Baghdad Science Journal

Volume 23 | Issue 4 Article 10

4-20-2026

Video Summarization Based on the Difference in Statistical Histograms of Fuzzy Clustering Features

Ekhlas Falih Naser
College of Computer Sciences, University of Technology, Baghdad, Iraq,
Ekhlas.F.Naser@uotechnology.edu.iq

How to Cite this Article

Naser, Ekhlas Falih (2026) "Video Summarization Based on the Difference in Statistical Histograms of Fuzzy Clustering Features," *Baghdad Science Journal*: Vol. 23: Iss. 4, Article 10.
DOI: https://doi.org/10.21123/2411-7986.5264

BAGHDAD SCIENCE JOURNAL 2026;23(4):1255–1269

RESEARCH ARTICLE

Video Summarization Based on the Difference in Statistical Histograms of Fuzzy Clustering Features

Ekhlas Falih Naser

College of Computer Sciences, University of Technology, Baghdad, Iraq

ABSTRACT

The quantity of video footage produced has skyrocketed in the last several years. The enormous increase in video content creates issues with content management. Video and image processing technologies need to receive greater attention to handle the increasing number of videos on the internet and to extract reliable information from them. Video summaries offer concise and streamlined depictions of the content of a video stream. This research aimed to build a methodology for video summarization utilizing the difference in the histogram of video frames using three stages. Frames were elicited from the video, and then SIFT detectors were utilized in the first stage to elicit interest points from each frame. Fuzzy C-means clustering was used in the second stage to collect the interest points. In the third stage, the histograms were made via the number of points for each cluster. The cluster number is represented by the x-axis, and the number of points in each cluster is represented on the y-axis. In order to elicit the summary frames, a value was manually entered, and then a query histogram was built based on these values. The Manhattan metric was used to compare the query histogram with all the histograms constructed in the third step. Experimental results displayed that the mean amount of time for clustering baby video is 3.860 with the proposed method, while the time was 9.670 to cluster the same video using all pixels; also, the precision was 94, the recall was 91, and the F1-score was 93.

Keywords: C-means clustering, Histogram, Manhattan distance measure, SIFT, Video summarization

Introduction

An effective design for the various image processing models is necessary to perform video summarization for large-sized sequences.[1] Any design should involve detecting shot boundaries:[2] different shot boundary clustering, event classification, shot boundary and event feature estimation, final summarization, etc.[3,4] Large-sized videos are converted into smaller chunks via event timestamp estimation.[5] For effective summarization, further analysis is performed for each of these smaller chunks, depending on various features extraction and classification methods.[6,7] To do so, a variety of event-based key-frame extraction models for video summarization are suggested by researchers.[8]. One method for cutting lengthy videos into shorter ones is video summarizing. Static and dynamic summarizing are the two main approaches to video summarization. Key frames and skimming are the two approaches utilized to get the summary.[9] The ability to view shorter videos from the original video, which may be difficult to watch in its entirety, where a summary is all that is needed to comprehend, is one of the many benefits of the video summarizing technique. Once more, while using a mobile device, the video summary helps save memory, extend battery life during downloads, and lower the cost of downloading a video, particularly for short videos that consume little data.[10] Once again, video summarizing techniques make browsing faster, particularly when indexing content, since many people stream stuff online, including music, movies, and soccer.[11]

Received 26 April 2025; revised 7 July 2025; accepted 11 July 2025.
Available online 20 April 2026

E-mail address: Ekhlas.F.Naser@uotechnology.edu.iq (E. F. Naser).

https://doi.org/10.21123/2411-7986.5264

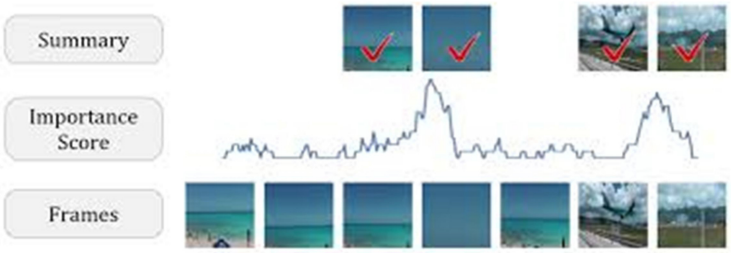

Fig. 1. Video summarization by mapping F: V—–>s.[4]

Fig. 1 expresses the essential architecture of video summarization with the mapping procedure among a large number of frame sequences (the input) and summarizes the short and selected frame sequences.[12] The contributions made to this study include:

1. The SIFT detector was utilized to extract additional interest points.
2. To effectively gather the interest points, fuzzy c-means clustering was utilized.
3. To reduce execution time and lower complexity in the video summarization process, a histogram was employed for the extracted interest points rather than using all pixels.

The remaining sections of the search consist of related works, methodology, Experiments and results, and conclusion.

Related works

Using frame-or shot-level video data as input, the video summarization approach trains the model to predict significance scores. This method uses the difference between the annotated significance score and the forecasted importance score to calculate the cost using the dataset.[13] The process lowers the price of identifying the ideal model. Numerous approaches have been put out, including:

- The authors in[14] introduced the mechanism that is built for the histogram of color, one of the lower-level attributes. In this research, detection of edges and histogram of color were the key metrics that were utilized for the elicitation of a key frame. The major objective beyond these metrics is to make certain that redundancy is isolated or discarded from the frame to permit sufficient efficiency of recognition and a smaller amount of complexity. The fact that they only use image color histograms is their main drawback. This indicates that the textures, edges, or true significance of the photographs' content are not captured.
- The authors in[15] introduced the ultimate significant mechanism for summarization of video via matching variances between two successive frames of video, which is calculated beside varying weights. Boundaries Detection of shots is achieved by relying on an automatic threshold. Ultimately, the elicitation of videos' key-frames is carried out via utilizing the frame's reference-based mechanism. The main limitation with successive frames of video is that each frame is typically very similar to the previous frame, as in the limiting case of video of a still image.
- The author of[16] looks into the value of nearby characteristics in producing video summaries. The suggested method makes use of SIFT features and is based on a bag of visual words. This method was tested against summaries produced using global features in an exploratory manner, and the results showed that while the local feature-based method did not perform better than the other methods, it appeared to be more reliable. Additionally, the extraction of features from SIFT and the discovery of visual words, which required 10 times longer than the extraction of all global features, demonstrates the disparity in runtime among the various approaches. High computational complexity, the requirement for a large number of values to represent features, and computational intensity are some of SIFT's drawbacks. In some applications, its high memory consumption and computational expense may act as a barrier.

The suggested methodology was similar to the method utilized in reference[14] in that both methodologies rely on calculating the histogram to elicit the key of the summarized frame, while the suggested methodology differs in that it relies on constructing the histogram based on the number of interest points in each cluster, and thus the execution time for extracting the summarized frame will be reduced and the complexity will also decrease, while in reference[14] the histogram was also constructed form all the colors of the image, which requires a longer execution time.

The suggested methodology was also similar to the methodology utilized in reference[16] in that both methodologies depend on the SIFT technique to elicit points of interest, but the suggested methodology differs in that the process of extracting the summarized frame is not directly based on these points, rather clusters were created from these points, and then the number of points in each cluster was relied upon, which reduced the need for storage space for save these points and reduced the degree of complexity, while in reference[16] the process of retrieving the summary frame was done on all SIFT attributes, thus increasing the degree of complexity and the need for a large storage space.

Methodology

The suggested methodology for video summarization consists of three stages. Frames were elicited from the input video, and then a SIFT detector and descriptor were utilized in the first stage to elicit points of interest from each frame. The SIFT technique was applied as follows in the proposed methodology: Eight neighbors, nine pixels from the scale above, and nine pixels from the scale below are compared to one pixel in a picture. A summary of the study's empirical findings on a number of parameters is as follows: Number of octaves = 4, number of scale levels = 5, initial σ =1.6, k = $\sqrt{2}$. An orientation histogram with 36 bins covering 360 degrees is produced by weighting it by gradient magnitude and using a Gaussian-weighted circular window with σ equal to 1.5 times the key-point scale.

The highest peak in the histogram is chosen to establish the orientation, and any peak that is higher than 80% of it is also considered. It generates key points with the same scale and location but different directions. It maintains the stability of the matching. Now, the key-point descriptor is ready. A 16 × 16 neighborhood is recorded around the key-point. There are sixteen 4 × 4 sub-blocks within it. For each sub-block, eight bin orientation histograms are created. Consequently, the total number of bin values is 128. It is represented as a vector in order to generate a key-point description. Furthermore, certain measures are used to guarantee resistance to rotation and illumination variations.

To collect the points of interest that were extracted from the first step, C-Means clustering was used in the second stage, and the proposed methodology relied on three clusters. The proposed methodology for constructing the clusters was based on the following parameters (the number of clusters was 3, the maximum number of iterations was 20, and the precision was 0.00001). Adopting these values for C-means coefficients reduces the time required for clustering.

In the third stage, the histograms were made based on the numbers of points for each cluster, and then the number of each cluster was stored in the file. The cluster number is represented by the x-axis, and the number of points in each cluster is represented by the y-axis. In order to elicit the summarized frames, the tested number of interest points was selected for each cluster from the file, then the tested histogram was built based on these points. The Manhattan scale was used to compare the tested histogram with all the histograms that were built in the third step, and all frames that had different histograms were determined to elicit the key frames and build the summarized video. The flowchart of the proposed method can be shown in Fig. 2.

Scale invariant feature transform algorithm (SIFT)

The SIFT algorithm consists of the following stages: detecting the extreme value of the scale space, localizing keys, determining orientation, and describing the key point. These are the four components that make up the conventional SIFT method.[17]

Detecting the extreme value of the scale space

Differential Gaussian functions are used to identify the sensitive spots, which are never changed to scale and rotation. Eqs. (1) and (2) illustrate how the Gaussian differential procedure and the initial picture are primarily convolved to produce the image's scale space Eq. (2).[18]

$$(x, y, z) = \frac{1}{2\pi\partial^2} e^{\frac{\left(x-\frac{m}{2}\right)^2+\left(y-\frac{n}{2}\right)^2}{2\partial^2}} \quad (1)$$

$$L(x, y, \partial) = G(x, y, z)\, xI(x, y) \quad (2)$$

Where the scale space is L(x, y, ∂) and Gaussian blur is G(x, y, z). The initial image is I(x, y), where (x, y) are the image coordinates of any pixel, d is the spatial scale, and (m, n) are the measurements of the image. Calculations determine the space maximum detection using the difference of Gaussian (DoG) detection can be illustrated in Eq. (3).[18]

$$\begin{aligned} D(x, y, \partial) &= \left(G(x, y, k\partial) - G(x, y, \partial)\right) \times I(x, y) \\ &= \left(L(x, y, k\partial) - L(x, y, \partial)\right) \end{aligned} \quad (3)$$

k = multiplication factor and S = integer value.[19] Eq. (4) describes k's computation:

$$k = 2^{\frac{1}{s}} \quad (4)$$

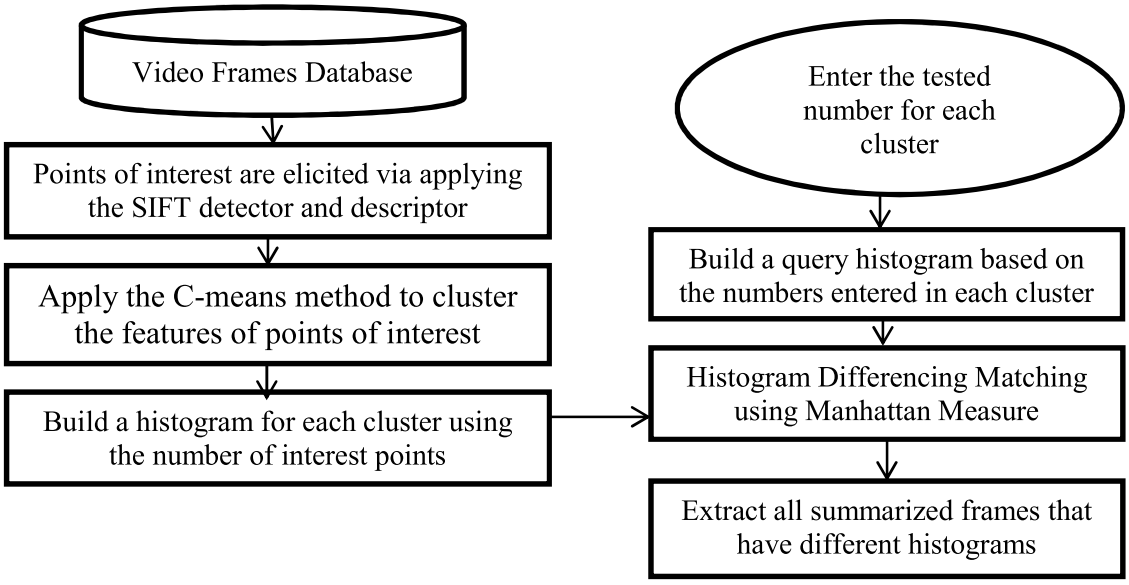

Fig. 2. A flowchart of the proposed system for summarized frames.

By contrasting images of double neighboring layers in the DoG space in the same set, the key points, which are local extremes, are first identified. A comparison is performed between each pixel and its neighbors to determine whether it is bigger or smaller than them in the image and the scale domain to locate the extreme point in DoG space. As described in Fig. 3, the detection point in the center (denoted by X in Fig. 3) is compared with its 8 neighboring pixels of the same plane, the 9 × 2 points corresponding to the higher and lower neighboring planes, for a total of 26 pixels, to make sure that maximum points are discovered in both scale space and 2-dimensional image plane.[20]

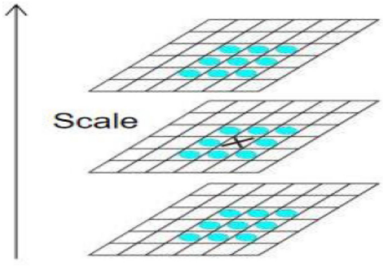

Fig. 3. Spatial extreme detection.[21]

Key point locating

The maximum point is not the actual extreme point in the distinct space. The process of employing well-defined discrete space point interpolation to acquire an extreme point in continuous space is known as sub-pixel interpolation.[22] The scale-space (DoG) function is subjected to a curve-fitting procedure to enhance the stability of the key spots. Using the Taylor extension of the plane space Eq. (5).[23]

$$D(x) = D + \frac{\partial D^T}{\partial X}X + \frac{1}{2}X^T\frac{D\partial^2}{\partial X^2}X \tag{5}$$

In Eq. (5), the key points are calculated using the formula X = (x, y, d) ^{T} to produce a robust edge reaction. To determine the threshold, it was necessary to remove the unstable edge response points.[23]

Determining the direction

After finding the key points (in the scale space), using local image features is necessary for assigning a reference direction to all the key points to get rotation invariance for the descriptor. The reason for using the image gradient method is to find the stable orientation of the local construction. Eq. (6) shows the magnitude and direction of the gradient.

$$m(x,y) = \sqrt{\begin{matrix}(L(x+1,y) - L(x-1,y))^2 \\ +(L(x,y+1) - L(x,y-1))^2\end{matrix}} \tag{6}$$

Where L is the value of the scale space where the key points are situated, q is the gradient direction, and m(x, y) is the gradient value. In a specific area, the histogram is employed to tally the key point gradient directions, to identify the direction that corresponds to the highest histogram peak, and then to employ that direction as the primary key point gradient direction.[23]

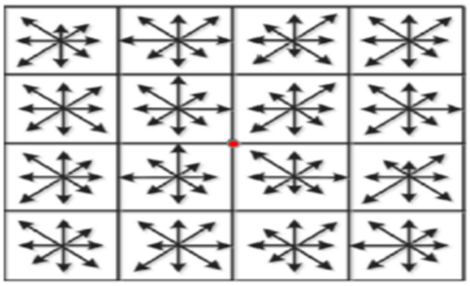

Fig. 4. 128 dimensional feature descriptor.[24]

Description of the key points

From the above, the following information is obtained: position, scale, and orientation. For each key point, a descriptor is established, and a set of vectors is utilized to define the key point to prevent any changes in lighting or perspective. In addition, this descriptor includes the pixels surrounding the key points that affect it. The descriptor must be unique in order to increase the likelihood of the feature points properly matching. The creation of the key point identifiers is depicted in Fig. 4. Before dividing the 4 × 4 grid into sections, choose a square area (of a 16 × 16 grid) to surround the feature points. Then, compute the collective gradient amounts of 8 orientations (one orientation for each 45 degrees) for each sub-region, and a 4 × 4 × 8 (128) dimensional feature descriptor is generated for each feature point.[24]

The feature vector of the critical point is the 4 × 4 × 8 = 128 gradient data. The feature vectors must then be normalized to eliminate the impact of illumination variations. The adjusted eigenvectors are displayed in Eq. (7).[22]

$$w_i = \frac{f_i}{\sqrt{\sum_{j=1}^{128} f_i}} \tag{7}$$

Where w_i = the normalized vector; fi = the initial feature vector; j = (1, 2, 3....). Setting the threshold number truncates the larger gradient values. Afterward, execute a normalization procedure to improve the ability to distinguish between features. SIFT characteristics need a lot of calculations and might not be the best for urgent applications, even though they are typically resistant to scaling and rotation. Since the suggested work is offline rather than real-time, SIFT is used in the suggested method. In addition to its ability to tolerate rotation, SIFT offers important points of interest. Future work should use ORB or SURF.

Features clustering using fuzzy C-means (FCM)

FCM is a powerful unsupervised method for data analysis, and its mode of construction is fuzzy clustering. Objects on the boundaries of multiple classes are given membership degrees ranging from 0 to 1, signifying their partial membership, rather than being compelled to fully belong to one of the classes. The FCM clustering algorithm is one of the often-used methods among related fuzzy models. Fuzzy clustering algorithms can be utilized to determine overlapping clusters in the data set and to count the function of membership, which indicates the degree to which objects belong to clusters. By using fuzzy partitioning, the FCM allows a data point to be a member of any group with a membership grade ranging from 1 to 0. This approach can be used to declare membership for each data point that corresponds to each cluster center depending on the distance between the cluster and the data point center. Additional data is located close to a cluster center, and its membership extends across a particular cluster center. Each data point's addition from membership must equal one. This algorithm assigns a membership value between 1 and 0 to the data items for the clusters. In order to support it, overlapping clusters can be formed by incorporating a certain number of fuzzy sets with partial membership.[25] An argument known as a fuzzification (m) with a domain [1, n] is needed for this procedure. The degree of fuzziness in these clusters is ascertained using this fuzzification argument. This algorithm functions as a crisp portioning algorithm when the value of (m) equals 1, and the overlapping of the clusters tends to be greater for bigger values of (m). The procedure uses Eq. (8)[26] to calculate a membership.

$$\text{value}.\mu_j(X_i) = \frac{\left(\frac{1}{d_{ji}}\right)^{\frac{1}{m-1}}}{\sum_{k=1}^{P}\left(\frac{1}{d_{ki}}\right)^{\frac{1}{m-1}}} \tag{8}$$

where (d_{ji}) is the distance of (X_i) in the cluster (C_i), (m) is the fuzzification parameter, (p) is the number of specified clusters, (d_{ki}) is the distance of (X_i) in the cluster (C_k), and (μ_j (X_i)) is the membership of (X_i) in the cluster's (j^{th}). By employing Eq. (9),[26] the most recent cluster entries are calculated with a membership value.

$$\sum_{j=1}^{p} \mu_j(x_i) = 1 \tag{9}$$

Eq. (9) is used to calculate the most recent membership values, which are then calculated using Eq. (10).

$$C_j = \frac{\sum_i \left[\mu_j(x_i)\right]^m x_i}{\sum_i \left[\mu_j(x_i)\right]^m} \tag{10}$$

Where (m) is the fuzzification parameter, (C_j) is the center of the (j^{th}) cluster, (x_i) is the i^{th} data point, and

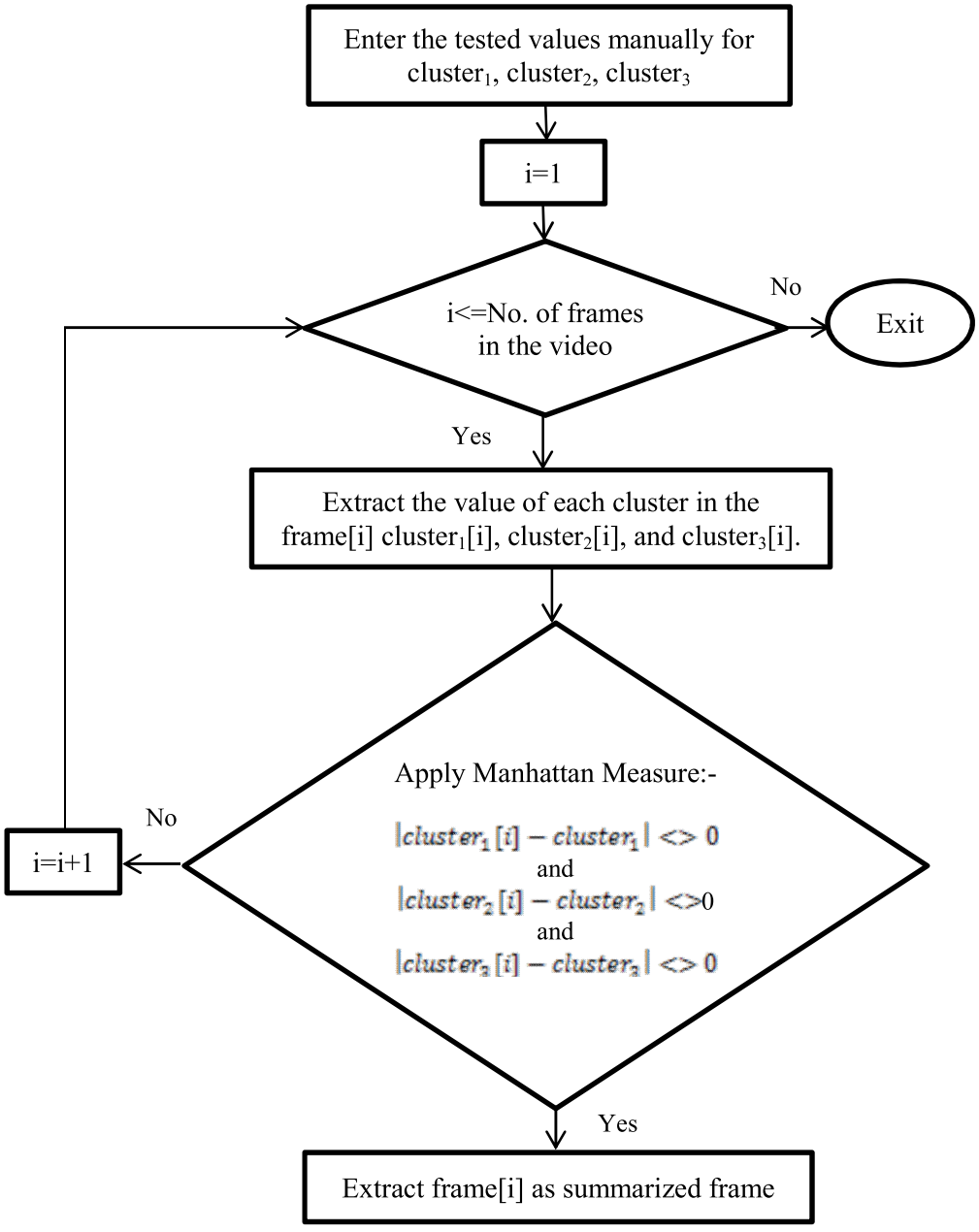

Fig. 5. The flowchart for extracting the summarized frames.

(μ_j) is the function that returns the membership. By using this form specifically, the weighted average's form can be calculated. It is possible to calculate the degree of fuzziness in (x_i's) present membership and multiply it by (x_i). The product obtained is divided by the sum of the fuzzified memberships. This method is utilized to calculate recent centroids.[27] FCM is frequently employed in target identification, shape analysis, image analysis, medical diagnosis, and pattern recognition. Using the idea of membership ratings of fuzzy logic sets, the FCM technique allows each point in the dataset to be an element of multiple sets to varying degrees. With this feature, it is inherently possible to learn a point's relationship to various cluster centers, which is impossible with hard k-means clustering. Fuzzy membership degrees are naturally interpreted to create the FCM clustering method. As a result, it was decided to concentrate on clustering using fuzzy logic in this research.[28] The FCM algorithm's stages are as follows:

1. The first stage in the FCM algorithm is the assignment of an initial random centroid to each cluster.

2. A straightforward algorithm is used to determine the separation between each spot and the cluster center.
3. Recalculate the membership function based on the separation between each point and the cluster at the middle.
4. Recalculate the center based on the membership function.

The algorithm will end if the difference between the initial centroid and the subsequent one is less than a predetermined threshold, such as € , otherwise, it will keep running until this condition is met.

Histogram differencing

The main purpose of the histogram differencing section is to construct histograms for all video frames based on the number of interest points in each cluster that were obtained using FCM. Then, the sequence of each frame in the video with the number of interest points in each cluster was saved in a file.[29] To extract the summarized frames, the different histograms were adopted after manually entering the query number for each cluster as a histogram query. The tested input values should be among those stored in the file, allowing the summarization process to be conducted using the histogram filter present in the video. The Manhattan method was used to extract the frame sequences with different histograms to generate the summarized frames and display them to get the summarized video. Through a special step of indexing, Manhattan's amount of clustering is used to achieve the haste of feature similarity.[30] If the grid is followed, the distance would be calculated by using the Manhattan function to go from one data point to another.[31] Manhattan distance is calculated by adding the differences between two components' corresponding parts.[31] The computation is described in Eq. (11).[32]

$$\sum_{i=0}^{n} |x_i - y_i| \tag{11}$$

Where, point X = (X1, X2, etc.) and point Y = (Y1, Y2, etc.)

The flowchart for extracting the summary frames is shown in Fig. 5.

Histograms differencing based on Euclidean distance captures the meaningful frames for summarization via comparing the interest points in cluster X with the values of the interest points in cluster Y by pixel-level-based euclidean distance measures the straight-line distance between two points and is suitable for continuous features with few attribute spaces. Manhattan distance calculates the sum of absolute differences and is suitable for high-dimensional data. Cosine similarity is derived from the cosine of the angle between two vectors, so it requires more complex calculations. For this reason, the proposed method relies on using Manhattan to find the difference between the query values of the histogram and all the stored histogram values in the file to extract the summarized keyframes based on different histograms, since the method deals with video data that are large in size and high in dimensions.

The proposed algorithm

The proposed algorithm is explained in Algorithm 1.

Algorithm 1: (Video Summarization Algorithm)

Input: Video File, the numbers which represent the features within **each** cluster
Output: A brief representation for Video Frames.
Begin:
Step 1: Load the video, extract all frames, and store them in a database called (frms).
Step 2: Insert test values into EC. (cluster_test) // EC = Each Cluster
For i = 1 to F. counter // F = no. of Frames
1. Detect the points of interest from (frms[i]) using the SIFT detector.
2. Implement the C-means clustering method for all the interesting points in frms[i] to get the clusters that include interesting point values in EC.
3. Create a histogram depending on the number of the interesting points in EC.
End loop
//For Extracting Keyframes
Step3: Insert the query number for each cluster, and then build the query histogram.
Step4: Extract all frames from frms[i] that contain different histograms than the query histogram by applying Eq. (8)
Step5: View all keyframes generated from step 4 as a representation of the summarized video.
END

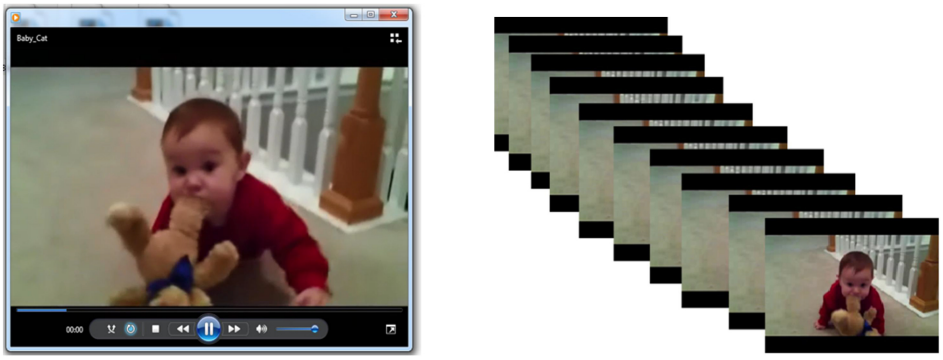

Fig. 6. a) Baby video, b) Original frames elicited from a video.

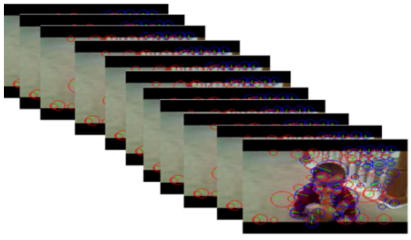

Fig. 7. Points of interest are elicited via applying the SIFT detector and descriptor.

Fig. 8. C-means clustering on baby frames.

Experiments and results

In this section, the experimental results of the proposed methodology are discussed and explained. The proposed method was implemented using the C# language. Ten types of videos were used to evaluate the proposed method, taken from the Kaggle dataset.[33] The dataset contains color videos. The proposed methodology standardized the size of each frame to 640 × 480 pixels. The proposed method includes several steps:

- Step 1. After uploading the video, all keyframes were elicited as illustrated, in Fig. 6 for the Baby video.
- Step 2. The points of interest were elicited in the second stage via applying the SIFT detector and descriptor. The empirical of this stage can be illustrated in Fig. 7 for Baby frames.
- Step 3. After the interest features were elicited from all keyframes, the approach of C-means clustering was used for those features to construct the clusters, as exhibited in Fig. 8 for the Baby.

The value of the number points for each cluster for a sample of baby video frames is shown in Table 1. While the value of the number of points of interest for each cluster for a sample of hammering video frames is shown in Table 2.

Table 1. Frame number with value of each cluster for a sample of Baby video frames.

Name No.	Cluster 1	Cluster 2	Cluster 3
Frame 1	0	8	6
Frame 30	1	13	10
Frame 100	5	13	0
Frame 170	9	3	6

Table 2. Frame number with value of each cluster for a sample of hammering video frames.

Name No.	Cluster 1	Cluster 2	Cluster 3
Frame 1	15	7	30
Frame 25	6	25	6
Frame 98	5	11	10
Frame 121	11	4	16

- Step 4. The histograms were created using the number of points for each cluster across all video frames, and the counts for each cluster were then saved to a file. To elicit the summarized frames, the selected number of interest points was manually entered for each cluster from the file. The query histogram was built based on the values of the clusters. The Manhattan scale was used to compare the query histogram with all the histograms that were built for all video frames, and all frames that had different histograms were determined to elicit the key frames and build the

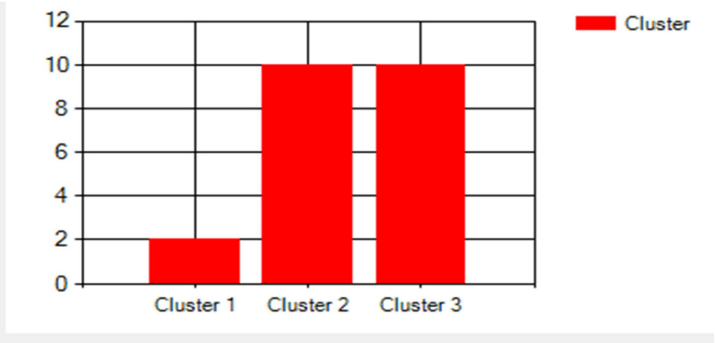

Fig. 9. Histogram was built for clustering interest points of baby key-frames.

Table 3. The mean time consumed for the clustering process using only SIFT features (proposed methodology) compared to the clustering process based on all colors (reference 14).

		Mean of amount of time consumed for Clustering in Second	
Video Name	No. of Frames	Clustering based on SIFT features Only	Clustering with all Colors features (Reference 14)
Baby	250	3.860	9.670
Hammering	133	2.709	7.882
Car	120	2.635	6.930
Scoter	50	2.044	5.226

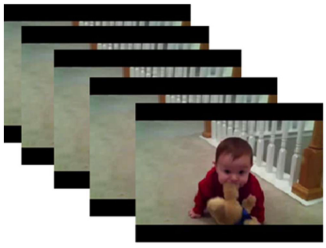

Fig. 10. Summarized keyframes are extracted using different histograms of the baby video.

summarized video. For example, when you manually enter a value of 2 for cluster 1 and a value of 7 for cluster 2 and cluster 3 to summarize baby video frames, the query histogram was displayed in Fig. 9.

- Step 5. The query histogram was compared with each video frame histogram by using the Manhattan distance metric to extract the summarized frames that have different histograms and display them as a summarized video, as presented in Fig. 10 for the Baby video frames.

For example, Frame 30, which is one of the summarized baby frames, has a different histogram (cluster 1 = 1, cluster 2 = 13, and cluster 3 = 10) than the query histogram, as can be seen in Fig. 11.

Fig. 12 shows the uploading of the Hammering video.

The points of interest for Hammering frames can be shown in Fig. 13.

The approach of C-means clustering for the Hammering can be shown in Fig. 14.

When you manually enter a value of zero for cluster 1, a value of 10 for cluster 2, and a value of 30 for cluster 3 to summarize hammering video frames, the query histogram is displayed in Fig. 15.

The summarized video, as presented in Fig. 16 for the hammering video frames.

For example, Frame 98, which is one of the summarized hammering frames, has a different histogram (cluster 1 = 5, cluster 2 = 11, and cluster 3 = 10) than the query histogram, as can be shown in Fig. 17.

Table 3 presents the time taken and the number of iterations required for clustering video frames, comparing results based on SIFT features alone with those utilizing whole-frame features for a sample of video frames.

Fig. 18 shows the the mean time consumed to cluster the all frames of videos when using the proposed method, which is based on clustering the features resulting from the SIFT, and clustering based on all frame color features (1-Baby, 2-Hammering, 3-Car and 4-Scoter).

For summarization quality, the proposed methodology calculates the precision, recall, and F1-score, as illustrated in Table 4. To compute the precision of the proposed method, it was utilized in Eq. (12). To compute the recall of the proposed method, it was utilized in Eq. (13). Also, to compute the F1-score of the proposed method, it was utilized in Eq. (14).[34]

$$Precision = \frac{TP}{TP + FP} \tag{12}$$

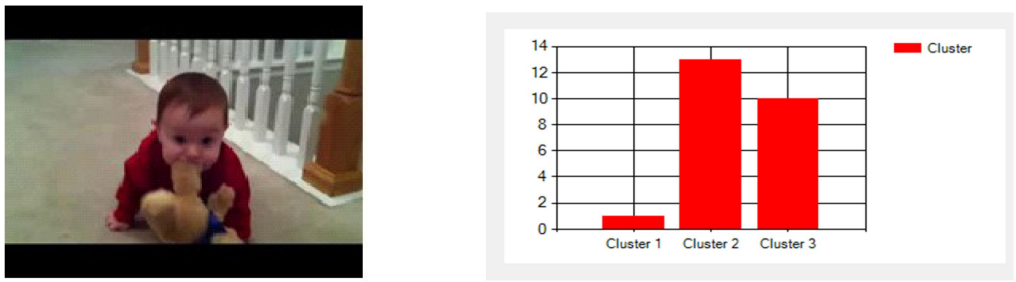

Fig. 11. a) Summarized Frame 30 b) Histogram of baby frame 30 histogram.

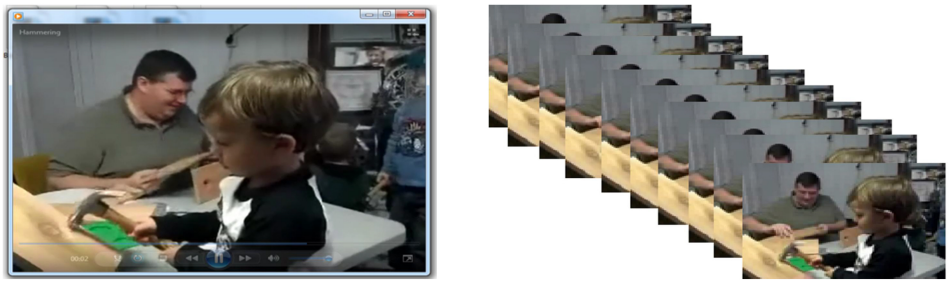

Fig. 12. Hammering video, a) Input video b) Original frames elicited from a video.

Fig. 13. Points of interest are elicited via applying the SIFT detector and descriptor.

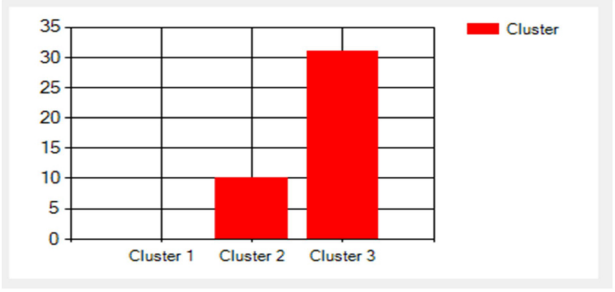

Fig. 15. Histogram for clustering interest points of hammering key-frames.

Fig. 14. C-means clustering on hammering frames.

Fig. 16. Summarized key-frames are extracted using different histograms of the hammering video.

a) b)

Fig. 17. a) Summarized frame 98 b) Histogram of hammering frame 98.

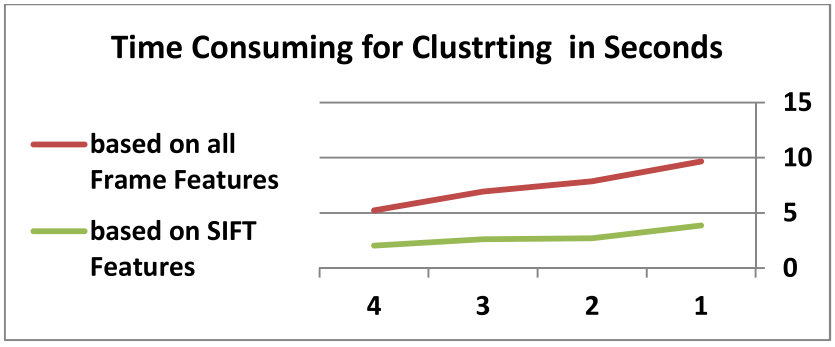

Fig. 18. The mean amount of time consumed for the clustering process.

$$Recall = \frac{TP}{TP + FN} \tag{13}$$

$$F1 - Score = \frac{Precision \times Recall}{Precision + Recall} \times 2 \tag{14}$$

- True Positive (TP): The number of cases that are correctly categorized as positive.
- False Positive (False Positive - FP): The number of cases that are incorrectly categorized as positive (when the actual status is negative).
- True Negative (True Negative - TN): The number of cases that were correctly categorized as negative.
- False Negative (False Negative - FN): The number of cases that are incorrectly categorized as negative (when the actual state is positive).

Table 4. Summarization quality measurements utilized to measure.

Video Name	Precision	Recall	F1-score
Baby	94	91	93
Hammering	97	93	96
Car	89	86	88
Scoter	90	87	89

From Table 2 above, we note that the precision value was 97, recall was 93, and the F1-score was 96 when summarizing the hammering video. The reason behind these high values was that the important features obtained from the SFT method were very numerous compared to the rest of the videos. Fig. 19 shows the values of summarization quality measurements, such as precision, recall, and F1-score, for summarizing the sampling videos (Baby, Hammering, Car, and Scoter).

Advantages and limitations of the proposed study

The advantages and limitations of the proposed study are tied to the constraints of its methodology. Each stage of the proposed study presents specific advantages and limitations, which can be illustrated as follows:

1. SIFT is resilient to rotation and scaling; however, it has its limitations. Its high computational cost makes it slower than some other approaches, which is a significant drawback. Additionally, it consumes a lot of memory, and in the case of large datasets, its 128-dimensional descriptor can result in slow image matching.
2. Despite its strength, fuzzy C-means (FCM) clustering has limitations. These include the

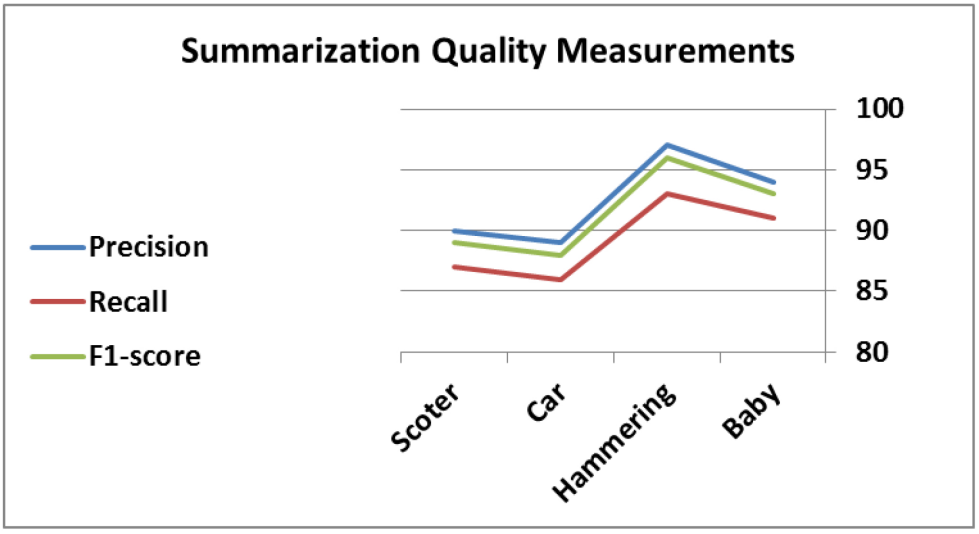

Fig. 19. The values of precision, recall and F1-score for video summarization rely on the proposed methodolgy.

requirement to predefine the number of clusters, the high computing complexity with large datasets, the sensitivity to initial conditions, and vulnerability to noise and outliers. Furthermore, it might not always converge to the global optimum and may have trouble with high-dimensional data.

3. Although they are helpful for showing the distribution of data, histograms have limitations. They may not adequately represent tiny datasets, have subjective bin size selections, and be challenging to compare with other histograms, particularly if their scales differ. Furthermore, histograms don't readily demonstrate correlations between several variables; instead, they mostly display the distribution of a single variable.

Conclusion

Summarization of video is the process of distilling valuable insights from a raw video by using a variety of criteria and mathematical formulations to reduce the film to a shorter summary. The goal of this paper is to summarize the video frames based on histogram differencing. The proposed methodology for video summarization consists of three stages. Frames were elicited from the input video, and then a SIFT detector and descriptor were utilized in the first stage to elicit points of interest from each frame. One great merit of SIFT is that it is able to generate a great number of features, covering the image across all locations and scales. To collect the points of interest that were extracted from the first step, fuzzy C-means clustering was used in the second stage, and the proposed methodology relied on three clusters. The major merit of fuzzy C-means clustering is that it allows gradual memberships of data points to clusters, weighted as degrees within [1,0]. This presents the elasticity to express that points of data can belong to further than one cluster. In the third stage, the histograms were made based on the number of points for each cluster, and then the number of each cluster was stored in a special file. The cluster number is represented on the x-axis, and the number of points in each cluster is represented on the y-axis. A histogram is often used to effectively illustrate the key features of a data distribution. It is particularly useful when analyzing large data sets, which consist of more than 100 observations. Additionally, it can help in detecting any gaps in the data or identifying unusual observations, known as outliers. In order to elicit the summarized frames, the tested number of interest points were manually selected for each cluster from the file. The query histogram was built based on these points, then the Manhattan scale was used to compare the tested histogram with all the histograms that were built in the third step, and all frames that had different histograms were determined to elicit the key frames and build the summarized video. As offered in Table 3, the experiment results displayed that the mean time consumed to cluster the frames based on SIFT features of the Baby video is 3.860 when using the proposed method, while the time used in the clustering process for the same frames is 9.670 when clustering depends on all pixel values of the frame. We also noticed that in Table 4 the precision value was 97, recall was 93, and the F1-score was 96 when summarizing the hammering video.

Table 5. A comparison with state of art for future works.

Proposed Stages Type	Comparison Description
Points of Interest Detection and Description	In computer-vision, both "SIFT" and "SURF" are feature detection and description methods; however, SURF is intended to be substantially quicker than SIFT. While SURF provides a decent trade-off between speed and performance, SIFT is renowned for its resilience to a variety of transformations, including scale, rotation, and lighting changes.
Clustering Method	The centroids are found by grouping the lower-dimensional representation using "fuzzy c-means". The centroids are converted back into the original format by the "autoencoder's" decoder so that they may be understood as the topics. In order to lessen bias in the detection model, a generative adversarial network provides generated data. In the fuzzy membership function, improved fuzzy c-means clustering takes into account the association between neighboring data points.
Extract Summarized Frames Methods	YOLOv11 is a cutting-edge object identification model, and a histogram is a visualization tool used in image processing to depict the distribution of pixel intensities. While YOLOv11 concentrates on recognizing and detecting objects within images, histograms are essential for comprehending image features. Compared to earlier YOLO versions, YOLOv11 delivers improvements in speed and accuracy, which makes it appropriate for real-time applications.

Future research suggestions

The suggestions for future works can be illustrated as follows:

1. I recommend using the "Speeded-Up Robust Features" (SURF) or "Oudemansiella raphanipies phenotype extractor" (ORP) or "Convolutional Neural Network" (CNN) methods to extract points of interests from video frames.
2. Autoencoders (AEs) or Generative Adversarial Networks (GANs) can be utilized to perform the clustering process.
3. Long Short Term Memory (LSTM) and Recurrent Neural Networks (RNNs) can be utilized to analyze information and events within video footage.
4. YOLO V11 can be used to identify summarized frames in real-time environments.

A comparison with a state of art for future work steps can be illustrated in Table 5.

Authors' declaration

- Conflicts of Interest: None.
- We hereby confirm that all the Figures and Tables in the manuscript are ours. Furthermore, any Figures and images that are not ours have been included with the necessary permission for re-publication, which is attached to the manuscript.
- No animal studies are present in the manuscript.
- No human studies are present in the manuscript.
- Ethical Clearance: The project was approved by the local ethical committee at University of Technology.

Data availability

The dataset was available on Kaggle based on the following link. Https://www.kaggle.com/datasets/pevogam/ucf101.

References

1. Kaur L, Mishra P K. Estimation of concise video summaries from long sequence videos using deep learning via LSTM. Int J Health Sci. 2022 Jun;6(3):9904–9914. https://doi.org/10.53730/ijhs.v6nS3.9287.
2. Hussien F T, Rahma A S, Wahab H B. A Block Cipher Algorithm Based on Magic Square for Secure E-bank Systems. Comp Math Sci J. 2022 Jan;73(1):1329–1346. https://doi.org/10.32604/cmc.2022.027582.
3. Karim A A, Sameer R A. Static and Dynamic Video Summarization. Iraqi J Sci. 2019 Jul;60(7):1627–1638. https://doi.org/10.24996/ijs.2019.60.7.2.
4. Ahmed S K, Ali E A, Naser E F. Iraqi license plate detection using edges and contours with different acquisition conditions. AIP Conf. Pro. 2023 Dec;2977(030007):030007–11. https://doi.org/10.1063/5.0182345.
5. Alawi A R, Hassan N F. A Proposal Video Encryption Using Light Stream Algorithm. J Eng Tech. 2021 Mar;39(1):184–196. https://doi.org/10.30684/etj.v39i1B.1689.
6. Abdulmohsin H A, Abdul wahab H B, Abdul hossen A J. A new proposed statistical feature extraction method in speech emotion recognition. J Comp Elect Eng. 2021 Jul;93(5):160–172. http://dx.doi.org/10.1016/j.compeleceng.2021.107172.
7. Hussain T, Muhammad K, Ding W, Lioret J, Baik W S, Hugo V, Albuquerq V H C. A Comprehensive Survey on Multi-View Video Summarization. Pattern Rec. 2021 Jan;109(107567):1–15. https://doi.org/10.1016/j.patcog.2020.107567.
8. Workie A, Sharma R, Chung Y K. Digital Video Summarization Techniques: A Survey. Int J Eng Res Tech (IJERT). 2020 Jan;9(1):81–85. https://doi.org/10.17577/IJERTV9IS010026.
9. Pan G, Zheng Y, Zhang R, Han Z, Sun D, Qu X. A bottom-up summarization algorithm for videos in the wild. EURASIP J Adv Signal Pro. 2019 Feb;15(2019):1–11. https://doi.org/10.1186/s13634-019-0611-y.

10. Apostolidis E, Adamantidou E, Metsai A, Mezaris V, Patras I. Performance over Random: A Robust Evaluation Protocol for Video Summarization Methods. 28th ACM Int Conf. Mult. 2020 Oct;1056–1064. https://doi.org/10.1145/3394171.3413632.
11. Atencio P, Sanchez G, Branch J, Delrieux C. Video Summarization by Deep Visual and Categorical Diversity. Instit Eng Tech. 2019 Aug;13(6):569–577. https://doi.org/10.1049/iet-cvi.2018.5436.
12. Ji Z, Zhao Y, Pang Y, Han J. Deep Attentive Video Summarization with Distribution Consistency Learning. IEEE Trans Neural Net Learn Sys. 2021 May;32(2021):1765–1775. https://doi.org/10.1109/TNNLS.2020.2991083.
13. Saini P, Kumar K, Kashid S, Saini A, Negi A. Video summarization using deep learning techniques: a detailed analysis and investigation. Artificial Intel Rev. 2023 Mar;56(11):12347–12385. https://dx.doi.org/10.1007/s10462-023-10444-0.
14. Billur D D, Manu T M, Patil V.. A Comparative Analysis of Video Summarization Techniques. Int J Eng Man. (IJEM). 2023 Jun;13(3):10–24. https://dx.doi.org/10.5815/ijem.2023.03.02.
15. Abdulsahib M G, Abdulmunim M E. Multimodal video abstraction into a static document using deep learning. Int J Elec Comp Eng (IJECE). 2023 Jun;13(3):2752–2760. https://doi.org/10.11591/ijece.v13i3.
16. Li F.Synchronous restoration of video key frame loss based on digital media communication protocol. SN App Sci. 2023 Feb;5(2):1–10. https://doi.org/10.1007/s42452-023-05286-y.
17. Abdulsahib M G, Abdulmunim M E. Convolutional Recurrent Neural Networks for Text Lecture Summarization. Iraqi J. Comp. Comm Cont Sys Eng (IJCCCE). 2022 Jun;22(2):27–39. https://doi.org/10.33103/uot.ijccce.22.2.3.
18. Ismail R, Zaki R M, Abdulmunim M E. A Study for Self-Driving Car Analysis. Iraqi J Comp Comm Cont Sys Eng (IJCCCE). 2024 Dec;24(4):45–60. https://doi.org/10.33103/uot.ijccce.24.4.4.
19. Al-zubaidi S M T. Application of Improved PSO in Augmented Reality for Dental Healthcare. J Adv Res Appl. Sci Eng Tech. 2025 Aug;50(2):90–102. https://doi.org/10.37934/araset.50.2.90102.
20. Naser E F, Khudair E T, Mahmood E S, Maolood A T. A Comparison between Backpropagation Neural Network and Seven Moments for More Accurate Fingerprint Video Frames Recognition. Baghdad Sci J. 2024 Apr;21(11):3583–3591. https://doi.org/10.21123/bsj.2024.8777.
21. Javaid M, Maqsood M, Aadil F, Safdar J, Kim Y. An Efficient Method for Underwater Video Summarization and Object Detection Using YoLoV3. Int Auto Soft Comp. 2023 Mar;35(2):1295–1310. https://doi.org/10.32604/iasc.2023.028262.
22. Abbas A R, Naser E F. Detecting Interpolated Video Frames based on Convolution Neural Networks. J Eng Rers. 2025 Dec; 22(2):53–162. https://doi.org/10.53540/1726-6742.1317.
23. Tang L, Ma S, Ma X, You H. Research on Image Matching of Improved SIFT Algorithm Based on Stability Factor and Feature Descriptor Simplification. App Sci J. 2022 Jul;1(17):1–19. https://doi.org/10.3390/app12178448.
24. Zhang W-Y, Zhou T, Xu C, Liu M. A SIFT-Like Feature Detector and Descriptor for Multi-beam Sonar Image. J Sens. 2021 Jul;3(2021):1–14. https://doi.org/10.1155/2021/8845814.
25. Rukundo O. Normalized weighting schemes for image interpolation algorithm. Appl Sci J. 2023 Jan;13(3):1–16. https://doi.org/10.3390/app13031741.
26. Yin T, Lyu Z. Optimal Extraction Method of Feature Points in Key Frame Image of Mobile Network Animation. Mob Net App Sprin. 2022 Dec;27(5):2515–2523. https://doi.org/10.107/s11036-022-02070-x.
27. Mohammed G M, Melhum A I. Implementation of HOG Feature Extraction with Tuned Parameters for Human Face Detection. Int J Mach Learn Comput. 2020 Jul;10(5):654–661. https://doi.org/10.18178/ijmlc.2020.10.5.987.
28. Jiao J, Wang X, Wei T, Zhang J. An Adaptive Fuzzy C-Means Noise Image Segmentation Algorithm Combining Local and Regional Information. IEEE Trans Fuzzy Sys. 2023 Aug;PP(99):1–14. http://doi.org/10.1109/TFUZZ.2023.3235392.
29. Zhang X, Wang H, Zhang Y, Gao X, Wang G, Zhang C. Improved fuzzy clustering for image segmentation based on a Low-rank prior. Comp. Visual Media. 2021 Dec;7(4):513–528. http://doi.org/10.1007/s41095-021-0239-3.
30. Zhang H, Huang S-L. Improved fuzzy C-means clustering algorithm based on fuzzy particle swarm optimization for solving data clustering problems. Math Comp Simu. 2025 Jul;233(2025):311–329. https://doi.org/10.1016/j.matcom.2025.02.012.
31. Najeeb H, Ghani R F. Proposed method for scale drawing calculating depending on the line detector and length detector .Iraqi J Comp Sci. Math. 2021 Jul;2(2):6–17. https://doi.org/10.52866/ijcsm.2021.02.02.002.
32. Kareem F H., Naser M A. Face Detection and Localization in Video Using HOG with CNN. Fus Prac Appl (FPA). 2025 Feb;17(1):229–237..https://doi.org/10.54216/FPA.170117.
33. Pevogam. UCF101 Videos [Internet]. Kaggle. 2020.
34. Lee M C H, Braet J, Springael J. Performance Metrics for Multilabel Emotion Classification: Comparing Micro, Macro, and Weighted F1-Scores. Appl Sci J. 2024 Oct;14(21):1–21. https://doi.org/10.3390/app 14219863.

تلخيص الفيديو بألأعتماد على الاختلاف في المدرجات الإحصائية لصفات التجميع الضبابي

إخلاص فالح ناصر

كلية علوم الحاسوب، الجامعة التكنولوجية، بغداد، العراق.

الخلاصة

شهدت كمية مقاطع الفيديو المنتجة ارتفاعًا هائلًا في السنوات الأخيرة، مما أدى إلى ظهور تحديات في إدارة المحتوى. لذا، بات من الضروري إيلاء المزيد من الاهتمام لتقنيات معالجة الفيديو والصور للتعامل مع الكم الهائل من الفيديوهات على الإنترنت واستخلاص معلومات موثوقة منها. تقدم ملخصات الفيديو عرضًا موجزًا ومبسطًا لمحتوى الفيديو. هدف هذا البحث إلى بناء منهجية لتلخيص الفيديو تعتمد على الفرق في الرسم البياني لإطارات الفيديو، وذلك عبر ثلاث مراحل. في المرحلة الأولى، تم استخراج الإطارات من الفيديو، ثم استُخدمت خوارزمية SIFT لاستخراج نقاط المهمة من كل إطار. وفي المرحلة الثانية، استُخدمت خوارزمية التجميع الضبابي C-means لجمع النقاط المهمه. أما في المرحلة الثالثة، فقد تم إنشاء الرسوم البيانية بناءً على عدد النقاط في كل مجموعة. يُمثل المحور السيني عدد المجموعات، بينما يُمثل المحور الصادي عدد النقاط في كل مجموعة. ولاستخراج الإطارات الملخصة، تم إدخال قيمة يدويًا، ثم تم بناء رسم بياني للاستعلام استنادًا إلى هذه القيم. استُخدم مقياس مانهاتن لمقارنة الرسم البياني للاستعلام مع جميع الرسوم البيانية التي تم إنشاؤها في الخطوة الثالثة. أظهرت النتائج التجريبية أن متوسط الوقت اللازم لتجميع فيديو الطفل هو 3.860 باستخدام الطريقة المقترحة، بينما استغرق تجميع الفيديو نفسه باستخدام جميع وحدات البكسل 9.670 ثانية؛ كما بلغت الدقة 94، والاستدعاء 91، ودرجة F1 93.

الكلمات المفتاحية: التجميع باستخدام C-Means, المدرج الأحصائي , مقياس مسافة مانهاتن, SIFT، تلخيص الفيديو.