) also known as **half-interval algorithm**. It is a searching algorithm that searches for a targeted image frame in a sorted array of frames. In our case, the frames are sorted based on their index...
Algorithm 1 Binary Search Pattern Algorithm

1: procedure BSPA(left, right)
2:    A ← Array_of_frames
3:    n ← Size_of_array
4:    x ← Boundary_frame
5:    Set left = 0
6:    Set right = n-1
7:    Compare ssim_left and ssim_right
8:   if ssim_left ≠ ssim_right then Set ssim_mid = (ssim_left + ssim_right) / 2
9:    else ssim_pivot = ssim_mid
10:   Compare ssim_pivot and ssim_right
11:  if ssim_pivot = ssim_right then boundary_frame = ssim_right
12: End Procedure

number. BSPA is required to reduce the computational time from O(n) to the logarithmic time, O(log n), where n is the number of elements in the array. Algorithm 1 represents how BSPA is used to divide the array of n frames and select the pivot frame.

C. STRUCTURAL SIMILARITY INDEX
The similarity is a computed value between two images that determines the pixel or visually similar the images are. There are various methods to compute the similarity between two images like Template Matching, Image Descriptors (such as SIFT, SURF and FAST) and Structural Similarity Index Measure (SSIM).

We have worked on the SSIM approach in this experiment because of its easy implementation with satisfactory results. SSIM is a metric that looks for the similarity between the pixels of two images. If the pixel density is remarkably similar, SSIM will return $\approx 1$, and if it is vastly different, SSIM will return $\approx -1$. The assessment index of quality is based on the computation of three expressions: Luminance expression, Contrast expression and Structural expression as per below equation 1:

$$SSIM(a, b) = \left[ l(a, b) \right]^{\alpha} \cdot [c(a, b)]^\beta \cdot [s(a, b)]^\gamma$$  (1)

Here, $l$ stands for Luminance, $c$ stands for the Contrast and $s$ stands for the Structural expressions respectively. The expressions for Luminance, Contrast and Structure can be observed in equations 2, 3 and 4.

$$l(a, b) = \left( \frac{2\mu_a\mu_b + z_1}{\mu_a^2 + \mu_b^2 + z_1} \right)$$  (2)

$$c(a, b) = \left( \frac{2\sigma_a\sigma_b + z_2}{\sigma_a^2 + \sigma_b^2 + z_2} \right)$$  (3)

$$s(a, b) = \left( \frac{\sigma_{ab} + z_3}{\sigma_a + \sigma_b + z_3} \right)$$  (4)

In the equations above $\mu_a, \mu_b$ shows the local means, $\sigma_a, \sigma_b$ shows the standard deviations, $\sigma_{ab}$ shows the cross-covariance for the images and $z_1, z_2, z_3$ are SSIM constants.
Here, the assumption is that $\alpha = \beta = \gamma = 1$ and $z_1 = \frac{z}{2}$, then the above equations simplifies to equation 5:

$$SSIM(a,b) = \left(\frac{2\mu_a\mu_b + z_1}{\mu_a^2 + \mu_b^2 + z_1}\right) \left(\frac{2\sigma_{ab} + z_2}{\sigma_a^2 + \sigma_b^2 + z_2}\right)$$

(5)

Technically, SSIM is based on three utility functions that perform all essential tasks.

- *gaussian Function* that generates an array of numbers sampled from a gaussian distribution. The length of the array is equal to the size of the window. Sigma $\sigma$ is used to represent the standard deviation of the distribution.
- *create_window Function* is used to create a 2-dimensional array by multiplying the array (generated by gaussian Function) with its transpose.
- *ssim Function* is used to perform the mathematical calculations, including luminance, contrast and SSIM score based on standard formulae.

There is a plot twist while implementing SSIM, that for image quality assessment, it is useful to apply the SSIM index locally rather than globally. Therefore, we have used the Mean Index Value for Structural Similarity.

Algorithm 2 represent the state-of-the-art SSIM approach implementation. It can be seen in algorithm 2 that the similarity score between two image frames is kept to be 0.42 (i.e. 42% similarity for non-adjacent frames), which is the mean similarity score for the SSIM approach.

**Algorithm 2 SSIM Threshold**

1. if the subtracted value of left and right is greater than five
2. return Algo(left, mid)
3. else right value is greater than left+1
4. if SSIM score is less than 0.42 then, return Algo(left, mid) return Algo(mid, right)

**end function**

### D. CANDIDATE SEGMENT IDENTIFICATION USING STRUCTURAL SIMILARITY AND BINARY SEARCH PATTERN

Identifying correct and accurate shot boundaries is the backbone for successful video segmentation and content-based video retrieval. The maximum part of this research revolves around the optimizing technique to identify abrupt transitions for shot boundary detection based on Structural Similarity (SSIM) fused with Binary Search Pattern Algorithm (BSPA). The algorithm for the same is represented in Algorithm 3. We tried to make the overall solution less costly and computationally efficient.

**Algorithm 3 Proposed Algorithm for Candidate Segment identification**

1. Score = Calculate SSIM(Left,Right)
2. mid = Calculate mid (Left,Right)
3. L_Score = Calculate SSIM(Left,mid)
4. R_Score = Calculate SSIM(mid,Right)
5. mod_L = mod(L_Score - Score)
6. mod_R = mod(R_Score - Score)
7. mod_LR = mod(L_Score - R_Score)
8. Sim,L_Sim,R_Sim = False
9. if mod_L & mod_R & mod_LR then L_Sim = True
10. if mod_L or mod_R or mod_LR then R_Sim = True
11. if L_Sim&R_Sim then Sim = True return Index

The proposed approach and the state-of-the-art SSIM Technique are examined for shot boundary detection on python 3.7.1 with a 1.80 GHz CPU and 8 GB RAM with 5900 image frames to gauge the effectiveness. The results quantified on the basis of precision, recall, and F1 score. The comparative results show that the proposed algorithm successfully detects abrupt transitions with reasonable accuracy in detection and significantly decreases the complexity of the algorithm.

The input to the system is a video sequence that is converted to image frames. An index number $0^{th}$ to $(N-1)^{th}$ is assigned to each frame, where $N$ is the total number of frames. A 5-frame window is created to check the structural similarity of the two frames.

The algorithm checks if the frames in the window are less than five or more. If the number is less than five, it is evident that the last frame is the boundary frame, and hence the algorithm returns the last frame; otherwise, the window is moved forward to the next set of 5-frames.

The location of the frames that have to be compared is divided into left, mid and right frames. The $0^{th}$ and $(N-1)^{th}$ frames are compared by the standard SSIM approach in the first iteration. If they are found similar, we can instantly conclude that the video contains only a single shot. However, this case is only possible for videos with few seconds of running time, and those videos are not targeted in the proposed research. Else wise, when the $0^{th}$ and $(N-1)^{th}$ are dissimilar, the algorithm will divide the frames in half. The execution of the 5-frame window will be as follows:

- Checks the SSIM score of the first and last frame inside the 5-frame window.
- If the SSIM score $\leq$ 0.42, then the frames do not belong to the same shot, and therefore ALGO(left, mid) is executed.
- Otherwise, it can be concluded that since the SSIM score $\geq$ 0.42, the frames are adjacent, i.e. belongs to the same shot.
- Move the 5-frame window to the next five frames by executing Algo(mid, right).
- Return the left frame as boundary frame.
• If not, then the algorithm checks if it is the last frame. If yes, it returns right as the boundary frame.

Table 3 shows the comparison between the state-of-the-art SSIM and proposed approach for abrupt shot boundary detection. The results are compared based on the average percentage of precision, recall and F1-Score as depicted through Figure 2, Figure 3 and Figure 4 respectively.

The comparative analysis of existing state-of-the-art techniques is compared with the proposed method, as displayed in Table 4. The only limitation of our proposed approach is that if a shot is repeated, the proposed method considers it as a new shot and fetch its boundary frame, so the boundary frames might be similar, hence increasing the redundancy.

The computational time for detecting Shot Boundary through the proposed approach improves significantly compared to the traditional SSIM Technique. The average time by proposed approach is 129 seconds, which is 21.4% lesser than the basic approach. We can conclude that the proposed algorithm is 20-25 times faster. The average precision is 83% which is the same for both approaches, but there is a significant change in average recall(97%) and average F1-Score (89%), respectively.

The results demonstrate that integrating Binary Search in the standard SSIM Algorithm gradually improves detection performance and reduces the computational time from $O(n)$ to the logarithmic time $O(\log n)$.

**E. INDEX POINT DETECTION USING YOLOV4**

After successfully identifying Candidate Segments, the next important task is to extract the index points from those image frames. YoloV4 Darknet custom object detector is used for automatic extraction of index points. Darknet is a robust open access neural network framework geared towards object detection, using CPU and GPU computation. The unique architecture and high speed distinct the Darknet Framework from other existing NN architectures. A specialized framework such as YOLO is found in the Darknet. YOLO (an acronym for “You Only Look Once”) can run on CPU, but it gains up to 500 times more speed on GPUs because it uses CUDA and cuDNN. Another positive aspect of Darknet is that it is comparatively convenient to train compared to the other heavily optimized and popular frameworks in customized data sets.

Through this research, we concluded that the method of automatic keyword extraction using Deep Neural Network gives satisfactory results in terms of speed and accuracy.

1) **Configuration**

This technique uses a 137-layer YoloV4 custom object detector neural network model for identifying the headings/keywords in the image frames. The neural network model is trained on approximately 6000 video frames and then tested on a suit of 50 educational videos of around 20 hours of run time. The comparative results viz. detection accuracy, precision, recall, and F1 score show that the proposed algorithm successfully generates a digital index with reasonable accuracy in topic detection.

The technical configuration of YoloV4 is mostly summarized.
by 4-python files as in figure 1 viz.:

1) Detection.py
   Detection.py uses SSIM, OpenCV as imports and gets the unique Frames from SSIM.py and gets the title frames from the YOLOV4 model, which is trained specifically for this purpose. Then, after detection, it yields the data to UI.py in the below format:
   
   'frame_no':frame_no,'x':x,'y':y,'x+w':x+w,'y+h':y+h

2) Reducer.py
   This python script gets the video, applies the SSIM to the video frames, and yields the frames to Detection.py when a unique frame is found.

3) UI Runner.py
   The file starts the UI—the program’s starting point (UI, Detection as Imports).

4) UI.py
   The file that initializes the UI Sets up VLC Media Player and uses the Extracted title frames from detection, gets the text image using OpenCV, applies SSIM like reducing redundant frames and converts them to text using PyTesseract and timestamps the unique remaining frames. (uses Detection, Pyside 2, python-VLC, SSIM as imports).

2) Heading Extraction
   The research states the application of Single Class YOLOV4 as a custom object detector through Google Colaboratory. Approximately 6000 video frames from different videos were included in our dataset. 90% of frames were used for training our model, and 10% frames were used for validation. Yolo performs unzipping and inflating all our training images and our validation images. Testing of the model is done explicitly on a suit of 50 different video lectures. A sample result of heading detection can be seen in figure 5.

IV. DATASET, HYPOTHESIS AND GROUND TRUTH
   Training and validation data comprises of extracted frames from video lectures of various domains. The frames are then manually labelled through the makesense.ai object detection toolkit. Since this is a single class object detector, the only heading label is shown in figure 6, and each training image will have a .txt file associated with it which contains the coordinates of the heading in the image.
\[ mAP = \left( \frac{\sum_{q=1}^{Q} \text{Ave} P(q)}{Q} \right) + \left( \frac{\sum_{q=1}^{Q} \text{Ave} R(q)}{Q} \right) + \left( \frac{\sum_{q=1}^{Q} \text{Ave} P(q)}{Q} \right) \] (6)

\[ mAR = \left( \frac{\sum_{q=1}^{Q} \text{Ave} R(q)}{Q} \right) + \left( \frac{\sum_{q=1}^{Q} \text{Ave} R(q)}{Q} \right) + \left( \frac{\sum_{q=1}^{Q} \text{Ave} P(q)}{Q} \right) \] (7)

\[ mAF1 = 2 \left( \frac{mAP \cdot mAR}{mAP + mAR} \right) \] (8)

The difference of precision, recall and F1 score for heading detection through the proposed approach and Mean Opinion Score given by the external volunteer can be seen through figure 7.

Several spurious frames are produced during detection, some false negatives are generated, and some unnecessary headings are generated due to PyTesseract’s efficiency. It can be observed from the figure that the range of Precision, Recall and F1 Score of the proposed approach lies between 65-70%.

V. RESULTS AND DISCUSSIONS

This paper analyzes the results in two categories. The first category results (as discussed in Subsection III-D) depicts the improvement in shot boundary detection when the state-of-the-art Structural Similarity Index Matrix(SSIM) algorithm is modified by fusing with a Binary search pattern algorithm. The results justify that there is a good enhancement in accuracy, computational complexity and average processing time.

The second category of results (as discussed in this Section V) demonstrates the performance of a custom YoloV4 object detection model for heading detection. In order to gauge the effectiveness, apart from calculating the confusion matrix, the accuracy of detected index points is also assessed based on the Mean Opinion Score. The model is trained on approximately 6000 video frames and then tested on a suite of 50 videos of around 20 hours of run time. We begin by determining the shot boundaries and keyframe using Algorithm 3. We then locate the cut transitions/heading changes by watching the videos and noting each shot’s beginning and end times.

The performance and throughput of the YoloV4 custom object detector during the training can be observed through figure 8 which shows average loss vs iterations. For a model to be ‘accurate’, we aim for a loss under 2, which is currently 1.83 for 8000 iterations.

The experimental setup is summarised in table 5. Using this information as a ground truth, we prepared the confusion matrix to calculate the Precision and Recall. The results show that the mean precision and mean F1 score is 70% as depicted in Figure 9, whereas there is a subsequent increase in recall with 76% as shown in Figure 10. This shows that the number of True Positives is better than False Negatives.

On the trained YoloV4 model, 81.24% of keyframe detection efficiency was obtained after the first test images and the Range of Precision, Recall, and F1 score varies between 60-75% which is considered to be reasonably good.

FIGURE7: Difference of precision, recall and F1 for heading detection through proposed approach and through Mean Opinion Score

FIGURE8: Average Loss vs Iterations.
TABLES: Evaluation of heading detection technique with respect to the videos. The results show a decrease in precision and increasing recall. The efficiency of keyframe detection, however, increases.

| S. No. | Video Name | True Positive (TP) | False Positive (FP) | False Negative (FN) | Precision | Recall | F1 |
|--------|------------|--------------------|---------------------|---------------------|-----------|--------|----|
| 1      | A Brief History of AI | 8 | 5 | 0 | 0.62 | 1.00 | 0.76 |
| 2      | A Brief Introduction of Micro-Sensors | 3 | 0 | 0 | 1.00 | 1.00 | 1.00 |
| 3      | Agriculture | 3 | 0 | 1 | 1.00 | 0.75 | 0.86 |
| 4      | Algorithm A. | 8 | 2 | 6 | 0.80 | 0.57 | 0.61 |
| 5      | ANOVA - I | 5 | 3 | 7 | 0.63 | 0.42 | 0.50 |
| 6      | Artificial Intelligence | 18 | 10 | 3 | 0.64 | 0.86 | 0.73 |
| 7      | Artificial Neural Networks | 8 | 3 | 4 | 0.73 | 0.67 | 0.70 |
| 8      | Biological Neural Network | 3 | 3 | 0 | 0.50 | 1.00 | 0.67 |
| 9      | Chi - Square Test of Independence - I | 2 | 3 | 4 | 0.40 | 0.33 | 0.36 |
| 10     | Cloud Bigtable as a NoSQL Option | 6 | 0 | 1 | 1.00 | 0.86 | 0.92 |
| 11     | Computer Integrated Manufacturing | 2 | 9 | 0 | 0.18 | 1.00 | 0.31 |
| 12     | Data frames | 9 | 2 | 2 | 0.82 | 0.82 | 0.82 |
| 13     | Deterministic Search | 5 | 2 | 2 | 0.71 | 0.71 | 0.71 |
| 14     | Environmental and Occupational Hazards | 1 | 1 | 5 | 0.50 | 0.17 | 0.25 |
| 15     | Introduction- Biosafety Aspects of GE Plants | 4 | 2 | 1 | 0.67 | 0.80 | 0.73 |
| 16     | Introduction to Mathematical Logic | 11 | 3 | 3 | 0.79 | 0.92 | 0.85 |
| 17     | Introduction to User-Centric Computing | 7 | 3 | 3 | 0.70 | 0.70 | 0.70 |
| 18     | Environmental Chemistry and Microbiology | 1 | 1 | 1 | 0.50 | 0.50 | 0.50 |
| 19     | Lec_8 - MIPS Pipeline for Multi-Cycle Operations | 5 | 2 | 0 | 0.71 | 1.00 | 0.83 |
| 20     | Lec_17 - Introduction to DRAM System | 6 | 2 | 6 | 0.75 | 0.50 | 0.60 |
| 21     | Lecture 2 - Introduction to Fuzzy Logic | 6 | 3 | 3 | 0.67 | 0.35 | 0.60 |
| 22     | Lecture 07 - Lexical Analysis | 8 | 1 | 2 | 0.63 | 0.80 | 0.73 |
| 23     | Lecture 07 - Projective Transformation | 11 | 4 | 3 | 0.73 | 0.79 | 0.76 |
| 24     | Lecture 7 | 11 | 5 | 2 | 0.69 | 0.85 | 0.76 |
| 25     | Lecture 9 - Introducing Functions | 6 | 6 | 2 | 0.50 | 0.75 | 0.60 |
| 26     | Lecture 10 - Misra-Gries sketch | 3 | 9 | 5 | 0.25 | 0.38 | 0.30 |
| 27     | Lecture 17 | 6 | 4 | 1 | 0.67 | 0.86 | 0.75 |
| 28     | Linear Regression vs Logistic Regression | 8 | 3 | 4 | 0.61 | 1.00 | 0.80 |
| 29     | Optical Fibers | 5 | 1 | 1 | 0.83 | 0.83 | 0.83 |
| 30     | Philosophical Foundations of Social Research | 2 | 2 | 0 | 0.50 | 1.00 | 0.67 |
| 31     | Phonetics and Phonology | 1 | 0 | 0 | 1.00 | 1.00 | 1.00 |
| 32     | Probability and NLP | 5 | 2 | 0 | 0.71 | 1.00 | 0.83 |
| 33     | Probability Distributions- Gaussian, Bernoulli | 7 | 1 | 0 | 1.00 | 0.88 | 0.93 |
| 34     | SAI Problem | 4 | 6 | 0 | 0.40 | 1.00 | 0.57 |
| 35     | SPM_Intro | 4 | 2 | 5 | 0.67 | 0.44 | 0.53 |
| 36     | Statistical Properties of Words - Part 01 | 3 | 3 | 8 | 0.50 | 0.27 | 0.35 |
| 37     | Tesseract | 8 | 2 | 3 | 0.80 | 0.73 | 0.76 |
| 38     | What Is Project Management | 25 | 9 | 4 | 0.72 | 0.85 | 0.78 |
| 39     | What are microservices | 8 | 1 | 3 | 0.89 | 0.73 | 0.80 |
| 40     | What Is Jenkins | 11 | 8 | 0 | 0.58 | 1.00 | 0.73 |
| 41     | What is Machine Learning | 23 | 5 | 1 | 0.82 | 0.96 | 0.88 |
| 42     | What is Scrum | 5 | 3 | 3 | 0.63 | 0.63 | 0.63 |
| 43     | Introduction to DevOps | 11 | 6 | 1 | 0.65 | 0.92 | 0.76 |
| 44     | Network Security Tutorial | 33 | 13 | 7 | 0.72 | 0.83 | 0.77 |
| 45     | NOOP vs REST | 5 | 0 | 2 | 1.00 | 0.71 | 0.83 |
| 46     | Introduction to AWS Services | 7 | 0 | 4 | 1.00 | 0.64 | 0.78 |
| 47     | POP3 vs IMAP | 3 | 0 | 2 | 1.00 | 0.60 | 0.75 |
| 48     | Kubernetes for Beginners | 8 | 1 | 1 | 0.89 | 0.89 | 0.89 |
| 49     | Need of IoT and Application | 9 | 0 | 2 | 1.00 | 0.82 | 0.90 |
| 50     | What is JWT JSON | 3 | 4 | 3 | 0.43 | 0.50 | 0.46 |
|        | Average Precision, Recall and F1 Score | | | | 0.70 | 0.76 | 0.70 |

FIGURE9: Precision representation for proposed approach

VI. CONCLUSION

The study establishes a basis for indexing digital video through the YOLO V4 Darknet Neural Network. Detection
accuracy, Precision, Recall, and F1 Score are used to reflect overall performance and outcomes. The threshold values are based on experimental observations of multiple videos. As this is a new approach, much work can be done in the future. For instance, an adaptive threshold value can be used instead, and the performance is still abysmal in the case of handwritten text in the videos. The framework used a basic binary search algorithm infused with Structural Similarity Index. The dataset used for training the neural network is made from 6000 video frames, and for testing, a suite of 50 videos was randomly selected.

It is important to note that the manual annotation used for training has limitations. For example, during the random data collection to create a training dataset, one cannot always mention the annotators’ context, accuracy, and qualification. However, to confine the scope of the study to a specific domain, the experiments were structured to include only participants with an educational context and to cover videos from educational portals. Furthermore, manual indexing is only used for evaluation and not for the generation of index points. As a result, our subjective assessment of the proposed algorithm using three different types of experiments is sufficient to determine its efficacy.

VII. FUTURE ASPECTS OF THE RESEARCH

The present study tackled some aspects; nevertheless, more in-depth optimization and benchmarking are necessary to study the effect on the efficiency of index point detection for standardization and relevant summaries. In the future, we shall explore the applicability of this technique and any potentially needed extensions. This research discovered that the frequency of words, n-grams, and the number of first-time words that appeared in a video provide valuable information for video segmentation by topic. It is therefore expected to do a long-term, real-world effect analysis to evaluate keywords in instructional videos. The research findings can also be improved by synchronizing speech with textual data. The spoken words can be extracted using speech transcripts and based on the context and semantics of the speech, and they can be associated with image text.

REFERENCES

[1] G. D. Abowd, “Classroom 2000: An experiment with the instrumentation of a living educational environment,” IBM systems journal, vol. 38, no. 4, pp. 508–530, 1999.
[2] T. Kar and P. Kanungo, “A novel method of shot boundary detection using center symmetric local binary pattern,” International journal of engineering research and technology, vol. 4, 2018.
[3] A. Coombe, “Global education census report,” Cambridge Assessment International Education, 2018.
[4] R. Lienhart, “Automatic text segmentation and text recognition for video indexing,” Multimedia Systems, Jan 2000.
[5] M. Abdel-Mottaleb, N. Dimitrova, R. Desai, and J. Martino, “Conivas: Content-based image and video access system,” in Proceedings of the Fourth ACM International Conference on Multimedia, MULTIMEDIA ’96, (New York, NY, USA), p. 427–428, Association for Computing Machinery, 1997.
[6] M. Riedl and C. Biermann, “TopicTiling: A text segmentation algorithm based on LDA,” in Proceedings of ACL 2012 Student Research Workshop, (Jeju Island, Korea), pp. 37–42, Association for Computational Linguistics, July 2012.
[7] F. Sauli, A. Cattaneo, and H. van der Meij, “Hypervideo for educational purposes: a literature review on a multifaceted technological tool,” Technology, Pedagogy and Education, vol. 27, no. 1, pp. 115–134, 2018.
[8] A. Biswas, A. Gandhi, and O. Deshmukh, “Mntoc: A multimodal method for table of content creation in educational videos,” in Proceedings of the 23rd ACM International Conference on Multimedia, MMA ’15, (New York, NY, USA), p. 621–630, Association for Computing Machinery, 2015.
[9] M. Lin, J. Junamaker, M. Chau, and H. Chen, “Segmentation of lecture videos based on text, audio and combining multiple linguistic features,” in 37th Annual Hawaii International Conference on System Sciences, 2004. Proceedings of the, pp. 9 pp., 2004.
[10] N. J. Uke and R. Thool, “Segmentation and organization of lecture video based on visual contents,” International Journal of e-Education, e-Business, e-Management and e-Learning, 2012.
[11] M. Merler and J. R. Kender, “Semantic keyword extraction via adaptive text binarization of unstructured unsourced video,” in 2009 16th IEEE International Conference on Image Processing (ICIP), pp. 261–264, 2009.
[12] J. Adcock, M. Cooper, L. Denoue, H. Pirsiavash, and L. A. Rowe, “Talkminer: A lecture webcast search engine,” in Proceedings of the 18th ACM International Conference on Multimedia, MM ’10, (New York, NY, USA), p. 241–250, Association for Computing Machinery, 2010.
[13] C. Hui, S. Yunyu, Y. Haisheng, G. Ming, and Y. Liu Xiang, X. “A fast and robust key frame extraction method for video copyright protection,” Journal of Electrical and Computer Engineering, March 2017.
[14] J. J. S. I. V. P.Devassia, “An innovative video searching approach using video indexing,” International Journal of ComputerScience & Network, vol. 8, pp. 144–147, 2021.
[15] S. Amirian, K. Rasheed, T. R. Taha, and H. R. Arabnia, “Automatic image and video caption generation with deep learning: A concise review and algorithmic overlap,” IEEE Access, vol. 8, pp. 218386–218400, 2020.
[16] S. Jones, “Automatic video indexing and transferred deep convolutional neural network,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–17, 2019.
[17] A. Podlesnya and S. Podlesnya, “Deep learning based semantic video indexing and retrieval,” in Proceedings of SAI Intelligent Systems Conference (IntelliSys) 2016, pp. 359–372, Springer International Publishing, 2018.
[18] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, “Beyond short snippets: Deep networks for video classification,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2015.
[19] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, “Large-scale video classification with convolutional neural networks,” in 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1725–1732, 2014.
[20] V. K. Kamabathula and S. Iyer, “Automated tagging to enable fine-grained browsing of lecture videos,” in 2011 IEEE International Conference on Technology for Education, pp. 96–102, 2011.
[21] S. Jothilakshmi, “Spoken keyword detection using autoassociative neural networks,” International Journal of Speech Technology, vol. 17, 2014.
[22] T. Zhou, K. Wang, J. Wu, and R. Li, “Video text processing method based on image stitching,” in 2019 IEEE 4th International Conference on Image, Vision and Computing (ICIVC), pp. 561–566, 2019.
[23] L. Zhang and Y. Lu, “Video object segmentation by latent outcome algorithmic overlap,” IEEE Access, vol. 8, pp. 30355–30367, 2020.
[24] W. Lu, H. Sun, J. Chu, X. Huang, and J. Yu, “A novel approach for video text detection and recognition based on a corner response feature map and transferred deep convolutional neural network,” IEEE Access, vol. 6, pp. 40198–40211, 2018.
[25] T. D. M. Chakraborty Saptarshi, Singh Alok, “A novel bifold-stage shot boundary detection algorithm: invariant to motion and illumination,” The Visual Computer, February 2021.
[26] N. H. S. Rashmi B. S., “Video shot boundary detection using block based cumulative approach,” Multimedia Tools and Applications, vol. 80, January 2021.
[27] R. Mishra, “Video shot boundary detection using hybrid dual tree complex wavelet transform with Walsh hadamard transform,” Multimedia Tools and Applications, vol. 80, January 2021.
[28] F-F. Duan and F. Meng, “Video shot boundary detection based on feature fusion and clustering technique,” IEEE Access, vol. 8, pp. 214633–214645, 2020.
A. B. Colony, “Video shot boundary detection method using modified artificial bee colony and fast accelerated segment test algorithm,” International Journal of Engineering & Technology, vol. 8, no. 1.5, pp. 177–182, 2019.

D. Brunner, “Frame rate: A beginner’s guide,” Tech Smith, January 2021.

R. Shen, Y. Lin, T. T. Juang, V. R. L. Shen, and S. Y. Lim, “Automatic detection of video shot boundary in social media using a hybrid approach of bfpn and keypoint matching,” IEEE Transactions on Computational Social Systems, vol. 5, no. 1, pp. 210–219, 2018.

C. Bi, Y. Yuan, J. Zhang, Y. Shi, X. Yang, W. Wang, and R. Zhang, “Dynamic mode decomposition based video shot detection,” IEEE Access, vol. 6, pp. 21397–21407, 2018.

S. Tippaya, S. Sitjongtsaporn, T. Tan, M. M. Khan, and K. Channongthai, “Multi-modal visual features-based video shot boundary detection,” IEEE Access, vol. 5, pp. 12563–12575, 2017.

B. Youssif, E. Fedwa, A. Driss, and S. Ahmed, “Shot boundary detection via adaptive low rank and svd-updating,” Computer Vision and Image Understanding, vol. 161, pp. 20–28, 2017.

R. Hanneke, A. Elboushaki, K. Afdel, P. Nagabushan, and M. Javed, “An efficient method for video shot boundary detection and keyframe extraction using sift-point distribution histogram,” International Journal of Multimedia Information Retrieval, vol. 5, pp. 89–104, 2016.

B. H. Shekar, K. P. Uma, and K. R. Holla, “Shot boundary detection using correlation based residual saliency map,” in 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 2242–2247, 2016.

Z. Li, X. Liu, and S. Zhang, “Shot boundary detection based on multilevel difference of colour histograms,” in 2016 First International Conference on Multimedia and Image Processing (ICMIP), pp. 15–22, 2016.

J. Xu, L. Song, and R. Xie, “Shot boundary detection using convolutional neural networks,” 2016 Visual Communications and Image Processing (VCIP), pp. 1–4, 2016.

F. Liu and Y. Wan, “Improving the video shot boundary detection using the hsv color space and image subsampling,” in 2015 Seventh International Conference on Advanced Computational Intelligence (ICACI), pp. 351–354, IEEE, 2015.

T. Vo, D. Tran, W. Ma, and K. Nguyen, “Improved HOG descriptors in image classification with CP decomposition,” in Neural Information Processing - 20th International Conference, ICONIP 2013, Daegu, Korea, November 3-7, 2013. Proceedings, Part III (M. Lee, A. Hirose, Z. Hou, and R. M. Kil, eds.), vol. 8228 of Lecture Notes in Computer Science, pp. 384–391, Springer, 2013.

J. Ren, J. Jiang, and J. Chen, “Shot boundary detection in mpeg videos using local and global indicators,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 19, no. 8, pp. 1234–1238, 2009.

Y. Yang, Z.-J. Zha, Y. Gas, X. Zhu, and T.-S. Chua, “Exploiting web images for semantic video indexing via robust sample-specific loss,” IEEE Transactions on Multimedia, vol. 16, no. 6, pp. 1677–1689, 2014.

B. C. O’Connor and R. L. Anderson, Video Structure Meaning. Morgan & Claypool Publishers, 2019.

T. Tuna, M. Joshi, V. Varghese, R. Deshpande, J. Subblok, and R. Verma, “Topic based segmentation of classroom videos,” in 2015 IEEE Frontiers in Education Conference (FIE), pp. 1–9, 2015.

M. Ma, S. Met, J. Hou, S. Wan, and Z. Wang, “Video summarization via temporal collaborative representation of adjacent frames,” in 2017 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS), pp. 164–169, IEEE, 2017.

A. Amiri and M. Fathy, “Video shot boundary detection using generalized eigenvalue decomposition and gaussian transition detection,” Computing and informatics, vol. 30, no. 3, pp. 595–619, 2012.

Z. Li, X. Liu, and S. Zhang, “Shot boundary detection based on multilevel difference of colour histograms,” 2016 First International Conference on Multimedia and Image Processing (ICMIP), pp. 15–22, 2016.

A. B. Colony, “Video shot boundary detection method using modified artificial bee colony and fast accelerated segment test algorithm,” International Journal of Engineering & Technology, vol. 8, no. 1.5, pp. 177–182, 2019.

C. Choudary and T. Liu, “Summarization of visual content in instructional videos,” IEEE Transactions on Multimedia, vol. 9, no. 7, pp. 1443–1455, 2007.

H. Liu, T. Tan, and T. Kuo, “A novel shot detection approach based on orb fused with structural similarity,” IEEE Access, vol. 8, pp. 2472–2481, 2020.

M. Mahrizi and S. Morwal, “Index point detection and semantic indexing of videos a comparative review,” Advances in Intelligent Systems and Computing 1617, pp. 20–28, 2017.

H. Liu, T.-H. Tan, and T.-Y. Kuo, “A novel shot detection approach based on orb fused with structural similarity,” IEEE Access, vol. 8, pp. 2472–2481, 2019.

S. R. Kumar, Rupesh and M. Sharma, “Abrupt scene change detection using spatiotemporal regularity of video cube,” Advances in VLSI Communication and Signal Processing, pp. 991–1002, 2020.

MEHUL MAHRISHI (IEEE Senior Member) is currently working as an Associate Professor in the Department of Information Technology at the Swami Keshvanand Institute of Technology, Management and Grantham, Jaipur – India. He is a Senior Member of the IEEE Delhi Section, a Life Member of the Institute of Engineers, India (I-IEI) and a Life Member of the International Association of Engineers (L-IAENG). In addition, he was a member of an Indian contingent of the BRICS International Festival that traveled to Moscow, Russia, to attend Russian Energy Week-2018 and the BRICS Youth Energy Summit. He has published more than 25 research papers in national and international journals/conferences, including 19th ISDA, 4th SoCTA, 3rd International Conference on Machine Learning and Computing (ICMLC), 3rd IACC and books/book chapters. He has also participated in various forums by Infosys, TC5, WIPRO-Mission 10 X and IBM. His research activities are currently twofold: while the first research activity is set to explore the developmental enhancements in applications in computer vision, the second central research theme focused on the emerging capabilities of parallel and cloud computing. Mr. Mahrizi is rewarded several occasions in various domains, including Recognition as an active reviewer by Journal of Parallel and Distributed Computing (JPDC, Elsevier, and SCI & Scopus Indexed), IEEE continuing education certification for “Cloud Computing Enable Technologies and Recognition for outstanding performance in Campus Connect Program by Infosys, India.

DR. SUDHA MORWAL is currently working as an Associate Professor in department of computer science, Banasthali Vidyapith. She is MSc, MTech and PhD in computer science and has played a prominent role in the department’s academic activities for the last decade. She is co-author of many books on computer science in Hindi and English. She has published many papers in national and international journals and conferences.
ABDUL WAHAB MUZAFFAR was born in Azad Jammu & Kashmir, Pakistan in 1983. He received a BS degree in Computer Science from the University of Azad Jammu and Kashmir, an MS degree in computing from the SZABIST Pakistan, in 2010 and a Ph.D. degree in software engineering from the National University of Sciences and Technology (NUST), Pakistan, in 2017. From 2010 to 2018, he worked in different roles at the National University of Sciences and Technology (NUST), Pakistan. Since August 2018, he has been an Assistant Professor with the Saudi Electronic University, Saudi Arabia. He is the author of 27 conference and journal papers. His research interests include software engineering, data and text mining, bioinformatics and machine learning. Dr. Abdul Wahab Muzaffar has participated in conferences in UAE, Thailand and the USA and presented his research. He also reviewed papers from a different journal, including PLOS ONE.

DR. SURBHI BHATIA is doctorate in Computer Science and Engineering from Banasthali Vidyapith, India, did her Masters in Technology from Amity University in 2012 and Bachelors in Information Technology in 2010. She has earned professional management professional certification from PMI, USA. She is currently an Assistant Professor in the Department of Information Systems, College of Computer Sciences and Information Technology, King Faisal University, Saudi Arabia. She has rich 8 years of teaching and academic experience. She is in the Editorial board member with Inderscience Publishers in the International Journal of Hybrid Intelligence, SN Applied Sciences, Springer. She has published many research papers in reputed journals and conferences in high indexing databases and have patents, granted from USA, Australia and India. She is currently serving as a guest editor of special issues in reputed journals. She has delivered talks as keynote speaker in IEEE conferences and in faculty development programs. She has authored two books and edited seven books from Springer, Wiley, and Elsevier. She has completed two funded research projects from Deanship of Scientific Research, King Faisal University and Ministry of Education, Saudi Arabia. Her research interests are Machine Learning, Sentiment Analysis, and Information Retrieval.

DR. PANKAJ DADHEECH received his Ph.D. degree in Computer Science & Engineering from Suresh Gyan Vihar University (Accredited by NAAC with ‘A’ Grade), Jaipur, Rajasthan, India. He received his M.Tech. degree in Computer Science & Engineering from Rajasthan Technical University, Kota and he has received his BE in Computer Science & Engineering from University of Rajasthan, Jaipur. He has more than 15 years of experience in teaching. He is currently working as an Associate Professor & Dy. HOD in the Department of Computer Science & Engineering (NBA Accredited), Swami Keshvanand Institute of Technology, Management & Gramothan (SKIT), Jaipur, Rajasthan, India. He has published 15 Patents at Intellectual Property India, Office of the Controller General of Patents, Design and Trade Marks, Department of Industrial Policy and Promotion, Ministry of Commerce and Industry, Government of India. He has published 4 Australian Patents at Commissioner of Patents, Intellectual Property Australia, Australian Government. He has also Registered & Granted Research Copyright at Registrar of Copyrights, Copyright Office, Department for Promotion of Industry and Internal Trade, Ministry of Commerce and Industry, Government of India. He has presented 51 papers in various National & International conferences. He has 38 publications in various International & National Journals. He has published 2 Books & 8 Book Chapters. He is a member of many Professional Organizations like the CSI, ACM, IAENG & ISTE. He has appointed as a Ph.D. Research Supervisor in the Department of Computer Science & Engineering at SKIT, Jaipur (Recognized Research Centre of Rajasthan Technical University, Kota). He has also guided various M.Tech. Research Scholars. He has Chaired Technical Sessions in various International Conferences & Contributed as Resource Person in various FDP’s, Workshops, STTP’s, and Conferences. He is also acting as a Guest Editor of the various reputed Journal publishing houses, Conference Proceedings and Bentham Ambassador of Bentham Science Publisher. His area of interest includes High Performance Computing, Cloud Computing, Information Security, Big Data Analytics, Intellectual Property Right and Internet of Things.

MOHAMMAD KHALID IMAM RAHMANI (IEEE Senior Member) was born in Patherghatti, Kishanganj, Bihar, India in 1975. He received the B.Sc. (Engg.) and the M.Tech. degrees in Computer Engineering from Aligarh Muslim University, India in 1998 and Maharshi Dayanand University Rohtak in 2010 respectively and the Ph.D. degree in Computer Science Engineering from Mewar University, India, in 2015. From 1999 to 2006, he was a Lecturer with Maulana Azad College of Engineering and Technology, Patna. From 2006 to 2008, he was a Lecturer and Senior Lecturer with Galgotias College of Engineering and Technology, Greater Noida. From 2010 to 2011 he was Assistant Professor in GSVNIEIT, Palwal. From 2017 he is Assistant Professor in the department of Computer Science, College of Computing and Informatics, Saudi Electronic University, Riyadh, Saudi Arabia. His research interests include Algorithms, IoT, Cryptography, Image Retrieval, Machine Learning, Deep Learning. He has published more than 30 research papers in journals and conferences of international repute and holds one patent of innovation. He also reviewed papers of different journal including Sādhanā (Springer) and International journal of advanced computer science and applications.