Towards Enhancing Keyframe Extraction Strategy for Summarizing Surveillance Video: An Implementation Study

Bashir Olaniyi Sadiq[1, *], Habeeb Bello-Salau[1], Latifat Abduraheem-Olaniyi[1], Bilyamin Muhammad[2] & Sikiru Olayinka Zakariyya[3]

[1]Department of Computer Engineering, Ahmadu Bello University, Sokoto Road, Samaru, P.M.B 1044, Zaria, Kaduna State, Nigeria
[2]Department of Computer Engineering, Federal Polytechnic Kaduna, Kaduna State, Nigeria
[3]Department of Electrical and Electronics Engineering, University of Ilorin, Kwara State, Nigeria
*E-mail: bosadiq@abu.edu.ng

Abstract. The large amounts of surveillance video data are recorded, containing many redundant video frames, which makes video browsing and retrieval difficult, thus increasing bandwidth utilization, storage capacity, and time consumed. To ensure the reduction in bandwidth utilization and storage capacity to the barest minimum, keyframe extraction strategies have been developed. These strategies are implemented to extract unique keyframes whilst removing redundancies. Despite the achieved improvement in keyframe extraction processes, there still exist a significant number of redundant frames in summarized videos. With a view to addressing this issue, the current paper proposes an enhanced keyframe extraction strategy using k-means clustering and a statistical approach. Surveillance footage, movie clips, advertisements, and sports videos from a benchmark database as well as Compeng IP surveillance videos were used to evaluate the performance of the proposed method. In terms of compression ratio, the results showed that the proposed scheme outperformed existing schemes by 2.82%. This implies that the proposed scheme further removed redundant frames whiles retaining video quality. In terms of video playtime, there was an average reduction of 27.32%, thus making video content retrieval less cumbersome when compared with existing schemes. Implementation was done using MATLAB R2020b.

Keywords: *keyframe extraction; surveillance video; video compression; video storage; video summarization.*

1 Introduction

Video is the recording of moving visual images made digitally or on videotape. A video recording contains audiovisual data that comprises multiple shots [1-3]. A video shot is a collection of interrelated frames caught by a solitary camera isolated by a limit [4-6], thus forming a polished video. The polished video

Received September 20th, 2021, Revised June 8th, 2022, Accepted for publication August 23rd, 2022.
Copyright © 2022 Published by IRCS-ITB, ISSN: 2337-5787, DOI: 10.5614/itbj.ict.res.appl.2022.16.2.5

prompts progressively redundant frames, thus expanding the video information [6-7]. Handling this video information requires the utilization of video abstraction techniques [8-9]. Video summarization, sometimes referred to as video abstraction, is the process of removing redundant frames, thus giving a complete perspective on the full-length video [10]. Video summarization is required because the advancement in digital video recording machines has increased the storage requirements for video data [11].

A typical scenario where video summarization is employed is in surveillance systems. In surveillance systems, cameras are employed for open security, ending up recording relevant as well as irrelevant video frames, subsequently, bringing about the generation of a huge amount of video frames for storage [9]. This huge amount of video data not only consumes storage space but also makes event retrieval tasking. Another example of video summarization is in movie trailers. In a movie trailer, a completed movie is often broken down for advertisement purposes.

Dynamic video summarization and keyframe determination are the two primary methodologies for video abstraction [8]. Dynamic video summarization produces a theoretical form of the entire video alongside its related soundtrack. Keyframe extraction, also known as static video summarization, is a methodology that gives a more consolidated adaptation of the original video by extricating representative frames from applicant shots [12]. A number of methods exist for the discovery of transitions and extraction of keyframes with a view to reducing the amount of redundancy in video frames transmitted over the system. These methods are said to have improved the data transfer capacity use, stockpiling limit, and reduction in transmission rate [13].

A histogram-based approach has been previously used to extract keyframes from full-length videos by computing the mean and standard deviation of the absolute difference between successive video frames [2]. The histogram-based approach was used to detect shot boundaries in video frames. The researchers that applied this approach also combined it with the k-means clustering technique to extract unique keyframes [1]. The present research proposes the development of an improved keyframe extraction method for video surveillance using the k-means clustering and statistical approach. The statistical approach is a filtering process based on changes in pixel levels.

It is an established fact that captured video data from surveillance systems is usually large [14-17], hence taking up large storage space [18] and making event retrieval tasking. To solve this problem, research works such as [2] and [19] have developed video summarization techniques using the k-means clustering and histogram-based approach to extract a set of keyframes from videos. The earlier

techniques presented by these researchers considered a predefined number of keyframes to be extracted with a view to reducing the storage requirement. Nonetheless, these techniques failed to detect any form of transitions in video frames, thus extracting redundant frames. The authors in the work [20] presented a correlation approach to extract frames at low computational time. However, this was truncated because it failed to detect transitions leading to the extraction of redundant keyframes. Reference [21] presented an approach for detecting and removing blurry motion images from videos. This was achieved using an improved nearest neighbor clustering approach. However, using this approach, frames involved in shot transitions cannot be extracted as keyframes due to the lack of shot boundary detectors. To improve the previous research works, the authors in [22] presented review techniques that can enhance the keyframe extraction techniques, while [1] and [2] developed an improved technique using the k-means clustering and histogram-based approach with a view to eliminating the shortcomings of the earlier research works.

The present study used a histogram-based approach for shot boundary detection and k-means clustering for the extraction of a unique set of keyframes. However, similar frames were still categorized as keyframes with the use of the histogram-based approach. Aiming at the challenge that the histogram-based approach still leaves redundant frames, a statistical approach is proposed here. The statistical approach was combined with k-means clustering based on the work of [2] for the extraction of a unique set of keyframes. This is achieved by computing the statistical feature for frame difference, thus making it a frame filtering process. This method further reduces the amount of redundant extracted keyframes to the barest minimum. The developed method was implemented on the surveillance system in the Department of Computer Engineering (Compeng), ABU, Zaria.

The rest of this article is structured as follows: Section 2 presents the research methodology used. The results and their discussion are presented in Section 3, and conclusions are drawn in Section 4.

2 Proposed Methodology

This section presents the proposed keyframe enhancement strategy for summarizing surveillance video. A 64-bit, Core i7, 10^{th} generation desktop system was used as a third-party system to implement the proposed technique. The overall process to demonstrate our approach is summarized in the block diagram in Figure 1, with an explanation of each block. An implementation-based conceptual framework for the case study (Department of Computer Engineering, Ahmadu Bello University Zaria (Compeng)) is presented in Figure 2.

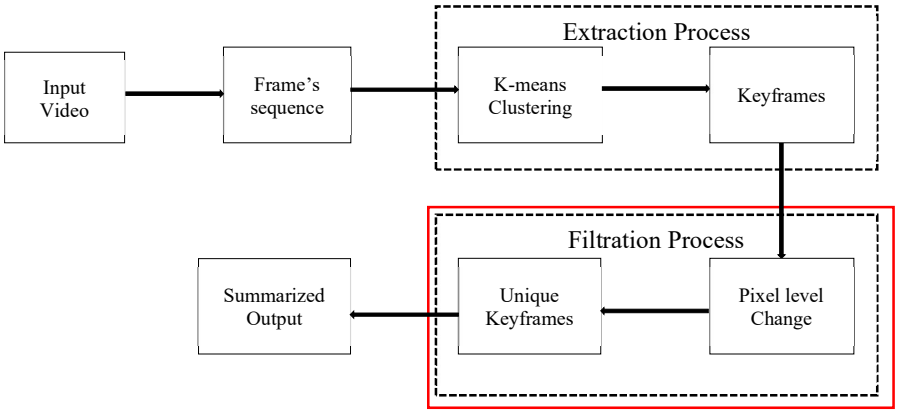

Figure 1 Proposed block diagram of the methodology.

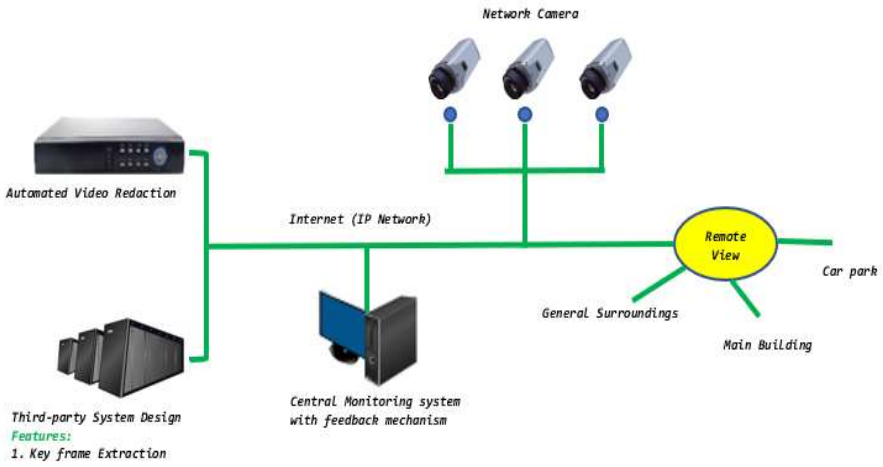

Figure 2 Implementation based conceptual framework diagram.

The steps in the proposed block diagram are explained below.

2.1 Video Acquisition

After the acquisition of a sample video from the databases used in [1] and [2], the total number of frames present in each of the collected sample videos were generated.

2.1.1 Input Video

The collected sample video was used as video input to the proposed system. Surveillance footage, movie clips, sports, and advertisements were among the sample videos obtained.

2.1.2 Frame Sequence

The next stage of our methodology involved the extraction of frames of each of the benchmark videos to a temporary storage location. Table 1 shows the total number of frames extracted from each of the videos used as input.

Table 1 Frames generated from sample videos.

S/n	Name	Format	Duration(sec)	Number of frames
1	Advert	AVI	29	746
2	Surveillance	MP4	7	228
3	Movie clip	MP4	8	254
4	Sport	MP4	5	168

2.2 Extraction Process

In this subsection, we will go over the step-by-step techniques for implementing the filtering process based on pixel level changes for the selection of a unique frame with a view to enhancing the keyframe extraction method. This is done with the adoption of the standard k-means clustering approach presented in [1]. Based on the video frames generated, the similarities between frames are obtained using the pixel distribution of the images between successive frames. After this is obtained, the k-means clustering approach is applied to the extracted frames. This implementation phase is divided into four stages: 1) feature selection, 2) clustering, 3) extraction of keyframes, and 4) summarization of video. The pseudocode for the filtering-based approach of the video summarization scheme is given in Algorithm 1, while a flowchart of the implementation process is presented in Figure 3.

As presented in the pseudocode and flowchart, a captured surveillance video file is used as the input and the expected output is a summarized video based on the extraction of a unique set of keyframes. The number of frames from the input video is extracted and stored in a temporary location. The temporary location is

then accessed, and the frames are sorted based on their numeric names. The difference between two consecutive frames is computed based on the pixel level changes of each of the frames and clustering of the frames is done with a view to achieving a summarized compressed video file for storage.

Algorithm 1: Video Summarization Based Compression Scheme

Input: E (Input Video)

Output: Video (summarized video based on keyframes)

Step1: Read the video

Step2: Extract the Frames from video

Step3: Sort Frames based on Numeric Names

Step4: Compute Frame difference

Step5: Initial centroid, Cj ← KMeans (dataset, k)

Step6: For Di ← (1<= i <=n)

Step7: Find the closest frame

Step8: Key ← (Di, Cj)

//end for

Step9: Repeat

Step10: For new cluster, Di ← (i <= Cj)

Step11: Frame stays in the cluster, Di ← i

//else

Step12: New cluster is form

//end for

Step13: Return assignment

Step14: Convert Clustered Frames to Video

Step15: Compare summarized frame with original video

Step16: Stop all process

Based on the presented pseudocode in Algorithm 1, a flowchart that further explains the process of the proposed enhanced keyframe extraction strategy is presented in Figure 3.

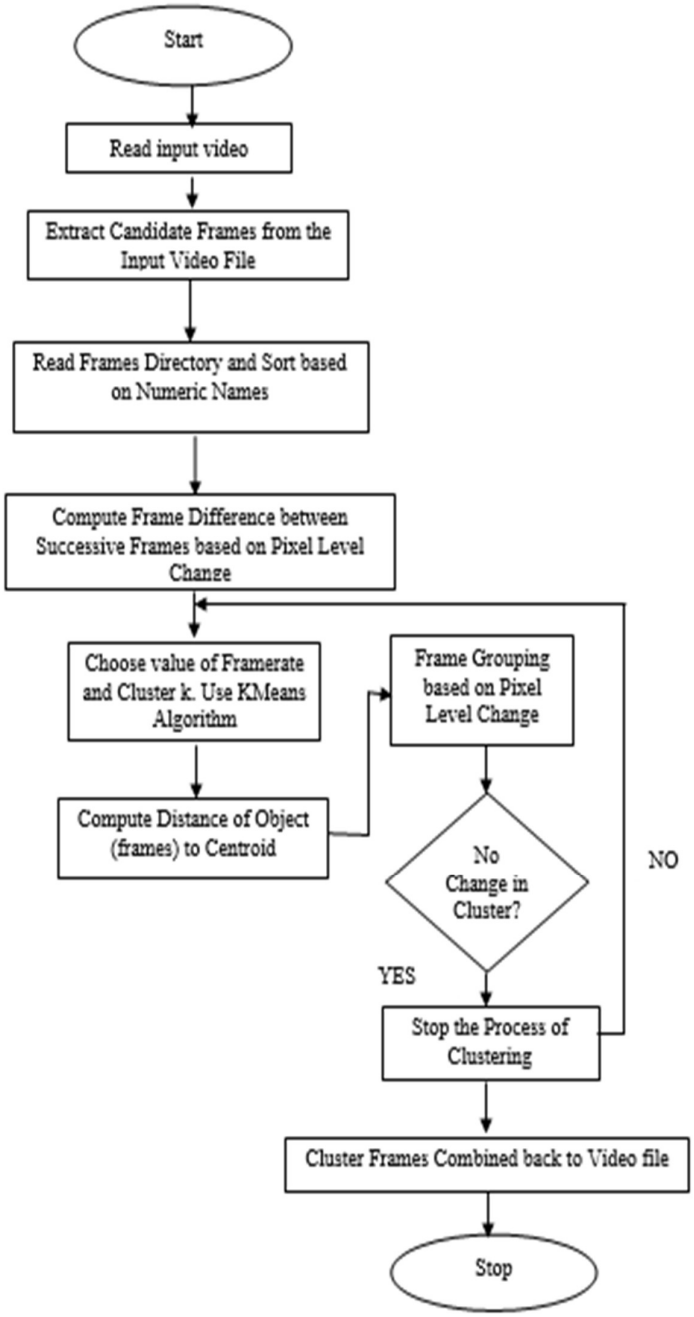

Figure 3 Flowchart of the implementation process.

2.2.1 Clustering Frames

With the use of the clustering technique, candidate frames are broken down into segments and each of the segments is represented by a value vector. The process involved in clustering is the process used in clustering the vector values. These vector values are clustered together to represent the segments of candidate similar frames based on pixel variation or pixel-level change. The number of segments is largely dependent on the frame rate (fps) used. The first step of the clustering techniques involves the definition of the number of clusters (*k*). In this study, the number of clusters used was 1. An objective function was defined, as presented in Eq. (1):

$$FC = \sum_{i=1}^{n}\sum_{k=1}^{k}(\| x^i - \upsilon_{ik} \|)^2 \tag{1}$$

where, FC is the clustered frame, and υ_{ik} is the centroid of cluster x^i.

$(\| x^i - v_{ik} \|)^2$ is the Euclidean distance between the cluster data points x^i and the cluster centroid v_{ik}. k is the number of clusters and n is the number of data points to be clustered. Therefore, to achieve clustered frames, the frame's data point x^i is assigned to the closest cluster based on the distance from the cluster's centroid. The averages of the data points associated with each cluster are used to re-calculate the new cluster center. This can be achieved with the use of Eq. (2):

$$\upsilon_{ik} = \frac{1}{n}\sum_{i=k}^{n} x^i \tag{2}$$

2.2.2 Keyframe Extraction

In order to determine which of the frames are keyframes to extract, the variation between the cluster centers and the frames within them is computed and the frame with minimum distance to the centroid is extracted as a keyframe.

2.2.3 Frame Difference Computation

This subsection presents the process of computing the frame difference between two consecutive frames in order to determine the pixel level change. First, the grayscale values of successive frames are taken to compute the differences between the frames. This is due to the fact that the grayscale representation of the original extracted frames requires less information for each pixel, thus making computation of frame differences easier. This is achieved via the creation of a function that first converts the colored images into their corresponding grayscale

images. Afterward, the grayscale images are transformed by their histogram representation to obtain the pixel level change. The total difference for each frame is computed using Eq. (3):

$$R(\mathrm{x}, x+1) = \sum_{n-1}^{n} \frac{[y(i,a) - y(i+1,a)]^2}{y(i,a)} \tag{3}$$

where $R(\mathrm{x}, x+1)$ denotes the frame difference, x^{th} is the current frame, and $x^{th}+1$ is the next frame.

Based on the frame difference computation, the unique key frames are determined.

2.2.4 Summarization of Video

With a view to obtaining a summarized video file, the highest valued cluster segment is chosen from each cluster. The final summarized video is generated by joining all the selected segments together. The compression ratio is then calculated to determine the rate of compression from the original video to the summarized video using Eq. (4):

$$R_C = \left[1 - \frac{E_{Orig}}{E_{Summ}}\right] x\ 100\% \tag{4}$$

where E_{Orig} are the frames extracted from the original video, in this case the benchmark video, and E_{Summ} are the keyframes that are extracted from the benchmark video.

2.3 Implementation of Case Study

As depicted in Figure 2, the keyframe extraction method was implemented on a third-party system. This third-party system was also used to access the installed Compeng IP camera via the system's web browser using a given IP address. The diagram in Figure 4 shows the accessed interface of the IP cameras. Typically, once a surveillance video has been recorded using an IP camera, the captured video is saved in the storage system. However, the saved video includes duplicate frames, which increases the storage space and video length.

To reduce the video length and storage space, the captured videos from the Compeng IP cameras is used as input for the enhanced keyframe extraction technique. With the aid of the enhanced keyframe extraction technique, the original captured video length can be reduced before saving on the storage system for future content retrieval.

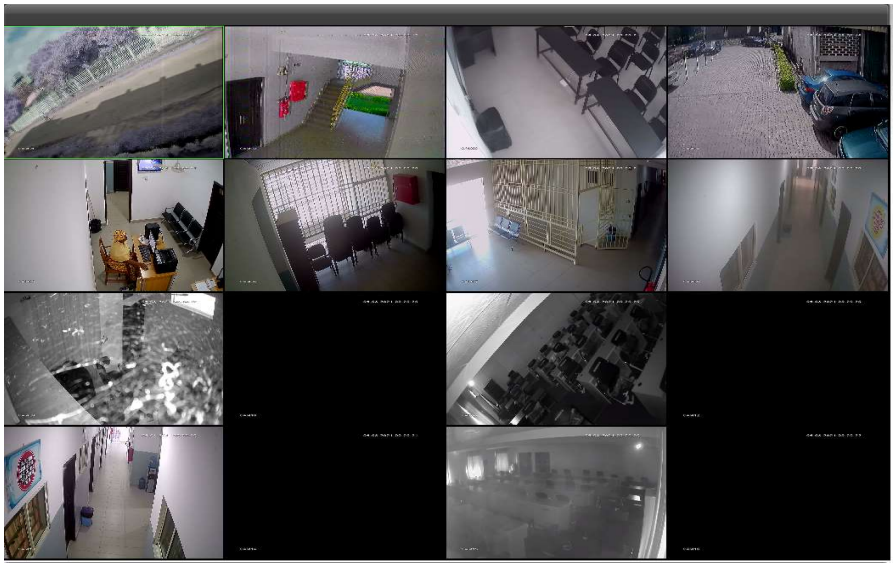

Figure 4 Interface of the Compeng IP camera.

3 Result and Discussions

The experimental results from the existing benchmark videos as well on real-life video data captured from an IP camera (Compeng video) are presented in this subsection. The performance of the proposed enhanced keyframe extraction strategy was compared to the results reported in [1] and [2].

3.1 Results of Extracting Keyframe

The proposed technique was tested on benchmark videos obtained from the database used in [1] and [2]. The benchmark videos were used as input and passed through the enhanced video summarization algorithm for the extraction of keyframes. The total video frames and keyframes collected from each of the videos used are shown in Table 2

Table 2 Extracted key frames.

Videos used	Total Frames	Keyframes		
		Proposed Scheme	[2]	[1]
Advert	746	13	173	47
Surveillance	229	11	101	11
Movie clip	254	12	40	18
Sport	168	11	35	23

The total number of frames in the original videos, as well as the matching keyframes recovered by both the developed and existing techniques, are presented in Table 2. In comparison to the existing technique, the current technique extracted a higher number of representative frames. This is due to the extraction of feature-related frames within the clusters. The proposed approach, on the other hand, was able to eliminate these unnecessary frames by clustering comparable frames and extracting the most representative ones as keyframes without compromising the frames' integrity, as presented in table 2. The clustered results of the devised system when tested on each of the videos, are depicted in Figures 5 to 8.

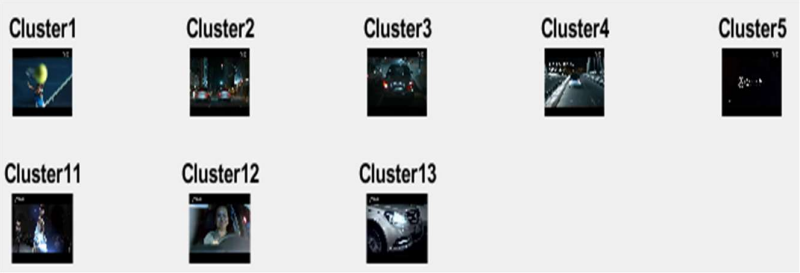

Figure 5 Sample of extracted clustered advertisement keyframes.

Samples of keyframes extracted by the proposed scheme from an advertisement video are shown in Figure 5. From the extracted sample of keyframes, it can be observed that no two or more feature-related frames were extracted from these keyframes, which represent the entire visual contents of the original video.

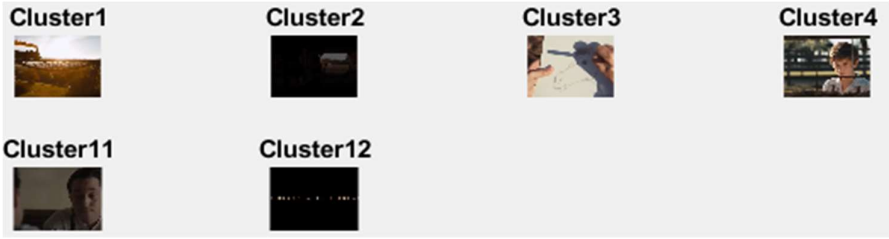

Figure 6 Sample of extracted clustered movie keyframes.

With the use of a movie clip as sample video for the developed technique, clustered keyframes recovered from the movie clip by the developed approach are shown in Figure 6. The literature explains that movies go through a number of video editing processes throughout production, which results in more redundant frames. Nonetheless, frames affected by these video editing effects are

removed, thereby presenting a unique set of keyframes as a representation of the entire video.

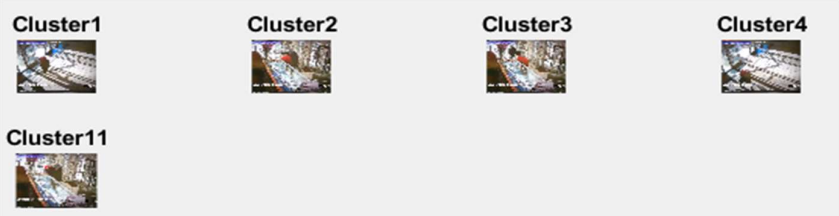

Figure 7 Sample of extracted clustered surveillance keyframes.

A sample of keyframes recovered from benchmark surveillance video is shown in Figure 7. Because surveillance video frames are taken in real time, they are usually unaffected by slow changes.

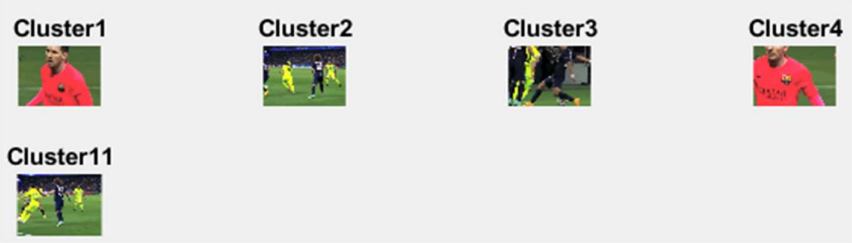

Figure 8 Sample of extracted clustered sport keyframes.

A sample of keyframes retrieved from the obtained benchmark video to represent an entire football match is shown in Figure 8. These extracted keyframes are representative of key activities during the match.

3.2 Comparison and Analysis of the Proposed Technique

To evaluate the performance of our developed keyframe extraction method, the compression ratio, video playtime were used as metrics. The results are presented in Tables 3 and 4, respectively.

Table 3 Compression ratio of developed and existing schemes.

Videos	Developed Scheme	[1]	[2]
Advert	97.3%	93.70%	76.81%
Surveillance	96.2%	95.20%	55.90%
Movie clip	93.5%	92.91%	84.25%
Sport	91.3%	86.71%	79.77%

Table 4 Compression ratio of developed and existing schemes.

Videos	Video Playtime in secs (Original Video)	Video Playtime in secs (Summarized Video)
Advert	29.840	8.133
Surveillance	7.706	7.133
Movie clip	8.512	8.363
Sport	5.851	5.851

As presented in Table 3, when compared to previous methodologies, such as [1] amongst others, video summarization utilizing the developed scheme gave a more reduced version of the full-length videos. The developed technique also did not degrade the frames during the extraction process, thereby producing a relatively good video quality. Table 3 also shows that the existing strategies had a lower compression ratio than the proposed scheme. This was due to the use of enhanced keyframes to extract several feature-related frames. A comparison of the developed scheme and the existing schemes with respect to the compression ratio is presented using a bar chart in Figure 9.

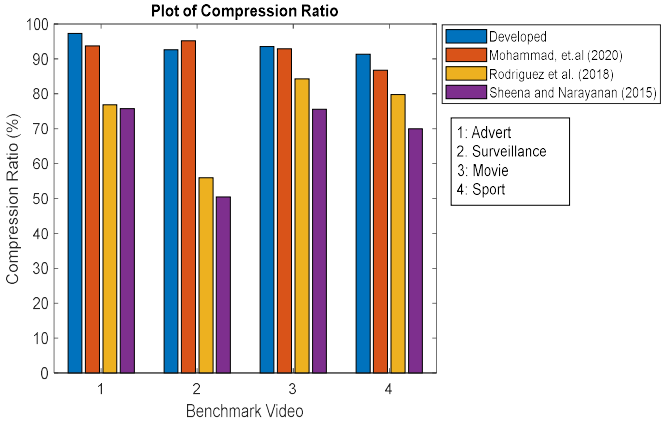

Figure 9 Compression ratio of various schemes.

The video playtime is the amount of time required from start to finish of a video. The video playtime for the developed and current techniques is shown in Table 4. From the results presented in Table 4, it can be observed that there was no significant improvement for videos with a short record time. Meanwhile, videos of longer length showed a significant difference between the video playtime of the original video and the summarized video. This can be largely attributed to the number of frames present in the video. The comparison of the developed scheme

and the existing schemes with respect to video playtime is presented using a bar chart in Figure 10

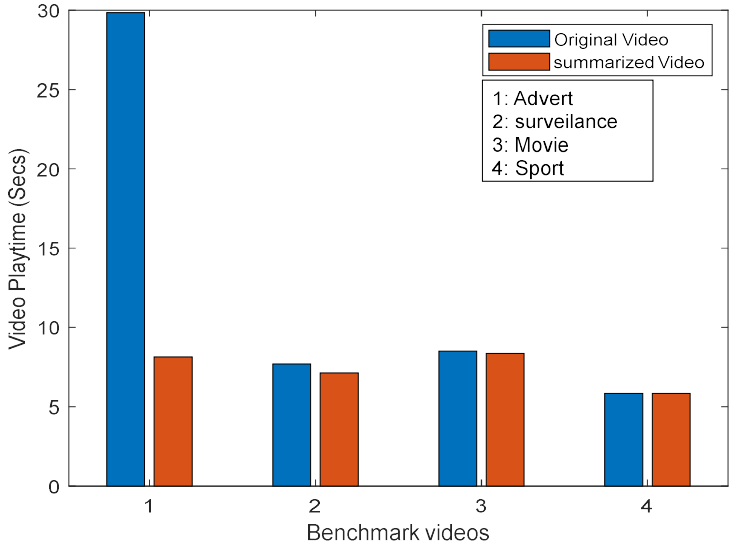

Figure 10 Comparison using video playtime.

With the aid of a third-party system, the Compeng IP cameras installed were accessed and real videos were obtained from the cameras. The real videos obtained were 36.352 secs and 12.629 secs in length on average. These obtained videos were used as sample input for the proposed enhanced keyframe extraction technique. The obtained values were recorded as presented in Table 5.

Table 5 Results obtained from real video.

	Video Playtime in secs (Original Video)	Video Playtime in secs (Summarized Video)	Compression Ratio
REAL 1	36.352	7.1167	95.5504
REAL 2	12.629	7.333	88.3117

Similar to the benchmark videos, it can be observed that there was no significant improvement for videos with a short record time (Real 2), while videos of longer length (Real 1) showed a significant difference between the video playtime of the original video and the summarized video. This can be largely attributed to the number of frames present in the video.

4 Conclusion

This paper proposed an enhanced keyframe extraction strategy for video summarization based on a statistical approach and k-means clustering. The strategies created were aimed at challenging existing systems that still leave redundant frames. Benchmark movies were retrieved from a database and fed into the proposed method, which summarized the video. The proposed method retrieved a set of unique keyframes while deleting duplicates, resulting in a shorter video playback time. As a result, the number of keyframes eliminated owing to visual editing effects such as progressive transitions, quick illuminance, and camera movement was decreased to the absolute minimum. The findings collected using various benchmarks and real videos of various playtimes show that the proposed technique greatly outperformed previous solutions.

Further work may consider the use of longer videos and of artificial intelligence to classify content for editing purposes.

References

[1] Muhammad, B., Sadiq, B., Umoh, I. & Bello-Salau, H., *A K-Means Clustering Approach for Extraction of Keyframes in Fast-Moving Videos*, International Journal of Information Processing and Communication (IJIPC), **9**(1&2), pp. 147-157, 2020.

[2] Rodriguez, J.M.D., Yao, P. & Wan, W., *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 (ICALIP), 2018.

[3] Kaur, P. & Kumar, R., *Analysis of Video Summarization Techniques*, International Journal for Research in Applied Science & Engineering Technology (IJRASET), **6**(01), 2018.

[4] Zedan, I.A., Elsayed, K.M. & Emary, E., *News Videos Segmentation Using Dominant Colors Representation*, Advances in Soft Computing and Machine Learning in Image Processing (pp. 89-109), Springer, 2018.

[5] Zhang, Q., Yu, S.-P., Zhou, D.-S. & Wei, X.-P., *An Efficient Method of Keyframe Extraction Based on a Cluster Algorithm,* Journal of Human Kinetics*,* **39**(1), pp. 5-14, 2013.

[6] Del Fabro, M. & Böszörmenyi, L., *State-of-the-art and Future Challenges in Video Scene Detection: A Survey*, Multimedia Systems, **19**(5), pp. 427-454, 2013.

[7] Li, X., Zhao, B. & Lu, X., *Key Frame Extraction in the Summary Space*. *IEEE Transactions on Cybernetics,* **48**(6), pp. 1923-1934, 2017.

[8] Paul, A., Milan, K., Kavitha, J., Rani, J. & Arockia, P.J., *Key-Frame Extraction Techniques: A Review*, Recent Patents on Computer Science. **11**(1), pp. 3-16, 2018.

[9] Asim, M., Almaadeed, N., Al-Máadeed, S., Bouridane, A. & Beghdadi, A. *A Key Frame based Video Summarization Using Color Features.* Paper presented at the 2018 Colour and Visual Computing Symposium (CVCS), Gjovik, Norway, pp. 1-6, 2018.

[10] Santini, S., *Who Needs Video Summarization Anyway?* Paper presented at the International Conference on Semantic Computing (ICSC), Irvine, CA, USA, pp. 177-184, 2007.

[11] Sheena, C.V. & Narayanan, N.J., *Key-Frame Extraction by Analysis of Histograms of Video Frames Using Statistical Methods*, Procedia Computer Science, **70**, pp. 36-40, 2015.

[12] Priya, G.L. & Domnic, S.J., *Shot Boundary-Based Keyframe Extraction for Video Summarisation*, International Journal of Computational Intelligence Studies. **3**(2-3), pp. 157-175, 2014.

[13] Yuan, J., Wang, H., Xiao, L., Zheng, W., Li, J. & Lin, F., *A Formal Study of Shot Boundary Detection*, IEEE transactions on circuits systems for video technology. **17**(2), pp. 168-186, 2007.

[14] Ejaz, N., Tariq & T. B., Baik, *Adaptive Key Frame Extraction for Video Summarization Using an Aggregation Mechanism*, Journal of Visual Communication Image Representation, **23**(7), pp. 1031-1040, 2012.

[15] Azhar, A.Z., Pramono, S. & & Supriyanto, E., *An Analysis of Quality of Service (QoS) in Live Video Streaming Using Evolved HSPA Network Media,* JAICT, **1**(1), pp.1-6, 2016.

[16] Kumar, K., Shrimankar, D.D. & Singh, N.J., *Eratosthenes Sieve-Based Key-Frame Extraction Technique for Event Summarization in Videos*. Multimedia Tools Applications, **77**(6), pp. 7383-7404, 2018.

[17] Gharbi, H., Bahroun, S. & Zagrouba, E., *A Novel Key Frame Extraction Approach for Video Summarization,* Paper presented at the 11th Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP), Rome, Italy, pp. 148-155, 2016.

[18] Sujatha, C. & Mudenagudi, U., *A Study on Keyframe Extraction Methods for Video Summary.* Paper presented at the 2011 International Conference on Computational Intelligence and Communication Networks, Gwalior, India, pp. 73-77, 2011.

[19] Ali, I.H. & Al–Fatlawi, T.T., *A Proposed Method for Key Frame Extraction*, International Journal of Engineering Technology, **8**(1.5), pp. 509-512, 2019.

[20] Satpute, A.M. & Khandarkar, K.R., *Video Summarization by Removing Duplicate Frames from Surveillance Video Using Keyframe Extraction*, International Journal of Innovative Research in Computer and

Communication Engineering, pp. 8501-8509, 2017. DOI:10.15680/IJIRCCE.2017. 050426.

[21] Lv, C. & Huang, Y., *Effective Keyframe Extraction from Personal Video by Using Nearest Neighbor Clustering.* Paper presented at the 2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Beijing, China, pp.1-4, 2018.

[22] Sadiq, B.O, Muhammad, B, Abdullahi, M.N, Onuh, G., Ali, A.M. & Babatunde, A.E. *Keyframe Extraction Techniques: A Review*, Journal of Electrical Engineering (ELEKTRIKA), **19**(3), pp. 54-60, 2020.