A Study on various Key Frame Detection Techniques in order to Develop an Efficient Method

Neha Katre, Meera Narvekar, Chirag Jain, Ishika Chokshi, Chirag Jagad

Dwarkadas J. Sanghvi College of Engineering, Mumbai, India
neha.mendjoge@djsce.ac.in, meera.narvekar@djsce.ac.in,
chiragjain55551@gmail.com, ishika.c1671@gmail.com,
chiragjagad08@gmail.com

Abstract: *Globally, enormous amounts of text, photos, and blog content are produced. Due to improvements in network designs, high storage availability, and widespread use of digital cameras, video processing has become increasingly important. High-definition digital video has rapidly replaced dull text material, nevertheless, as a result of the extraordinary advancements in multimedia and Internet technology. This has made it one of the primary means through which people disseminate information. Each video consists of a large number of frames that are crucial pieces of information. To process the video, it is essential to extract these frames. Identifying key frames from every frame that contain distinctive aspects of the video aids in the development of cutting-edge technologies that support a variety of video analytics applications, such as object and anomaly detection. In this study, a comparison of traditional key-frame extraction methods, viz., Clustering, Motion Analysis, and Shot-Boundary based, as well as deep learning key-frame extraction methods is done. In this study, existing deep learning key-frame extraction approaches are compared to more established key-frame extraction techniques like clustering, motion analysis, and shot-boundary based methods. The aforementioned techniques have been implemented, and a thorough comparison study has been provided along with a list of their benefits and drawbacks.*

Keywords: Key-Frame, Object Detection, Video Analytics, Clustering, Histogram, Motion Analysis, Pixel Based, Edge Difference

1. Introduction

Video data can be utilized for monitoring, analysis, and reporting due to the ongoing advancement of technology. Different machine learning techniques can be used to analyze video data and identify various temporal and spatial events that take place. Systems for monitoring and surveillance can benefit greatly from video analytics. It has numerous commercial uses, including intrusion management, people counting, facial recognition, and anomaly detection. It has various industrial applications like anomaly detection, people counting, facial recognition, intrusion management, etc.

Anomaly detection with video analytics necessitates real-time video data analysis and insight delivery. Rapid monitoring, analysis, and reporting are necessary for an anomaly like an arson in order to take quick action and prevent any unfavorable outcomes. The processing of enormous amounts of video footage and live video surveillance, however, present significant obstacles to real-time processing. Depending on the model

type, the various security cameras operate at varied frames per second (FPS). An ideal frame rate for surveillance cameras is 15 to 30 frames per second. At 30 frames per second, video is most effective at capturing intruders or threats. The standard for business as of 2019 is 15 FPS. However, 30 FPS is progressively replacing 24 FPS as the industry standard as high-quality security cameras become more affordable [1]. So, there is a need for real-time processing speed to meet the frame rate of these video cameras.

Key-frames are described as a group of frames that, with the fewest possible frames, describe the main characteristics of the entire video and convey its main ideas and information. Real-time video analysis presents a number of difficulties, such as time restrictions and resource shortages. Additionally, it is meaningless because some of the frames don't include important data. Therefore, rather than examining every frame, it is necessary to pinpoint the ones that can capture the key details and information of the video. Key-frames are those particular frames. A keyframe is a specific point in time in a sequence of frames that is marked to indicate a significant change in the value of an animated parameter such as position, orientation, size, color, or opacity for an object or character. After the extraction of key-frames, instead of analyzing the contents of all video frames, only the key frame images are analyzed. Since the video is compressed into fewer frames, the time and resources required for processing them are reduced. Also, key-frames are analyzed after their extraction process, so the algorithm for extraction should not be overly complex or time-consuming to achieve real-time processing [2].

There are different types of techniques and algorithms available to extract key-frames from videos. These techniques are majorly divided into four categories namely - Shot-Boundary based, Motion-analysis based, Clustering based, and Deep Learning based. A detailed introduction of these categories is provided. Further, an overview of various techniques within each of these categories has been discussed. Lastly, a comparative analysis of these methods based on applications is discussed to identify the techniques best suited for key-frames detection.

2. Categorising Key-Frame Detection Techniques

A key frame serves as an example and includes all the data from the video collection. Traditional key frame extraction methods eliminate comparable and pointless frames from videos without sacrificing the semantic information included in the visual material. These methods either take a single video as input or one that has been divided into shots utilizing shot boundary detection algorithms. Relevant and necessary information retrieval is a crucial task in video analysis and processing and can be accomplished with the aid of key-frame detection [3]. This is because it can be difficult to analyze a large video in a short period of time without losing its semantic intricacies.

Figure 1 shows the categorisation of various key-frame detection techniques, viz., clustering based methods, motion analysis based methods, shot boundary based methods and methods using deep learning.

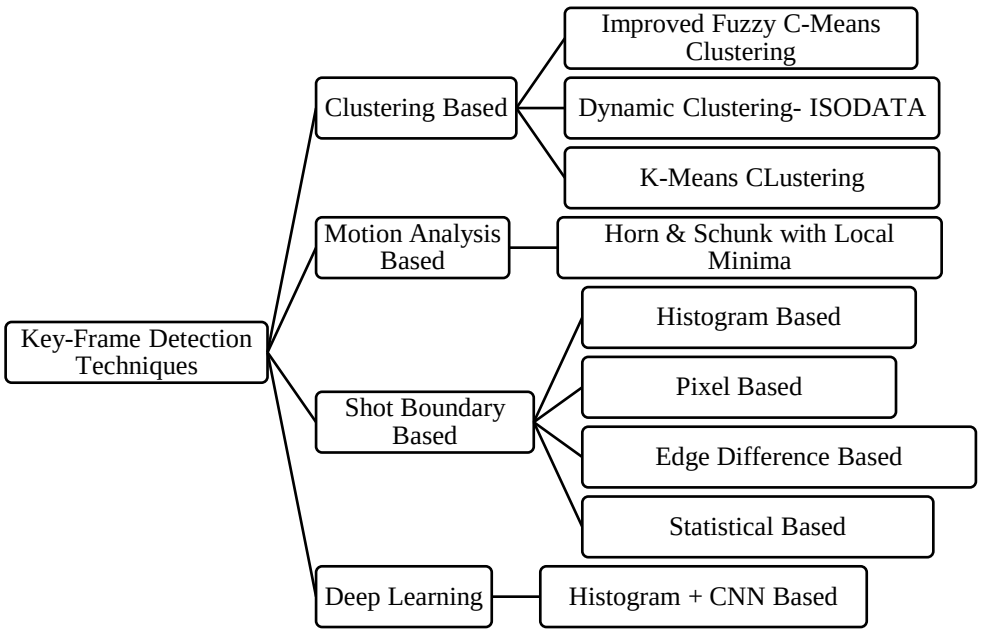

Figure 1: Key-Frame Detection Techniques

2.1. Clustering Based Method

Key frames are extracted from each set of frames in a video using different methods after the entire video has been divided into groups of frames with comparable low-level properties using clustering-based key frame extraction algorithms. This method can assist in cutting down on extraneous frames and works well with videos that have comparable content. However, this method completely disregards the significance of time sequence. During the clustering process, video frames will be scrambled with the time sequence, losing important time information. Therefore, this method is not practical for fields where real-time needs are crucial [4].

2.2. Motion Analysis Based Method

Key frame extraction algorithms based on motion analysis use motion features to identify crucial frames so that the original video can be compressed without missing crucial activities. The visual changes in the video are described by motion characteristics using temporal differences. Utilizing methods for estimating motion, such as optical flow, it can be estimated. The motion of image intensities, which can be attributed to the mobility of objects in the scene, is estimated using optical flow. The Farneback-based dense optical flow, the Lucas-Kanade-based spare optical flow, etc. are some examples of optical flow techniques [5].

2.3. Shot-Boundary Based Method

Visual dissimilarity is used by shot boundary-based key frame extraction techniques to locate key frames. These differences, which might be rapid or gradual, are brought on by the transitions in video. This method's crucial stage is shot boundary identification, which can be calculated using a variety of techniques such pixel differences, statistical differences, histogram comparisons, edge differences, etc. Videos that are not structured respond very poorly to this strategy. For structured video, it generates comprehensive results with a relatively constant shot change rate [6].

2.4. Deep Learning Based Method

In deep learning-based key frame extraction techniques, convolution neural networks are used to extract the features of the frames in a video. The process takes a long time and uses a lot of memory because the neural network is supposed to process the entire video. Researchers coupled this strategy with additional key frame extraction methods to achieve better results. The primary frame extraction problems involving object recognition, such as face recognition, pedestrian detection, video segmentation, etc., are best served by this method. This approach is most beneficial for key frame extraction problems involving object recognition, such as face recognition, pedestrian detection, video segmentation, etc.

3. Review of Prominent Key-Frame Detection Techniques

This section contains a review of some current conventional key-frame detection techniques.

3.1. Clustering Based Method

3.1.1. Clustering Based on Density Peak: The HSV histogram is used in a density peak clustering method for video key frame extraction that reduces the computational complexity by converting high-dimensional abstract video image material into a measurable two-dimensional input matrix. The density peak clustering algorithm is used to cluster this low-dimensional data in order to identify the cluster nodes. The disadvantage of the conventional technique, which extracts a set number of key frames, is overcome by combining these results to obtain a different number of key frames. This method can efficiently extract important frames while integrating the characteristics of video footage. The extracted key frames can produce clusters of any shape without the need to artificially build them up, have little redundant information, reflect the main information of a video more effectively, and are noise resistant. The extracted key frames can better reflect the primary information of a video, have minimal redundancy, are noise resistant, and can create clusters of any shape without the need to artificially build up the beginning parameters [7].

3.1.2. Clustering Based on Improved Fuzzy C-means Clustering: A more advanced key-frame extraction strategy based on fuzzy C-means clustering has been presented to overcome problems with conventional clustering algorithms. The shots are divided into several sub-shots by using the color feature information in the video frames, followed by the video sequence clustering technique to determine the center value of various classes and the membership degree of each frame relative to the classes. Based on the consistency of the contents in the sub-shots and the significant differences between various classes, as well as the fact that the value of the maximum image entropy corresponds to the maximum amount of information in the information theory, the maximum entropy frame value is extracted as the keyframe from each class. This method selects the frame with the largest entropy as the key-frame from each class in accordance with the properties that after clustering the content within each class is relatively consistent, the differences between two classes are greater, and the information theory fact that the larger the image entropy is, the more information the image contains. The method overcomes the drawbacks of traditional key-frame method extraction techniques where the keyframe numbers are fixed [8].

3.1.3. Dynamic Clustering Algorithm – ISODATA: The creation of an automated key-frame extraction technique based on the adaptive threshold is the primary focus of this research. The nearby frames were combined into a motion sequence in order to determine the thresholds needed for the second stage. A modified ISODATA approach was used to locate the proper key-frames. One of the most used clustering methods in image processing for geosciences and remote sensing applications is ISODATA. The sample is divided into two groups using similarity distances between successive frames in a motion sequence to give the thresholds needed for the second stage of clustering. The key-frames may then be chosen from the frames that are closest to the center of the final clustering using the enhanced ISODATA, which can also be utilized for dynamic clustering. The mean absolute errors from the original data and the rebuilt data, along with the reconstructed motion, were used to establish a relevant technique to compare findings to those of two earlier methods [9].

3.1.4. K-Means Clustering: Using k-means clustering and the mean squared error method, this research suggested and implemented a novel robust key frame extraction and foreground isolation methodology for variable frame rate films. The video's foreground objects have been isolated while the noise created during recording has been removed. By using this method, the flickering of the frames caused by a changing frame rate in a recorded video is considerably reduced. K-means clustering is used in Apache's Hadoop architecture to accelerate computing results. The results of the method have been compared to those obtained using comparable methodologies, such as the Gaussian Mixture Model, and it has been found that the results of the method are superior. Once the background has been modelled to separate the foreground components, the video frames are subtracted from it. To further minimize noise and enhance the clarity of the foreground objects, a bilateral filter is applied to the frame. The background noise is completely eliminated by color quantization using k-means clustering once the foreground mask has been obtained and denoised. After clustering the foreground masks, the frames that cause flicker as a result of the frame rate appear as completely black frames. After sorting these frames according to the number of black pixels, a mean squared error comparison is made between the foreground mask frames and the black frames. A key frame is one that is distinctive [10].

3.2. Motion Analysis Based Method

3.2.1. Horn and Schunck's Algorithm with Local Minima: The motion-based method is used to extract key frames from the video. The optical flow between frames in a shot is calculated and frames that are at local minima of the motion of the shot are considered keyframes [11]. The algorithm involves two steps.

1. Horn and Schunck's algorithm is used to calculate optical flow. The sum of the magnitudes of the components of optical flow at each pixel as a motion metric *M(t)* for frame *t* is computed as:

$$M(t) = \sum_{i=1}^{k} \sum_{j=1}^{l} |OF_x(i,j,t)| + |OF_y(i,j,t)| \qquad (1)$$

where $OF_x(i, j, t)$ is the x component of optical flow at pixel (i, j) in frame t, and similarly for y component.

2. The second step involves identifying local minimas. The graph between *M(t)* vs *t* is plotted and it identifies two local maxima m_1 and m_2 such that the value at m_2 varies by at least N% from the *M(t)* value for m_1.

The local minimum of M(t) between these local maxima is selected as a key frame. An important advantage of this algorithm is that it does not assume a fixed number of keyframes per shot. Instead, it selects the number of keyframes appropriate to the composition of the shot [12].

3.3. Shot-Boundary Based Method

3.3.1. Histogram Based Method: Key frames are extracted utilizing the histogram-based method employing the threshold and Difference of Histogram technique. The threshold is determined by the difference between the histograms. All of the target video's frames are initially extracted and saved in a directory. A comparable grayscale image is created for each frame. Now, the entire video is iterated, and at each stage, the histogram difference between two subsequent grayscale photos is calculated, returning the total of all the histogram's components. The mean and standard deviation are calculated after every iteration. The threshold is finally computed using the mean and standard deviation values. Now, this threshold value is compared to the sum value subtracted from the previously calculated histogram difference at each iteration's step. The second frame is regarded as a key frame if the total of the difference histogram values for two subsequent frames is more than the threshold value. In this way, after an entire iteration, a set of key frames from the entire video is obtained based on the threshold value [13][14].

3.3.2. Pixel Based Method: In order to determine key frames, the pixel difference method takes into account the change in pixels between subsequent frames. The percentage of altered pixels is taken into account, or the pixel difference between two successive frames is determined. The pair-wise comparison strategy is used in [15] and counts the number of pixels whose value changes by a predetermined threshold. Segment boundaries are identified if a sizable number of pixels shift, larger than threshold T. This technique reacts very quickly to camera motion [13].

3.3.3. Edge Difference Based Method: A keyframe extraction technique based on edge differences depends on the contents and changes in the contents of the frames. Since the edge depends on the content, edge difference is taken into account [6]. To detect changes in the content of the frames, this approach maps the edge pixels of one frame to nearby edge pixels of the following frame. In order to determine the difference between edge pixels in two frames, Canny Edge detection is employed. Every time the video is iterated, the current frame and the frame before it are transformed into the equivalent grayscale image, and the edge difference between them is determined using the Canny edge detector. This procedure is carried out for each iteration, and the difference value is kept. The differences between each successive frame are totaled up at the end of the process and used to determine the mean and standard deviation. Utilizing mean and standard deviation values, the threshold is calculated. Keyframes are differences that are greater than the threshold value [16].

3.3.4. Statistical Based Method: Utilizing statistical differences, statistically based algorithms choose keyframes that provide the desired contextual information. In order to decrease the computer resources and processing time needed to analyze the massive amount of data in aerial surveillance photography, the method detects keyframes using the statistical difference method [6]. In this method, frames are divided into small sections, and for each succeeding frame, statistical features like the mean and standard deviation of each pixel inside these regions are computed. The adaptive threshold is calculated using mean and standard deviation. Keyframes are those frames with statistical differences over the adaptive threshold. Since frames are divided into small parts for the computation of statistical differences, this approach is advantageous in detecting frames with slight content changes. So, this technique is noise-tolerant. However, because all of the frames are divided and examined, this procedure might be cumbersome due to complex statistical calculations [17].

3.4. Deep Learning Based Method

To extract the pertinent key frame from the video, a combination of deep learning and histogram approaches is applied. After being normalized, the histograms are contrasted with those of the subsequent frames. A list of keyframes is created by comparing the value of the histogram with the threshold, and this list is then compared to the actual list of frames to remove superfluous keyframes. A convolutional neural network is used for feature extraction and classification in the following stage. During the classification phase, recovered unique keyframes are used as testing queries, and the convolutional neural network feature extraction module is used to extract input frame features. To create the best match frame, which is regarded as key in the output as a frame index number, the learnt features are matched with various keyframe features [3]. Another strategy is to extract video frame attributes using a deep learning approach, more especially an auto-encoder network. The method automatically removes the unnecessary portions of the video, keeping just the video clips and key frames that contain useful information by computing the difference between the features of consecutive frames. Utilizing cutting-edge methods to capture spatial and temporal relationships between frames, such as multi-scale feature extraction, recurrent neural networks, and convolutional neural networks, can improve this strategy even further [18].

4. Evaluation Metrics

The performance metrics that were utilized to evaluate the suggested method are summarized in this section. Since standard performance metrics or ground truth are not available for key frame extraction methods on video surveillance to detect abnormalities, it is challenging to compare the results with previously completed research.

Standard performance criteria for keyframe extraction techniques are not established since there is no literature that provides the formal definition of "keyframe." The attributes that keyframes must have vary depending on the application. Since there is no literature that offers a formal definition of "keyframe," there are no recognized standard performance requirements for keyframe extraction algorithms. Depending on the application, different keyframes are required to have different properties. Although the keyframes' information content is crucial for content-based video retrieval, the compression ratio is the essential aspect of video compression [19].

The goal of this study is to recreate the video with the fewest keyframes possible. But it's also critical to pick frames that include important information about anomalies. In order to effectively summarize the entire video, the fewest possible frames must be used. The key frame extraction approach should also deliver important frames as quickly as practical in order to detect irregularities in real-time. To satisfy this need, fidelity and compression ratio are computed as evaluation metrics. While fidelity provides the exactness of keyframes, compression ratio determines compactness [20]. The compression ratio is calculated as in Eq. (2)

$$C_{ratio} = 1 - \frac{N_{kf}}{N_{vf}} \qquad (2)$$

where N_{kf} are the key frames detected and N_{vf} are the total number of frames in a video.

Keyframe accuracy is provided by fidelity. Semi-Hausdorff distance is used in the computation of fidelity metrics. Euclidian Distance is calculated between each keyframe and each frame of the test video. The distance between that keyframe and the original video is then set to the minimal distance. The greatest distance thus achieved is denoted as $Dist\ (V_{seq}, Key_f)$ and is considered as the distance between a set of keyframes and a set of original frames. The largest distance that may be calculated between a keyframe and the original video frames is referred to as the maximum dissimilarity measure and is denoted by the Max_{Dist}. Fidelity (Fi) between original videos V_{seq} and extracted key frames Key_f is given in Eq. (3)

$$Fi(V_{seq}, Key_f) = Max_{Dist} - Dist(V_{seq}, Key_f) \qquad (3)$$

A high-fidelity rating denotes an accurate reproduction of the original video in fewer frames. Also, a higher compression ratio means that the video is more efficiently represented in a compact form. Since high fidelity and high compression ratio are desired, a new performance metric termed CRNF is created and is defined in Eq. (4), where NF stands for normalized fidelity and CF for compression ratio.

$$CRNF = (CR) * (NF) \qquad (4)$$

5. Analysis of Reviewed Techniques

Based on previous research, a comprehensive review and analysis of Key Frame Detection techniques is presented. Table 1 provides a detailed overview of various techniques along with some of the concerned disadvantages of the key-frame detection techniques reviewed.

Table 1. Key Frame Detection Techniques

Category	Method	Summarization	Disadvantage
Clustering	Density Peak Clustering Algorithm	HSV histogram turns high-dimensional abstract video image content into a quantifiable two-dimensional input matrix, this data is then clustered using the density peak clustering algorithm, to find the	The technique is inconvenient for the domains where real-time requirements are essential as it ignores

		cluster centres. These results are further combined to get a different number of key frames.	the importance of time sequence completely.
	Fuzzy C-means Clustering Algorithm	The information theory fact that the larger the image entropy is, the more information the image contains, this method chooses the frame with the largest entropy as the key-frame from each cluster.	
	ISODATA Algorithm	An automated key-frame extraction method based on the adaptive threshold. Using similarity distances between consecutive frames in a motion sequence, the sample is grouped into two groups to provide the thresholds. The key-frames are the ones closest to the centre of the clustering using the enhanced ISODATA.	
	K-Means Algorithm & Mean Square Error	First noise is removed, and video frames are subtracted from the background, the foreground mask is acquired, de-noised, and then color quantization using k-means clustering is carried out. All these frames are sorted according to how many black pixels are present in them. The distinctive one is a key frame. Then mean-square error comparison of the foreground masks frames and black frames is done.	
Motion Analysis	Horn and Schunk's Algorithm with Local Minima	Horn and Schunck's algorithm is used to calculate optical flow. Then the local minima are identified. The frames that are at local minima of the motion of the shot are considered key-frames	Computationally complex, highly noise-sensitive
Shot Boundary	Histogram	Extracts key frames using threshold and Difference of Histogram technique. Histogram difference is used for calculating the threshold.	Entirely loses the location information. For example, two images with similar histograms may have completely different content.
	Pixel	Considers the change of pixels between successive frames to identify key frames. Either pixel	This method is extremely sensitive to camera motion

		difference between two successive frames is calculated or the percentage of pixels which are varied is considered.	
	Edge Difference	Considers edge difference since the edge is content dependent. Edge pixels of one frame are mapped to nearby edge pixels of the next frame to detect the changes in the content of the frames.	Canny edge detector is not suitable when there is noise in the form of snow or rain. Cannot detect small objects
	Statistical	Selects keyframes that contain the desired contextual information using statistical differences. Frames having statistical differences greater than the adaptive threshold are considered keyframes.	The process is slow due to intricate statistical computation.
Deep Learning	CNN	The neural network is used to extract information from video frames. To select key frames, the feature differences across video frames are compared and the frames are either categorized or clustered.	Requires a lot of memory and is very time consuming

To detect and analyze keyframes in real time, the keyframe detection techniques need to be fast and accurate. The selected techniques should not be complex and time-consuming. From Table 1, it can be inferred that clustering-based key frame extraction techniques are not suitable for real-time purposes since they completely lose the significance of time sequence.

Statistical-based and Pixel Based techniques from the Shot Boundary-based category are also not suitable for detecting keyframes in real-time. In the Statistical Based approach, a complex computation is carried out by dividing each frame into multiple small regions. This consumes a lot of processing time and hence cannot provide keyframes in real-time. In Pixel based technique, each pixel of two consecutive frames is compared. This process takes a lot of time. Also, this technique is extremely sensitive to camera motion. Hence it cannot be used for the real-world use case of detecting anomalies from cameras.

So, after a comprehensive comparative review, four Key Frame Detection techniques, namely, Histogram Based, Motion Analysis Based, Canny Edge Difference-based, and Deep Learning based methods are implemented to further analyze them based on evaluation metrics and execution time to observe which method will be the most suitable for our application of detecting anomalies in real-time from surveillance videos.

The aforementioned techniques were tested on numerous surveillance videos which were captured by CCTV cameras. Some of the characteristics, i.e., the duration and number of frames of these videos are shown in Table 2.

Table 2. Details of Videos

Video ID	Title	Length	No of Frames
1	Fire due to car-truck collision	10s	319
2	Fire due to oil truck collision	6s	151
3	Fire caused in running electric vehicle	7s	181

The results of evaluation metrics and execution time are obtained by assessing the predefined techniques on the above-mentioned videos. Table 3 and Table 4 demonstrate the results of the fidelity measure and compression ratio of the four techniques across 3 different videos of anomalies. For analysis purposes, we have considered average values obtained for each technique across different videos.

From Table 3, it is clearly inferred that the Deep Learning-based method has a very high-fidelity measure and outperforms all other techniques in providing a concise summary of the full video using the fewest number of frames. So, Deep Learning based method can be used to produce an accurate reproduction of the original video.

Table 3. Fidelity Measure

ID	Histogram-Based	Canny Edge Based	Deep Learning	Optical Flow
1	805.31	1763.56	40289.62	1012.74
2	5520.80	8676.65	250635.01	10475.47
3	10449.41	32954.26	179402.985	12204.67
Average	**5591.84**	**14464.823**	**156775.87**	**7897.63**

The Deep Learning-based method has the highest compression ratio followed by the Canny Edge-based method and Histogram-based method as noticed in Table 4. The Optical Flow method has a very low compression ratio as compared to the other techniques.

Table 4. Compression Ratio

ID	Histogram Based	Canny Edge Based	Deep Learning	Optical Flow
1	0.78	0.88	0.98	0.66
2	0.801	0.814	0.94	0.69
3	0.89	0.94	0.97	0.67
Average	**0.82**	**0.87**	**0.96**	**0.67**

The figure given below gives the comparative analysis of the four methods mentioned above in terms of average CRNF which is calculated using Eq. (3). For CRNF, normalized fidelity is used. In Fig. 2, we can see that the Deep learning method has the highest CRNF value over different videos. The other three techniques provide poor CRNF value compared with Deep Learning based method.

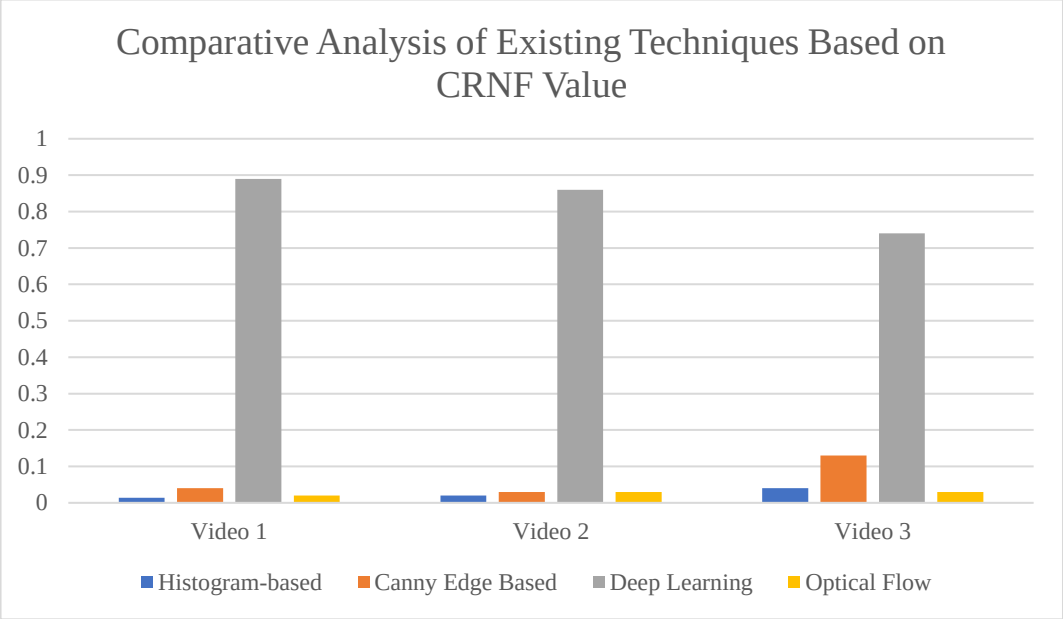

Figure 2: Comparative Analysis of Existing Techniques Based on CRNF Value

Table 5 provides information about the execution time taken by the four techniques for extracting keyframes. To detect anomalies in real time, the execution time for extracting key frames needs to be minimum.

Table 5. Execution Time

ID	Histogram Based	Canny Edge Based	Deep Learning	Optical Flow
1	5s	2s	32s	18s
2	6s	4s	34s	40s
3	6s	4s	35s	38s
Average	**6s**	**3s**	**34s**	**32s**

Figure 3 gives the comparative analysis based on average execution time taken by aforementioned techniques to extract key frames. From figure 3, it can be ascertained that the Deep Learning based method and Optical Flow based method takes a lot of processing time. The Canny edge detection method takes the least execution time followed by the Histogram based method which requires relatively low time of execution.

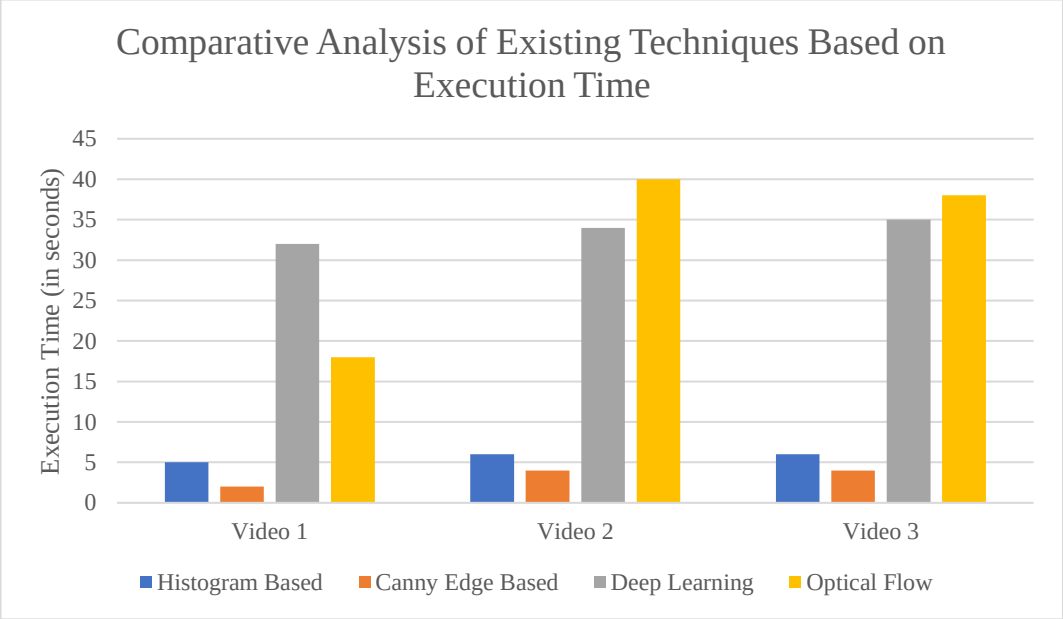

Figure 3: Comparative Analysis of Existing Techniques Based on Execution Time

6. Conclusion

According to the analysis of numerous key-frame detection algorithms, clustering-based methods are inappropriate for real-time applications because they disregard the importance of time. For real-time applications, the approaches must be rapid and computationally efficient. Motion analysis methods have been shown to be difficult and time-consuming. The histogram-based technique is undesirable because two frames with different contents could have the same histogram. Canny-based key frame extraction algorithms appear to be quite beneficial in real-time analysis, as may be deduced from the discussions and given the speed of execution. Canny edge detection works better than other algorithms in terms of results, but its noise sensitivity makes it less effective. Deep Learning-based approaches extract distinctive and few critical frames, which is demonstrated by their higher CRNF values, but they are complicated and time-consuming. A technique that will require less processing time but extract more distinct frames is therefore required for further analysis.

Conflict of Interest: On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

[1]. Security Camera Frame Rate (FPS) Explained (no date) ProtectFind. Security Cameras in Australia – Buyers Guide. Available at: https://protectfind.com.au/security-cameras/frame-rate-fps/#:~:text=A%20good%20frame%20rate%20for%20security%20cameras%20is%20around%2015,15%20FPS%20as%20of%202019. (Accessed: February 25, 2023).

[2]. K. Agrawal et al., "Automatic Traffic Accident Detection System Using ResNet and SVM," 2020 Fifth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), Bangalore, India, 2020, pp. 71-76, doi: 10.1109/ICRCICN50933.2020.9296156.

[3]. U. Gawande, K. Hajari, and Y. Golhar, “Deep Learning Approach to Key Frame Detection in Human Action Videos,” Recent Trends in Computational Intelligence, May 2020, doi: 10.5772/intechopen.91188.

[4]. Sahu, K., and S. Verma. "Key frame extraction from video sequence: a survey." International Research Journal of Engineering and Technology (IRJET) 4, no. 05 (2017).
[5]. Mizher, Manar Abduljabbar Ahmad, Mei Choo Ang, Siti Norul Huda Sheikh Abdullah, and Kok Weng Ng. "Action key frames extraction using l1-norm and accumulative optical flow for compact video shot summarisation." In Advances in Visual Informatics: 5th International Visual Informatics Conference, IVIC 2017, Bangi, Malaysia, November 28–30, 2017, Proceedings 5, pp. 364-375. Springer International Publishing, 2017. https://doi.org/10.1007/978-3-319-70010-6_34
[6]. Pal, Gautam, Dwijen Rudrapaul, Suvojit Acharjee, Ruben Ray, Sayan Chakraborty, and Nilanjan Dey. "Video shot boundary detection: a review." In Emerging ICT for Bridging the Future-Proceedings of the 49th Annual Convention of the Computer Society of India CSI Volume 2, pp. 119-127. Springer International Publishing, 2015. https://doi.org/10.1007/978-3-319-13731-5_14
[7]. Zhao, Hong, Tao Wang, and Xiangyan Zeng. "A clustering algorithm for key frame extraction based on density peak." Journal of Computer and Communications 6, no. 12 (2018): 118-128. https://doi.org/10.4236/jcc.2018.612012
[8]. Rong Pan, Yumin Tian and Zhong Wang, "Key-frame extraction based on clustering," 2010 IEEE International Conference on Progress in Informatics and Computing, Shanghai, 2010, pp. 867-871, doi: 10.1109/PIC.2010.5687901.
[9]. Zhang Q, Yu SP, Zhou DS, Wei XP. An efficient method of key-frame extraction based on a cluster algorithm. J Hum Kinet. 2013 Dec 31;39:5-13. doi: 10.2478/hukin-2013-0063. PMID: 24511336; PMCID: PMC3916911.
[10]. A. Nasreen, K. Roy, K. Roy and G. Shobha, "Key Frame Extraction and Foreground Modelling Using K-Means Clustering," 2015 7th International Conference on Computational Intelligence, Communication Systems and Networks, Riga, Latvia, 2015, pp. 141-145, doi: 10.1109/CICSyN.2015.34.
[11]. Kulhare, S., Sah, S., Pillai, S., & Ptucha, R. (2016, December). Key frame extraction for salient activity recognition. In 2016 23rd International Conference on Pattern Recognition (ICPR) (pp. 835-840). IEEE.
[12]. W. Wolf, "Key frame selection by motion analysis," 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings, Atlanta, GA, USA, 1996, pp. 1228-1231 vol. 2, doi: 10.1109/ICASSP.1996.543588.
[13]. John S. Boreczky, Lawrence A. Rowe, "Comparison of video shot boundary detection techniques," Proc. SPIE 2670, Storage and Retrieval for Still Image and Video Databases IV, (13 March 1996); https://doi.org/10.1117/12.234794
[14]. Ghatak, Sanjoy, and Rangpo SMIT. "Key-frame extraction using threshold technique." International Journal of Engineering Applied Sciences and Technology 1, no. 8 (2016): 2455-2143.
[15]. Zhang, HongJiang, Atreyi Kankanhalli, and Stephen W. Smoliar. "Automatic partitioning of full-motion video." Multimedia systems 1 (1993): 10-28. https://doi.org/10.1007/BF01210504
[16]. Khurana, Khushboo, and M. B. Chandak. "Key frame extraction methodology for video annotation." International Journal of Computer Engineering and Technology 4, no. 2 (2013): 221-228.
[17]. A. Essa, P. Sidike and V. Asari, "A modular approach for key-frame selection in wide area surveillance video analysis," 2015 National Aerospace and Electronics Conference (NAECON), Dayton, OH, USA, 2015, pp. 41-44, doi: 10.1109/NAECON.2015.7443036.
[18]. Chen, Junyu, Ganlan Peng, Yuanfang Peng, Mu Fang, Zhibin Chen, Jianqing Li, and Liang Lan. "Key Clips and Key Frames Extraction of Videos Based on Deep Learning." In Journal of Physics: Conference Series, vol. 2025, no. 1, p. 012018. IOP Publishing, 2021.
[19]. Jahagirdar, Aditi, and Manoj Nagmode. "Two level key frame extraction for action recognition using content based adaptive threshold." Int. J. Intell. Eng. Syst 12, no. 5 (2019): 43-52.
[20]. Papadopoulos, Dim P., Vicky S. Kalogeiton, Savvas A. Chatzichristofis, and Nikos Papamarkos. "Automatic summarization and annotation of videos with lack of metadata information." Expert Systems with Applications 40, no. 14 (2013): 5765-5778. https://doi.org/10.1016/j.eswa.2013.02.016