VOL. 19, NO. 3, 2020, 54-60
www.elektrika.utm.my
ISSN 0128-4428

ELEKTRIKA
Journal of Electrical Engineering

Keyframe Extraction Techniques: A Review

Bashir Olaniyi Sadiq[1], Bilyamin Muhammad[2*], Muhammad Nasir Abdullahi[3], Gabriel Onuh[1], Ali Abdulhakeem Muhammed[1] and **Adeogun Emmanuel Babatunde[2]**

[1]Department of Computer Engineering, Ahmadu Bello University Zaria, Kaduna State, Nigeria.
[2]Department of Computer Engineering, Kaduna Federal Polytechnic, Kaduna State, Nigeria.
[3]Department of Informatik and Automation, Technische Universität Ilmenau, Thuringia Germany.

[*]Corresponding author: bsmuhd@gmail.com, Tel: +2347064627804

Abstract: Video is an audiovisual data that comprises of large number of frames. Analyzing and processing such large amount of data is difficult to many applications. Therefore, there is need for an effective video management scheme to manage these huge volume of video frames in order to provide easy access to the video content in lesser time. Keyframe extraction is the first step for video browsing, indexing and retrieval. Many techniques exist for the extraction of Keyframes. However, some of the present techniques come with one or more limitations. In this paper, a brief review on the existing techniques is presented. Also, the merits and demerits of each technique is also stated.

Keywords: Keyframe Extraction, Shot Boundary Detection, Shot Transitions, Video Hierarchy

Article History: received 9 June 2020; accepted 6 December 2020; published 26 December 2020.

1. INTRODUCTION

The rapid increase in broadband data connection and digital video capturing devices has resulted to digital videos been widely utilized [1]. However, this growth in the availability of digital videos has not been accompanied by an increase in its accessibility [2]. In a situation where certain frames of interest are needed to be reviewed, the user has to browse/view the whole video data, and due to the huge number of frames available in the video, the process becomes difficult and time consuming [3]. To address this problem, an effective content-based video indexing, browsing, and retrieval (CBVIBR) approach is required [4].

Video summarization (also known as video abstraction) is the process of presenting an abstract view and comprehensible analysis of a full-length video within the shortest period of time [5]. Video skimming and keyframe selection are the two basic methods for video abstraction [6]. Video skimming is a method of summarization that extracts frames together with their matching soundtracks from a given video file. While keyframe extraction (also known as static video summarization or representative frames) is an efficient method that produces a more condensed version of the full-length video [7].

The keyframe extraction is a prerequisite needed for CBVIBR. The main idea of keyframe extraction is to manage large amount of video data by selecting unique set of representative frames while preserving the essential activities of the original video. Hence, resulting to simplicity in video analysis and processing [8]. Some of the areas touched by the development of the keyframe extraction techniques are; e-learning, news broadcast, home videos, sports, movies among other areas [9].

In general, the performance of a keyframe extraction technique is based on its ability to correctly extract unique keyframes from a given video. However, effects such as gradual transitions between successive frames have proven to be a challenging task for many keyframe extraction techniques as it normally spans for one or more seconds in videos depending on the editing effects used. In addition, the presence of sudden illuminance can also affect the efficiency of extracting a keyframe in a video sequence.

The rest of the paper is structured as follows: Section 2 presents the video hierarchy. Sections 3 presents video transitions. Section 4 presents the video summarization system, shot boundary detection technique, and keyframes extraction techniques. The metrics used for evaluating the performance of these techniques are discussed in section 5. While conclusions are done in section 6

2. VIDEO HIERARCHY

A video hierarchy is the entire structure of a video which comprises of scenes, shots, and frames. A story comprises of a number of scenes that captures sequence of event. Hence it is made up of interrelated shots recorded at different camera positions [10]. A shot is the smallest unit of temporal visual information that contains a sequence of interrelated frames captured uninterruptedly by a single camera [8]. These frames represent certain related actions or events in time and space. Figure 1 gives an illustration of a video hierarchy.

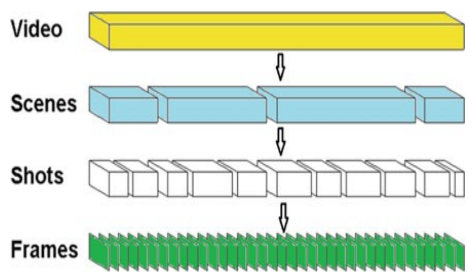

Figure 1. Video Hierarchy

3. VIDEO TRANSITIONS

Transition is a frontier between multiple video shots [8]. The Video Editing Process (VEP) is employed to merge multiple shots to generate a video during the Video Production Process (VPP) [11]. These VEPs allows the generation of various transition effects. The main types of shot transitions are shown in Figure 2.

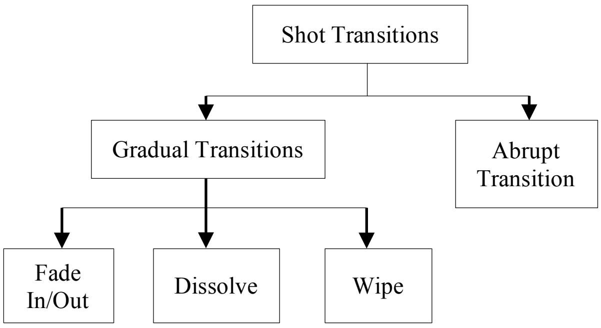

Figure 2. Types of Shot Transition [12]

3.1 Abrupt Transition

Abrupt (cut) transition is a sudden change that occurs between two successive video shots without any video editing process [13]. Figure 3 shows a sudden change between the last frame of the current shot and the first frame of the subsequent shot (i.e. between frames 3 and 4).

Figure 3. Abrupt transition

3.2 Gradual Transition

Gradual (soft) occurs when two successive shots are combined by making use of video editing process (VEP) throughout the course of production. It may span two or more video frames that contains truncated information and are visually interdependent [14]. In detecting shot boundary from a given video file containing soft transition, the result of the operation might not be efficiently achieved. This is attributed to the high visual content similarities between the consecutive frames involved in the VEP [8]. The gradual transitions are classified into three, namely; dissolve, fade in/out, and wipe transitions.

3.2.1 Dissolve Soft Transition (DST)

DST is the process in which the pixel intensity values gradually diminish from the current shot, and the values of the pixel intensity of the next shot gradually appears [15]. In DST, two or more frames may have different pixel intensity values but contains the same visual information as shown in figure 4 [12]. The figure depicts only one frame (i.e. 2nd frame) that is utilized in the dissolve transition.

Figure 4. Dissolve transition

3.2.2 Fade in/out Soft Transition (FST)

FST is the type of transition that is usually applied in movies to start a scene smoothly. In fade-in transition, one or more end frames of shot is directly changed by fixed intensity frame, and the pixel intensity values of the next shot gradually appears into position from a completely dark sequence [16]. Figure 5 shows an example of a fade-in transition with 4 frames involved in the transition (n = 1-4).

Figure 5. Fade-in Transition

In fade-out the transition, only frames at the end of the current shot are involved in the transition process, with no frames from the next shot are involved in the transition process. It is usually applied at the end of a movie scene. Figure 6 illustrates a fade-out transition involving 3 frames (n = 1-3).

Figure 6. Fade-out Transition

3.2.3 Wipe Soft Transition (WST)

WST is the process in which the current shot pixels are progressively superseded by the corresponding pixels from the next shot by following an organized spatial pattern [17]. Figure 7 shows the gradual substitution of the column pixels from left to right of the 10 frames involved (n= 1054-1080).

4. VIDEO ABSTRACTION

The rapid growth in network infrastructure together with the use of advanced digital video technologies necessitated the need for video abstraction technologies to manage the huge volumes of video data generated by enormous multimedia applications [4]. Hence, allowing the users to access and retrieve the relevant contents of the video easily without viewing the entire video.

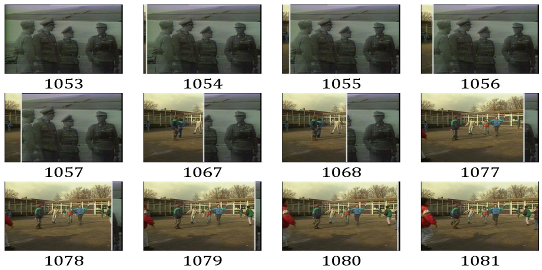

Figure 7. Wipe transition

The primary role of video summarization is to save and improve the storage capacity of the video contents efficiently, and also decrease the amount of data required when streaming or downloading video contents from the web [1]. Figure 8 shows a block diagram of a video summarization technology. The system consists of the shot boundary detection module where the video frames are partitioned into shots, and the keyframe extraction module where the number representative frames are identified and selected.

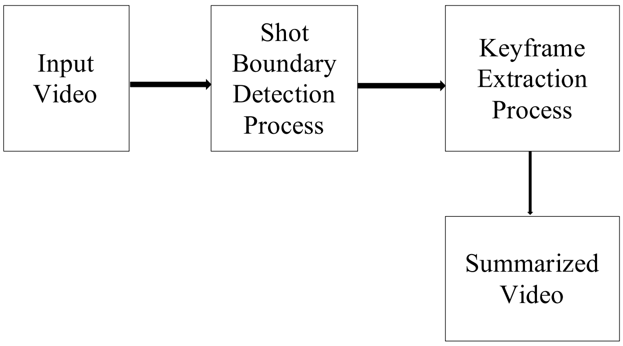

Figure 8. Block Diagram of Video Abstraction System [9]

4.1 Shot Boundary Detection Techniques (SBD)

Shot boundary detection (also known as temporal video segmentation) is the process of segmenting video frames into number of shots by determining the frontier between the successive video shots in order to make video analysis and processing easy [8]. The basic idea of shot boundary detection techniques is to find the dissimilarities of visual content. These variation between successive images are computed and a comparison with a threshold is established. A transition is then detected if a significant change occurs between the shots as shown in figure 9.

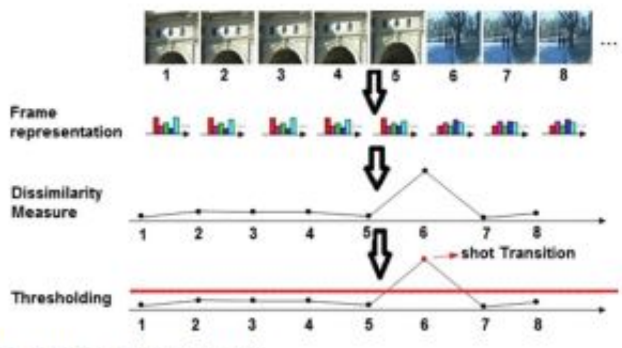

Figure 9. Shot Boundary Detection [12]

The SBD algorithms comprises of three core elements: Frame representation, Dissimilarity measure, and thresholding [18]. The major challenges faced by these SBD techniques is finding transitions in the presence of sudden illuminance and large camera/object movement, and such leading to the extraction of false keyframes [19]. Some of the existing SBD techniques are:

4.1.1 Pixel-Based Technique (PBT)

In this type of SBD method, the difference between two successive images is determined by directly comparing their pixel values using equation 1 [20]. PBT is computationally simple and best suited for the detection of abrupt transition [21]. However, a slight change in camera movement will result in multiple identical images becoming dissimilar. In addition, a minor variation in illuminance (flash light) results in false detection [3].

$$\frac{\left|\sum_{x=1}^{row}\sum_{y=1}^{col} k(m,x,y)-\sum_{x=1}^{row}\sum_{y=1}^{col} k(m-1,x,y)\right|}{256\cdot row\cdot col} > \tau \qquad (1)$$

Where *k (m, x, y)* and *k (m-1, x, y)* is the intensity value of the current and previous images in pixel *(x, y)*, and τ is the threshold value. A boundary is detected if the summation of the difference exceeds the threshold value [22].

4.1.2 Histogram-Based Technique (HBT)

HBT is the most common and widely approach used for detecting shot transitions [23]. A transition is detected if the variation between the histograms of two successive frames is greater than a threshold value as shown in equation 2 [20]. HBT is computationally simple and can detect abrupt, fade, and dissolve transitions efficiently [23]. Although HBT is insensitive to minor camera movement, it is sensitive when the movement is large [3]. Furthermore, the approach is sensitive to sudden illuminance and does not integrate the spatial distribution information of several colors, resulting in two different frames having the same histograms [21].

$$\frac{\sum_{c\in\{channels\}}\sum_{b=0}^{bins}|H(n,c,b)-H(n-1,c,b)|}{2\cdot|pixels|\cdot|channels|} > \tau \qquad (2)$$

Where H (n, c, b) and H (n-1, c, b) represent the values of bin b in color channel c for the histogram of two consecutive frames.

4.1.3 Statistical-Based Technique (SBT)

The SBT approach involves computing the histogram difference of consecutive frames and finding the threshold value by calculating the mean and standard deviation of the histogram differences. Then, a comparison is established between the absolute histogram differences of the frames and a threshold value. The histogram difference is computed using equation 3 [24].

$$R(i,i+1) = \sum_{a=1}^{3}\frac{[H(i,a)-H(i+1,a)]^2}{H(i,a)} \qquad (3)$$

The mean and standard deviation of the histogram differences are computed using equation 4 and 5 [24].

$$Mean = \sum_{i=1}^{M-1} \frac{R(i,i+1)}{M-1} \quad (4)$$

$$STD = \sqrt{\sum_{i=1}^{M-1} \frac{[R(i,i+1)-Mean]^2}{M-1}} \quad (5)$$

The threshold value is determined using equation 6 [24].

$$\tau = Mean + STD \times C \quad (6)$$

Where $R(i, i+1)$ is the histogram difference. i^{th} and $(i+1)^{th}$ is the current and next frames. *H(i, a)* and *H(i+1, a)* is the histogram of the color channels for consecutive frames. *M* is the total number of frames. *C* is pre-specified constant.

A shot transition is detected if the difference between the successive frames is greater than the threshold value. This approach has a high computational time due to the statistical calculations involved [3].

4.1.4 Edge-Based Technique (EBT)

This type of technique efficiently detects a boundary when the positions of edges of current frame shows a huge difference with that of the next frame [8]. The edge change ratio (ECR) is utilized to find the edge changes using equation 7 [20]. In EBA, transitions are detected by looking for large edge ratio [20]. Although, EBTs detects abrupt transition more accurately and can eliminate false positives resulting from flash light occurrence; however, they are less reliable compared to the HBAs in terms of performance and computational time [25]. Some of the EBAs employed for the detection of shot boundaries are; Roberts, Sobel, Prewitt, Laplacian of gaussian, and canny edge detection techniques [26].

$$ECR_n = max\left(\frac{X_n^{in}}{\sigma_n}, \frac{X_{n-1}^{out}}{\sigma_{n-1}}\right) \quad (7)$$

Where σ_n and σ_{n-1} are the numbers of edge pixels in the current and previous frames. X_n^{in} and X_{n-1}^{out} are the numbers of edge pixels entering and leaving in two successive images.

4.1.5 Machine Learning Based Technique

Recently, using machine learning techniques in the area of video processing have received tremendous attention from multimedia industries and academia. This is as a result of its significant capabilities to extract high level features from video frames [27]. However, it is computationally expensive due to the well-trained network model. The machine learning based technique is classified into two main classes namely; supervised and unsupervised. The unsupervised learning technique is the most common and widely used approach usually employed when the prior information about the dataset is unknown. While the supervised learning system can learn only those tasks that it's trained for [28]. A method based on the assumptions that hierarchy assists in making decision by reducing the number of unspecified transitions was proposed to detect abrupt transitions between successive shots [29]

A Support Vector Machine (SVM) learning approach was trained to classify frames as abrupt transitions or non-abrupt ones through the use of information theory to find the variation between the consecutive frames [30]. Similarly, an interpretable TAGs trained by convolutional neural networks (CNN) to predict the position at which a transition occurs in the video sequence [31].

4.2 Keyframe Extraction Techniques

Keyframe extraction is an efficient method used to clearly express the important contents of a video file by extracting a set of representative frames and removing/deleting the duplicated ones from the original video [6]. These extracted keyframes are expected to represent and provide comprehensive visual information of the whole video [7]. The keyframe approach is employed to reduce the computational burden and the amount of data needed for video processing so as to make indexing, retrieval, storage organization, and recognition of video data more convenient and efficient [32]. These techniques can be classified into three main classes namely; shot based, sampling-based, and clustering-based techniques [33].

4.2.1 Sampling-Based Technique

This is a type of method that selects representative frames by uniformly or randomly sampling the video frames from the original video, without giving importance to the video content [33]. The concept of this technique is to choose every k^{th} frame from the original video. This value of k is determined by the duration of the video. A usual choice of duration for a summarized video is 5% to 15% of the whole video. For the case of 5% summarization, every 20th frame is selected as the keyframe, while for the case of 15% summarization, every 7th frame is selected as the keyframe [34]. These keyframes extracted do not represent all the content of the original video, and may also result in redundant frames having similar contents [35].

4.2.2 Shot-Based Technique

In this approach, an efficient SBD method that detects shot boundary/transition is utilized first. After segmenting the video frames into various shots, the keyframe extraction process is then performed. Different kinds of literature have discussed different techniques for the selection of key frame. The traditional approach is to select the first and last frames of the candidate shot as the key frames [35]. These extracted key frames are the representative frames of the shots, which in turn produces the summary of the original video in a more condensed manner [36].

4.2.3 Clustering-Based Technique

Clustering is an unsupervised learning approach that finds sets of similar data points and cluster them together. In this method, frames within a video file having similar visual contents are partitioned into different number of clusters. And from each cluster, the frame that is nearest to the center of the candidate cluster is extracted as key frame [6]. The frame similarities are determined by the features they exhibit such as color histograms, texture, saliency maps, and motion [37]. The main drawback of the clustering-based technique is; it is difficult to determine the number of clusters in a given video file before performing the clustering operation [38].

4.2.4 Other Keyframe Extraction Approaches

The demerit of various existing techniques in the area of keyframe extraction is and high computational time in the process of extracting the keyframes. In addition, most them fail to select unique frames from a video sequence, which resulted to more redundant frames, and as such increasing the time required for video analysis and processing. In this regard, an AdaBoost classifier was trained to extract representative frames from vehicle surveillance footage. The algorithm was implemented in two modules. This proposed method has a high computational time due to the well-trained model needed [39].

A novel approach for shot transition detection and selection of representative images using Eigen values was presented [40]. In this approach, a data matrix was first created for all the successive frames in the original video. Covariance matrix was then calculated to determine the dissimilarities between the intensity levels of successive images. To reduce the computational burden, a modified approach for calculating the covariance matrix was utilized to recalculating the whole matrix whenever a new image is added to the data matrix. The calculated covariance matrix was then utilized to determine the Eigen values. The minimum Eigen value selected was utilized to determine the variations between the frames. A comparison was established between the minimum Eigen value and a predefined threshold. If the eigen value exceeds the threshold, then the previous image is considered as a transition point and the current image is selected as the representative frame.

A higher order color moment was used to extract keyframes from a video sequence by partitioning the video frames into M X N block shots [41]. From each shot, frames with most mean and standard deviation values are selected as the representative frames. Another method based on bitwise exclusive or (XOR) logical operation was presented to select keyframes by dissimilarity between two successive images [42].

5. EVALUATION METRICS

To validate the performance of the keyframe extraction techniques, several evaluation metrics are utilized namely; compression ratio, precision and recall, f-measure, and computational cost [32].

5.1 Compression Ratio

The Compression Ratio (CR) is used to determine the compactness of the technique due to the extracted key frames. CR is computed using equation 8 [7].

$$CR = \left\{1 - \frac{N_k}{N_f}\right\} \times 100\% \tag{8}$$

Where N_f is the total number of frames in the original video, and N_k is the total number of the extracted keyframes.

5.2 Precision and Recall

Precision also known as positive predictive value [43]. It is the ratio of the total number of keyframes extracted accurately to the total number of keyframes extracted by the technique from the video sequence. In other word, it measures the accuracy of a keyframe extraction technique, and computed using equation 9 [6].

$$Precision = \frac{N_a}{N_k} \times 100\% \tag{9}$$

Recall also known as sensitivity is computed using equation 10 [43].

$$Recall = \frac{N_a}{N_a + N_m} \times 100\% \tag{10}$$

Where N_a is the number of keyframes extracted accurately. N_k is the total number of the keyframes in the video sequence. N_m is the number of missed extractions.

5.3 F-Measure

F-measure (also known as f-score) is the method of evaluating the performance of an algorithm by merging both precision and recall to obtain one metric using the Harmonic mean. F-score is computed using equation 11 [8].

$$F = 2 \times \frac{precision \times recall}{precision + recall} \tag{11}$$

5.4 Computational Time

Computational cost is the time (measured in seconds) taken for the technique to extract the keyframes.

6. CONCLUSION

In this paper, a brief review of keyframe extraction techniques was carried out. Video structure, transition types, video abstraction, and metrics employed for measuring the performance of the techniques were discussed. Also, the advantages and disadvantages of each technique was stated. Although the performance of these techniques is acceptable, Keyframe extraction still face some challenges due gradual transitioned frames, camera operations (zooming, tilting, or panning) and sudden illuminance (flashlights) in the video sequence. Addressing these problems will improve the performance of keyframe extraction techniques.

REFERENCES

[1] C. Sujatha and U. Mudenagudi, "A study on keyframe extraction methods for video summary," in 2011 International Conference on Computational Intelligence and Communication Networks, 2011, pp. 73-77: IEEE.

[2] J. Li, Y. Ding, Y. Shi, and W. Li, "A Divide And Rule Scheme For Shot Boundary Detection Based on SIFT," International Journal of Digital Content Technology and its Applications, vol. 4, pp. 202-214, 2010.

[3] J. H. Yuan, H. Y. Wang, and B. Zhang, “A formal study of shot boundary detection”, Journal of Transactions on Circuits and Systems for Video Technology, vol. 17, no. 2, pp. 168-186, February 2007

[4] G. Kumar, Naveen, V. Reddy, and S. S. Kumar, "Video shot boundary detection and key frame extraction for video retrieval," in Proceedings of the Second International Conference on Computational Intelligence and Informatics, 2018, pp. 557-567: Springer.

[5] S. Santini, "Who needs video summarization anyway?," in International Conference on Semantic Computing (ICSC 2007), 2007, pp. 177-184: IEEE.
[6] A. Paul, K. Milan, J. Kavitha, J. Rani, and P. Arockia, "Key-Frame Extraction Techniques: A Review," Recent Patents on Computer Science, vol. 11, no. 1, pp. 3-16, 2018.
[7] H. Gharbi, S. Bahroun, and E. Zagrouba, "A Novel Key Frame Extraction Approach for Video Summarization," in VISIGRAPP (3: VISAPP), 2016, pp. 148-155.
[8] S. H. Abdulhussain, A. R. Ramli, M. I. Saripan, B. M. Mahmmod, S. A. R. Al-Haddad, and W. A. Jassim, "Methods and challenges in shot boundary detection: a review," Entropy, vol. 20, no. 4, p. 214, 2018.
[9] M. Furini, F. Geraci, M. Montangero, M. Pellegrini, and Applications, "STIMO: STIll and MOving video storyboard for the web scenario," Multimedia Tools Applications, vol. 46, no. 1, p. 47, 2010.
[10] C. Liu, D. Wang, J. Zhu, and B. Zhang, "Learning a contextual multi-thread model for movie/tv scene segmentation," IEEE transactions on multimedia, vol. 15, no. 4, pp. 884-897, 2013.
[11] O. Küçüktunç, U. Güdükbay, Ö. Ulusoy, and I. Understanding, "Fuzzy color histogram-based video segmentation," Computer Vision Image Understanding, vol. 114, no. 1, pp. 125-134, 2010.
[12] I. A. Zedan, K. M. Elsayed, and E. Emary, "News Videos Segmentation Using Dominant Colors Representation," in Advances in Soft Computing and Machine Learning in Image Processing: Springer, 2018, pp. 89-109.
[13] X. Ling, O. Yuanxin, L. Huan, and X. Zhang, "A method for fast shot boundary detection based on SVM," in 2008 Congress on Image and Signal Processing, 2008, vol. 2, pp. 445-449: IEEE.
[14] X. Jiang, T. Sun, J. Liu, J. Chao, and W. Zhang, "An adaptive video shot segmentation scheme based on dual-detection model," Neurocomputing, vol. 116, pp. 102-111, 2013.
[15] K. Choroś, "Reduction of faulty detected shot cuts and cross dissolve effects in video segmentation process of different categories of digital videos," in Transactions on computational collective intelligence V: Springer, 2011, pp. 124-139.
[16] Z. Cernekova, I. Pitas, and C. Nikou, "Information theory-based shot cut/fade detection and video summarization," IEEE Transactions on circuits systems for video technology, vol. 16, no. 1, pp. 82-91, 2005.
[17] Y. Kawai, H. Sumiyoshi, and N. Yagi, "Shot Boundary Detection at TRECVID 2007," in TRECVID, 2007: Citeseer.
[18] J. Yuan et al., "A formal study of shot boundary detection," IEEE transactions on circuits systems for video technology, vol. 17, no. 2, pp. 168-186, 2007.
[19] L. Xue, C. Li, H. Li, and Z. Xiong, "A general method for shot boundary detection", In Proceedings of the International Conference on Multimedia and Ubiquitous Engineering, pp. 394-397, 2008.
[20] K. Wu, "Simple Implementations of Video Segmentation, Key Frame Extraction and Browsing," 2011.
[21] H. H. YU, and W. WOLF, "A hierarchical multiresolution video shot Transition detection scheme", Journal of Computer Vision and Image Understanding, vol. 75, no. 1/2, pp. 196-213, 1999.
[22] M.-S. Lee, Y.-M. Yang, and S.-W. Lee, "Automatic video parsing using shot boundary detection and camera operation analysis," Pattern Recognition, vol. 34, no. 3, pp. 711-719, 2001.
[23] C. Vora, B. K. Yadav, and S. Sengupta, "Comprehensive Survey on Shot Boundary Detection Techniques," International Journal of Computer Applications, vol. 140, pp. 24-30, 2016.
[24] Kathiriya, Dhaval S. Pipalia, Gaurav B. Vasani, Alpesh J. Thesiya, and D. J. Varanva, "X^2 (Chi-Square) Based Shot Boundary Detection and Key Frame Extraction for Video," International Journal of Engineering and Science, vol. 2, no. 2, pp. 17-21, 2013.
[25] A. Dailianas, R. B. Allen, and P. England, "Comparison of automatic video segmentation algorithms," in Integration Issues in Large Commercial Media Delivery Systems, 1996, vol. 2615, pp. 2-16: International Society for Optics and Photonics.
[26] S. Bhardwaj and A. Mittal, "A survey on various edge detector techniques," Procedia Technology, vol. 4, pp. 220-226, 2012.
[27] Nishani, E.; Çiço, B. Computer vision approaches based on deep learning and neural networks: Deep neural networks for video analysis of human pose estimation. In Proceedings of the 2017 6th Mediterranean Conference on Embedded Computing (MECO), Bar, Montenegro, 11–15 June 2017; pp. 1–4.
[28] N. J. Janwe and K. K. Bhoyar, "Video key-frame extraction using unsupervised clustering and mutual comparison," International Journal of Image Processing,vol. 10, no. 2, pp. 73-84, 2016.
[29] C. G. Chávez, F. Precioso, M. Cord, S. Phillip-Foliguet, and A. d. A. Araújo, "Shot Boundary Detection by a Hierarchical Supervised Approach," pp. 197-200, 2007.
[30] J. Bi, X. Liu, and B. Lang, "A Novel Shot Boundary Detection Based on Information Theory using SVM," International Congress on Image and Signal Processing, pp. 512-516, 2011.
[31] W. Tong, L. Song, X. Yang, H. Qu, and R. Xie, "CNN-based shot boundary detection and video annotation," in 2015 IEEE international symposium on broadband multimedia systems and broadcasting, 2015, pp. 1-5: IEEE.
[32] C. V. Sheena and N. Narayanan, "Key-frame extraction by analysis of histograms of video frames using statistical methods," Procedia Computer Science, vol. 70, pp. 36-40, 2015.

[33] M. Asim, N. Almaadeed, S. Al-Máadeed, A. Bouridane, and A. Beghdadi, "A key frame based video summarization using color features," in 2018 Colour and Visual Computing Symposium (CVCS), 2018, pp. 1-6: IEEE.
[34] S. Jadon and M. Jasim, "Video Summarization," EasyChair2516-2314, 2019.
[35] S. M. Tirupathamma, "Key frame based video summarization using frame difference," International Journal of Innovative Computer Science & Engineering, vol. 4, no. 03, pp. 160-165, 2017.
[36] P. Kaur and R. Kumar, "Analysis of Video Summarization Techniques," International Journal for Research in Applied Science & Engineering Technology (IJRASET), vol. 6, no. 01, 2018.
[37] X. Li, B. Zhao, and X. Lu, "Key frame extraction in the summary space," IEEE transactions on cybernetics, vol. 48, no. 6, pp. 1923-1934, 2017.
[38] C. Huang and H. Wang, "A Novel Key-frames Selection Framework for Comprehensive Video Summarization," IEEE Transactions on Circuits and Systems for Video Technology, 2018.
[39] J. Yuan, W. Wang, W. Yang, and M. Zhang, "Keyframe extraction using AdaBoost," in Proceedings 2014 IEEE International Conference on Security, Pattern Analysis, and Cybernetics (SPAC), 2014, pp. 91-94: IEEE.
[40] V. Benni, R. Dinesh, P. Punitha, and V. Rao, "Keyframe extraction and shot boundary detection using eigen values," International Journal of Information Electronics Engineering, vol. 5, no. 1, p. 40, 2015.
[41] P. Jadhava and D. Jadhav, "Video summarization using higher order color moments," in Proceedings of the International Conference on Advanced Computing Technologies and Applications (ICACTA), 2015, vol. 45, pp. 275-281.
[42] B. Rashmi and H. Nagendraswamy, "Shot-based keyframe extraction using bitwise-XOR dissimilarity approach," in International Conference on Recent Trends in Image Processing and Pattern Recognition, 2016, pp. 305-316: Springer.
[43] A. S. Murugan, K. S. Devi, A. Sivaranjani, and P. Srinivasan, "A study on various methods used for video summarization and moving object detection for video surveillance applications," Multimedia Tools Applications, vol. 77, no. 18, pp. 23273-23290, 2018.