VOL. 21 NO. 2, 2022, 1-6
www.fke.utm.my/elektrika
ISSN 0128-4428

ELEKTRIKA
Journal of Electrical Engineering

Keyframe Extraction for Low-Motion Video Summarization Using K-Means Clustering

Bilyamin Muhammad*, **Mariam Abdulazeez Ahmed, Ibrahim Haruna,** and **Usman Ismail Abdullahi**

Department of Computer Engineering, Kaduna Polytechnic, Kaduna State, Nigeria.

*Corresponding author: bilyaminmuhammad@kadunapolytechnic.edu.ng, Tel: +2347064627804

Abstract: The rate of increase in multimedia data required the need for an improved bandwidth utilization and storage capacity. However, low-motion videos come with a large number of feature-related frames due to its static background. These redundant frames result to difficulty in terms of video streaming, retrieval, and transmission. In other to improve the user experience, video summarization technologies were proposed. These techniques were presented to select representative frames from a full-length video and remove the duplicated ones. Though, an improvement was recorded in the keyframe extraction process. However, a large number of redundant frames were observed to be extracted as keyframes. Therefore, this study presents an improved keyframe extraction scheme for low-motion video summarization. The proposed scheme utilizes a k-means clustering approach to group the feature-related frames within a given video data into number of clusters. Furthermore, a representative frame from each cluster was extracted as keyframe. The results obtained showed the proposed scheme outperforms the existing schemes that utilized the histogram based approach.

Keywords: K-Means Clustering, Keyframe, Low-Motion Video, Video Summarization.

Article History: received 8 October 2021; accepted 19 July 2022; published 25 August 2022.

1. INTRODUCTION

Video summarization has received a lot of attention from the multimedia industries as well as the academia due to the advancement in digital video recording technologies and the increasing rate at which videos are been retrieved and transmitted over the internet. These industries are faced with the problem of bandwidth utilization and limited storage capacity as a result of the large volume of video frames [1]. In low-motion videos such as surveillance footage, e-learning and news broadcast, various frames exhibit similar features due to its static background. Hence, necessitating the need for an efficient keyframe video summarization scheme that will precisely select unique frames to represent the entire video and eliminate the redundant frames [2].

The keyframe video summarization scheme (also referred to as static video abstraction) is the process of providing a comprehensive version of an entire video through the extraction of key images while maintaining the quality as well as the important events of the original video. The goal of this scheme is to shrink the volume of video frames required for processing in order to create an effective means for retrieving, storing, and transmitting the video data conveniently. Furthermore, it provides the users an easy access to the significant features of the video file without viewing the entire video content [3].

Several studies have been carried out in the area of keyframe video summarization so as to improve the transmission rate and storage capacity. However, frames generated due to the presence of gradual transitions, camera zooming, and sudden illuminance in the video can never guarantee optimal utilization of bandwidth and storage [4]. Therefore, this study seeks to improve on the existing schemes by employing a k-means clustering approach to group similar video frames into a single cluster and frames closest to the centroid are selected as the representative frames. During streaming process, frames extracted as keyframes using the proposed scheme are sent from the server to the client. Each extracted keyframe information serve as a reference to the successive frame that will be received from the server. This process continues until all the representative frames in the whole video file are transmitted. Hence, eliminates redundant frames in the video file. Consequently, improving bandwidth utilization during streaming and providing a better compression ratio as well as minimal memory space requirement.

The rest of the paper is organized as follows: Section 2 presents review of related works, section 3 presents the materials and method utilized, and the experimental results are presented in section 4, while conclusions are done in section 5

2. RELATED WORK

This section presents the review of related works that have been carried out in the area of keyframe extraction. The major limitation of these works is their ineffectiveness to select unique keyframes in a video data and consequently, resulting to high bandwidth and storage capacity requirement.

An approach based on color features for the extraction

of representative frames was presented in [5]. The authors considered the first image in any given video as the reference image, and segment the remaining images into blocks. The color mean variations between corresponding blocks in the reference and current image were then calculated. The varying blocks in the current images in relation to the varying blocks in the reference image were then counted. If the count number is more than a predefined threshold, then the current image is selected as keyframe. It was observed that the proposed approach can detect camera movement efficiently, and extracts keyframes in both fast-moving and low-motion videos. However, this work can only select keyframes in videos having a high variation in color intensity between the frames.

Yuan et al. [6] implemented a method for extracting representative frames from vehicle surveillance video using an AdaBoost classifier. The algorithm was implemented in two modules. The first module involved training the AdaBoost classifier to select the region and integral channel features as the frame feature descriptors. The second module involved utilizing the trained AdaBoost classifier to select the representative images. Experimental result showed that the proposed method can extract unique set of representative frames with less transitioned frames. However, it has a high computational time because of the well-trained model needed. Another method based on higher order color moments was presented in [7]. The video frames were first partitioned into M X N block. Then each block is divided into shots using frame histogram, skew and kurtosis values. From each shot, frames with most mean and standard deviation values are selected as the representative frames.

A novel technique was presented for the selection of representative images using Eigen values [8]. The authors first created a data matrix 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. A modified approach for calculating the covariance matrix was also presented to reduce the computational cost of 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.

Raikwar et al. [9] proposed a method for the selection of representative frames using human perception. At first, the entire video frames was extracted and segmented into number of shots. From each shot the first frame is directly selected as the representative image. Though, the proposed method summarized entire video at a very low processing time. However, the selected keyframes might not be the most representative frames of the entire video.

Li et al. [10] presented a keyframe extraction scheme based on sparse coding. The video frames were first extracted and segmented into number of shots using the dictionary items created by the sparse coding. The similarities between consecutive frames were then computed. Finally, frames higher features are selected as keyframes. However, frames affected by sudden illuminance were extracted as keyframes.

Lee et al. [11] presented an approach for selecting keyframes in 3D motion videos. The authors used the depth information of the video to find the respective gradient of each frame and composed the selected frames into a single frame summary. This single frame summary comprises various foreground visual features which are built based on the significance of each image computed using their respective gradient. Furthermore, the authors employed a threshold based technique to minimize the number of the foreground features and thus, reducing visual complexity of the video frames. Nonetheless, important information present in the temporal frames is missed.

Kumar and Kumar [12] presented an indexing frame approach for improving bandwidth utilization in live video streaming. The proposed approach comprises two modules namely; the server and client modules. The server module sends unique frames' pixels for reconstruction at the client end using the first video frame as reference frame. Another authors improved on the work presented in [12] by developing a spatio temporal based frame indexing approach to improve Quality of Service (QoS) in live low motion video [13]. The proposed approach exploit both the spatial and temporal information between successive frames, and redundant pixels presents are eliminated. Despite the improvement in compression ratio and bandwidth utilization, it was observed that certain number of redundant pixel affected by light intensity is sent to the client end. In addition, the approach is computationally complex due to the reconstruction process.

A statistical technique for selecting keyframes in video data was proposed in [14]. The total video frames were first extracted and the absolute difference between their histogram was calculated. A comparison was then established with a defined threshold value. If the difference between successive frames is more than the threshold, then a frame is selected as representative frame. Results obtained showed that the proposed method is computationally easy. However, gradually transitioned frames were extracted as keyframes. Rodriguez et al. [15] improved on this approach by computing the histogram equalization between the successive video frames. Though, an improvement was recorded in the keyframe extraction, however, the frames affected by sudden flashlight were extracted as keyframes.

Similarly, Muhammad et al. [16] implemented an approach for extracting keyframes in fast-moving videos using histogram difference and k-means clustering. The authors first adopted a histogram based approach for segmenting the video frames into a number of shots. A k-means clustering approach was then employed to group similar frames within a candidate shot into clusters, and keyframes were selected from each cluster. However, the proposed approach was only implemented on fast-moving movies.

3. MATERIALS AND METHOD

In this section, the materials and method employed for extracting keyframes in low-motion videos is presented. This involves the utilization of a computer system with the following specifications: 8GB RAM, 1TB HDD, and Intel (R) Core i5 @ 2.54GHz processor. In addition, MATLAB version R2018a was used for implementing the proposed methodology.

3.1 Input Videos

The first step of the keyframe extraction process is video acquisition. The proposed scheme was tested on a standard low-motion video (Akiyo) acquired from YouTube. The video can be accessed via the link: https://www.youtube.com/watch?v=onfjPHNU9EM.

3.2 Keyframe Extraction

A k-means clustering is employed to extract the representative frames from the input videos. The total number of frames within the candidate video is first extracted and clustered based on their feature similarities. From each cluster, a centroid is determined by computing the sum of distances between all the cluster frames. The difference between the cluster centroids and the frames within them is computed. The frame with least distance to the centroid is selected as the representative frames. The Pseudo code for the keyframe extraction using k-means clustering approach is presented in algorithm 1.

Algorithm 1: Developed Scheme

Input: *E (input videos)*
Output: *Key (keyframes)*
Step1: Read the total video frames
Step2: find number of clusters
Step3: Initial centroid, Cj ← mean(c1, c2,….ck)
Step4: For Di ← (1<= i <=n)
Step5: Find the closest frame
Step6: Key ← (Di, Cj)
//end for
Step7: Repeat
Step8: For new cluster, Di ← (i <= Cj)
Step9: Frame stays in the cluster, Di ← i
//else
Step10: New cluster is form
//end for
Step 11: Return assignment

Algorithm 1 depicts the step by step procedure followed in the extraction of the representative frames from the original video file. An explanation of each step is presented herewith.
Step 1: The total video frames in the input video was extracted and stored in a defined location.
Step 2 and 3: The number of cluster centroid is randomly selected and the distance between each video frame and cluster centroid is computed using Equation (1) [17].

$$Dist_{XY} = \sqrt{\sum_{k=1}^{m}(X_{ik} - X_{jk})^2} \quad (1)$$

Step 4: The frame is then allocated to a cluster whose distance from the cluster centroid is minimum of all the cluster centroids.
Step 5 and 6: The distance between frames within a candidate cluster and its centroid is then computed using Equation (1)
Step 6: Frame closest to the centroid is selected as keyframe.
Step 7: Repeat step 5 for optimal extraction of the cluster's representative frame
Step 8: New cluster centroid is computed using Equation (2) [17].

$$V_i = (\frac{1}{C_i}) \sum_1^{C_i} X_i \quad (2)$$

Step 9 and 10: The distance between each frame and new obtained cluster centroid is recalculated.

3.3 Performance Evaluation

To test the performance of the proposed scheme, several evaluation metrics are utilized viz: compression ratio, precision and recall rates [18].

2.3.1 Compression Ratio (CR)

It is used to determine the compactness of the proposed scheme due to the extracted keyframes. CR is computed using Equation (3).

$$CR = \left\{1 - \frac{N_k}{N}\right\} \times 100\% \quad (3)$$

Where N is the total number of frames in the original video. N_k is the total number of the extracted keyframes.

2.3.2 Precision and Recall Rates

Precision rate is the measure of accuracy of the proposed scheme, and it's computed by the ratio of the total number of keyframes extracted correctly to the total number of keyframes extracted from the original video. Equation (4) is used to determine the accuracy of the scheme.

$$Precision = \frac{N_a}{N_a + N_f} \times 100\% \quad (4)$$

Recall rate is determined by computing the total number of keyframes precisely extracted divided by the total number of ground truth frames. It is measured using Equation (5).

$$Recall = \frac{N_a}{N_a + N_m} \times 100\% \quad (5)$$

Where N_a is the number of frames extracted accurately. N_f is the number of false extraction; and N_m is the number of missed key frames from the video frames.

4. RESULTS AND DISCUSSION

This section present the simulation results obtained using the proposed methodology. The performance of the proposed scheme is evaluated and compared with existing works presented in [14] and [15] based on the evaluation metrics presented in section 3.3. MATLAB version R2018a was utilized for the implementation of the proposed methodology. The results obtained are summarized in Table 1.

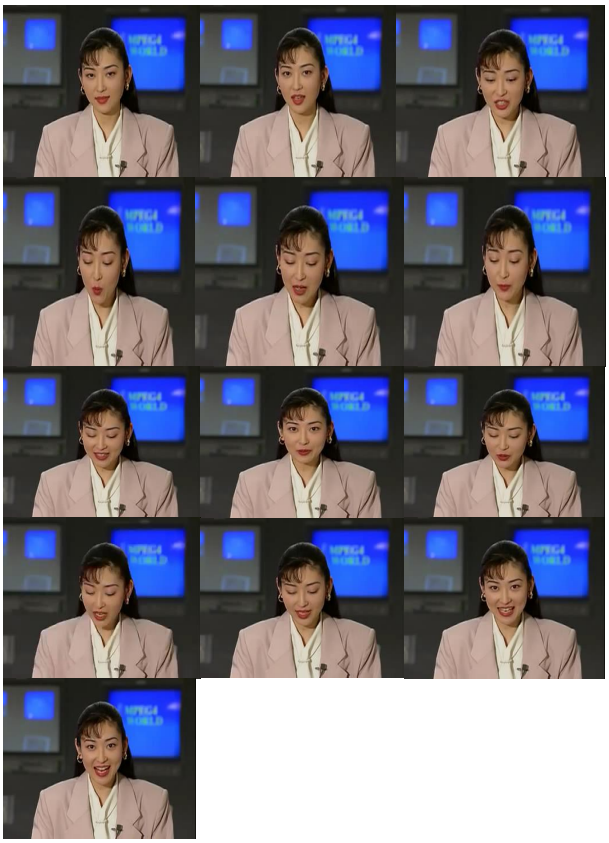
Table 1. Simulation results

Total Frames	Keyframes		
	Sheena and Narayanan [14]	Rodriguez et al. [15]	Proposed Scheme
301	96	72	13

Table 1 shows the total number of frames in the original video, and its corresponding keyframes extracted using both the proposed and existing methodologies. It can be observed that the existing schemes extracted a higher number of keyframes compared to the proposed scheme. This is due to the extraction of feature related frames. However, the proposed scheme was able to reduce these redundant frames by clustering the similar frames and extracting the most representative one among them as a keyframe without affecting the integrity of the video file.

Figure 1 present sample of the entire video frames and their representative frames extracted using the proposed scheme.

(a)



(b)

Figure 1. (a) Total Video Frames, (b) Keyframes.

Figures 1 depict the entire video frames and the keyframes extracted using the proposed scheme. These frames represent the entire video, as no multiple feature-related frames were extracted.

Observe that the proposed scheme was able to extract unique keyframes from the standard low-motion video used. This is due to the utilization of the k-means clustering approach which grouped feature-related frames into clusters. The most representative frames among them were selected as keyframes without degrading the integrity of the video. Furthermore, the duplicated frames are eliminated; hence, improved the storage capacity and transmission rate of the video data. Table 2 shows the performance evaluation of the proposed scheme using compression ratio, precision and recall rates.

Table 2. Comparison using CR, precision and recall rates

Scheme	CR	Precision	Recall
Sheena and Narayanan [14]	68.10%	82.46%	89.70%
Rodriguez et al. [15]	76.08%	90.52%	91.03%
Proposed	95.68%	100%	92.86%

Table 2 depict a comparison between the proposed scheme and two existing histogram based schemes in terms of CR, precision and recall rates. It can be seen that the proposed scheme extracted less keyframes with only

one frame missed compared to the existing schemes. Thus, minimize the redundancy of extracted keyframes.

Figure 2 depict the comparative results of the proposed scheme and that of Sheena and Narayanan [14] and Rodriguez et al. [15].

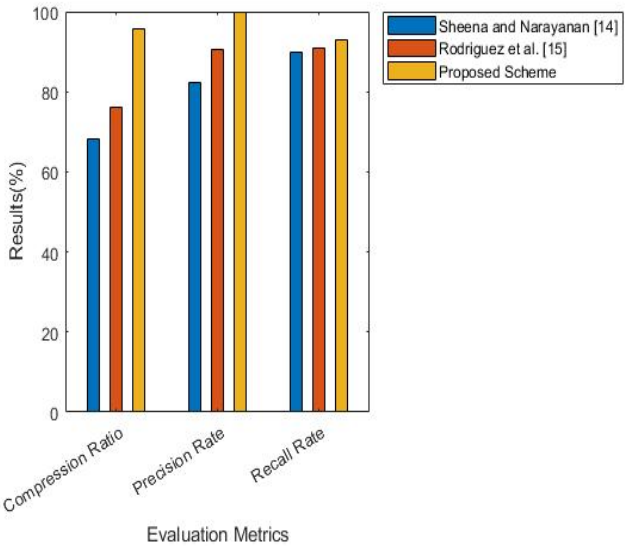


Figure 2. Comparison Results of the Proposed Approach

It can be seen the proposed scheme outperforms the existing schemes of Sheena and Narayanan [14] and Rodriguez et al. [15] by 25.76% and 40.50% in terms of compression ratio, and 21.27% and 10.47% in terms of precision rate. In addition, it outperforms the existing schemes by 3.52% and 1.83% in terms of recall rate.

5. CONCLUSION

In this study, a k-means clustering approach is presented for the extraction of representative frames in low-motion videos. The k-means clustering was utilized to cluster feature related video frames into a single cluster and from each cluster, frames closest to the centroid are selected as the representative frames. The proposed scheme was tested on a standard low-motion video obtained from YouTube, and was able to provide a condensed version of the entire video. Thus, making it suitable for browsing, retrieval, and indexing. Furthermore, the experimental results show that the proposed scheme outperforms the existing schemes in terms of compression ratio, precision and recall rates.

ACKNOWLEDGMENT

The authors wish to thank Engr. Ali Abdulhakeem Muhammed for all the corrections and technical support.

REFERENCES

[1] C. S. Mithlesh and D. Shukla, “A case study of key frame extraction techniques,” International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, vol. 5, no 3, pp. 1292-1298, 2016.

[2] 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.

[3] B. O. Sadiq, B. Muhammad, M. N. Abdullahi, G. Onuh, A. A. Muhammed, and A. E. Babatunde, "Keyframe Extraction Techniques: A Review," ELEKTRIKA-Journal of Electrical Engineering, vol. 19, no. 3, pp. 54-60, 2020

[4] 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.

[5] C. Cao, Z. Chen, G. Xie, and S. Lei,”Key frame extraction based on frame blocks differential accumulation,” 24th Chinese Control and Decision Conference, 2012, pp. 3621-3625.

[6] 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.

[7] 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.

[8] 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, pp. 40, 2015.

[9] S. C. Raikwar, C. Bhatnagar, and A. S. Jalal, "A frame work for key-frame extraction from surveillance video," Paper presented at the 2014 fifth IEEE International Conference on Computer and Communication Technology (ICCCT), 297-300, 2014.

[10] 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.

[11] H. J. Lee, H. J. Shin, and J. J. Choi, "Single image summarization of 3D animation using depth images," Comput Animat Virtual Worlds. 23(3-4), 417-424, 2012.

[12] R. J. N. Kumar, and A. P. Kumar, "Improving QoS in live video streaming for low-motion videos through frame indexing method, International Journal of Communicaton Technology for, 2016.

[13] A. E. Adedokun, M. B. Abdulrazak, M. O. Momoh, H. Bello-Salau, and B. O. Sadiq, "A spatio-temporal based frame indexing algorithm for qos improvement in live low-motion video streaming," ATBU Journal of Science, Technology Education, 7(3), 305-315, 2019.

[14] 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.

[15] J. M. D. Rodriguez, P. Yao, and W. Wan, "Selection of key frames through the analysis and calculation of the absolute difference of histograms," Paper presented at the 2018 International Conference on Audio, Language and Image Processing, 2018.

[16] B. Muhammad, B. O. Sadiq, I. J. Umoh and H. Bello-Salau, “A k-means clustering approach for extraction of keyframes in fast- moving videos,” International Journal of Information Processing and Communication (IJIPC), vol. 9, no. 1&2, pp. 147-157, 2020.

[17] S. Archana, Y. Avantika, and R. Ajay, "K-means with Three different Distance Metrics," International Journal of Computer Applications, 67(10), 13-17, 2013.

[18] H. Gharbi, S. Bahroun, and E. Zagrouba, "A novel key frame extraction approach for video summarization," in VISIGRAPP (3: VISAPP), 148-155, 2016.