Key-frame-Based Video Summarization Using Similarity Measure

Manjusha Yeola and Sunita Barve

Abstract The volume of video data has increased dramatically in recent years, necessitating the development of efficient browsing, indexing, retrieval, and sharing technologies. Most of the time, users watch brief or summarized videos. With the help of the shortened video, users may grasp the subject matter rapidly. The most important scenes in a video are recognized as key-frames and their overall context is preserved. A logical synopsis is then created by combining these key-frames. Since key-frames include a variety of important and interesting information, extracting them is an important step in the video summarizing process. The literature uses a number of different ways. It is less effective and performs poorly. The goal of this research study was to use the cosine similarity metric to obtain key-frame-based video summarization. Five sample videos from SumMe dataset were used for experimental analysis, and the summarized videos are evaluated by compression ratio and F1-score.

Keywords Key-frame extraction · Video summarization · Feature extraction · Cosine similarity

1 Introduction

In the era of rapidly advancing technology, the widespread adoption of digital telecommunications and an enormous rise in video recording devices, together with the widespread sharing of videos on social media websites like Facebook, LinkedIn, Twitter, YouTube, Instagram, and others, are aging quickly. The quick expansion of digital resources—text documents, audio files, images, videos, and so on—has led to issues with memory explosion, mass video retrieval, and resource consumption. Finding irregularities in the CCTV footage requires many hours of work and the

M. Yeola (✉)
Department of Computer Engineering, Dr D Y Patil Institute of Technology Pimpri, Pune, India
e-mail: manjusha.amritkar@gmail.com

S. Barve
School of Computer Engineering, MIT Academy of Engineering Alandi, Pune, Maharashtra, India
e-mail: ssbarve@mitaoe.ac.in

S. J. Nanda et al. (eds.), *Data Science and Applications*, Lecture Notes in Networks and Systems 1266, https://doi.org/10.1007/978-981-96-2647-2_33

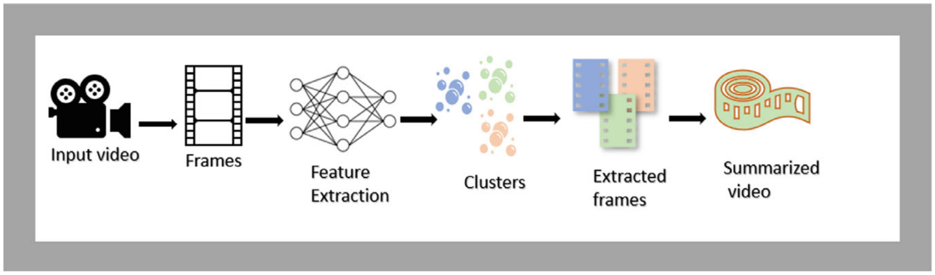


Fig. 1 Video summarization using key-frame extraction

expertise of a trained person. Wearable computers can record egocentric movies; however, it is exceedingly challenging to identify recurrent patterns in these kinds of videos. Watching the lengthy videos of medical procedures also takes time. Muhammad et al. [1, 2] proposed the solution of automating the process of removing unnecessary content from these movies and gathering relevant evidence is necessary, as processing these movies by hand would be quite time-consuming. Every time, a video overview resolves this issue successfully. It will be useful for people to spot any interesting patterns in the footage immediately. It's been demonstrated that the best approach to find news, movie trailers, and highlights from sports videos is to summarize them.

The aim of the Key-frame extraction technique is to wrench out the frames with pivotal information from the original video. The most insightful frames in a video that catches significant and varying contents are called Key-frames. Therefore, accuracy in key-frame extraction techniques which directly impacts the user experience is essential. Basic structure of key-frame extraction is illustrated in Fig. 1.

Extracting key-frames is an important and mostly used technique in the field of video summarization. The literature employs a variety of ways. It involves extracting key-frames using deep learning techniques like CNN and LSTM-based, as well as conventional techniques like color-based, motion-based, cluster-based, and so on.

Color is regarded as a crucial component of video. For this reason, video summarization has made extensive use of it. Muhammad et al. in [3] proposed a color-based key-frame extraction method. A histogram with gradient, color, saturation, and contrast that is taken from patches represents the video material. Jadon and Jasim [4] extracted key-frames based on dissimilarities between histograms of two frames. The difference of two frames in the histogram is used to infer a scene change, and the frames themselves are treated as necessary components in the summary's construction. Chai et al. [5] utilized graph-based structure to represent video. Structural changes between graphs of consecutive frames obtained to extract key-frames. These techniques are easy to use; however, they don't really improve accuracy.

Commonly used color-based techniques do not contain motion information in videos. One of the most important aspects of video elements is motion. It is user-friendly and offers an economical computing cost. Because of this, Mendi et al. [6]

obtained key-frames by employing optical flow and computed the brightness difference between two consecutive frames. Mohan and Nair [7] proposed a method for combining frames, based on the criteria of reduced motion or none at all. Zeng and Huang [8] adopted the motion vector's magnitude and the Kanade–Lucas–Tomasi (KLT) feature tracker to determine this noteworthy change, and the frames are divided. It will be possible to extract and use representative frames from a video by using this motion data. Videos with low to medium motion are ideal for motion-based summary. It makes more sense for surveillance-style videos when there is less apparent movement.

Because videos move erratically, motion-based algorithms are unable to identify redundancy in frames. Deep features from the video have to be extracted in order to identify important frames. Sushma and Pulikala in [9], extracted high-level characteristics using convolutional auto-encoders for endoscopic images. Harakannanavar et al. [10] recommended ResNet-18 for feature extraction and is pre-trained on the ImageNet dataset. It locates a chronologically cohesive summary with success.

Despite significant advances in this field of research, video summarizing remains an unsolved topic due to the current solutions' poor performance and efficiency. This research proposes a straightforward approach to extract relevant frames from video using similarity measures. When extracting features from video frames, a deep learning method yields far better results. Paper outline: Sect. 2 presents related work; the proposed methodology is described in Sect. 3. Experimental results are depicted in Sect. 4, and finally, a conclusion is made predicting future directions in Sect. 5.

2 Related Work

One of the most important steps in the process of video summarization is choosing the key-frames. A key-frame is a frame that serves as a summary of the video and highlights a significant part of the shot. Numerous methods have been used in the literature to extract key-frames, and they can be divided into two primary categories: deep learning-based and traditional-based.

2.1 Key-Frame Selection Using Traditional Approaches

Lai and Yi [11] proposed a clustering algorithm in which frames that are similar are grouped together. The author used the HSV color histogram as the visual material, and key-frames are clustered using widely used similarity metrics. Clustering-based key-frame extraction techniques do not give priority to objects or events that are interesting. The researchers developed methods for locating semantically closer frames in order to increase accuracy. Semantically significant frames are extracted using approaches that rely on visual attention. The authors made an effort to ascertain attention levels both statically and dynamically for a variety of elements, including color, motion, and texture. An attention curve is produced when all of these data are put together. Retrieved are key-frames with the highest saliency value. In order to locate a movie thematically, key-frame extraction is essential. Lei et al. [12] proposed the technique for key-frame indexing of a video. Singular value decomposition is utilized to rank the frames, and key-frames are picked using an inter-frame distance metric, resulting in minimal redundancy and semantically organized content.

The representative summary must also provide motion information in addition to still images. Huand and Wang [13] introduced the concept of shot segmentation based on spatial–temporal information. Key-frames are selected from the segmented pictures, and the inter-frame motion curve is built using the CapsuleNet function. Gawande et al. [14] used a histogram to extract features and the Bhattacharya distance method to determine frame distance. Key-frames are used as a testing module and are added to the queue based on the threshold value and matched score. The best match frame, which serves as the key-frame, is obtained by extracting features from the input frame and matching them with features that have been acquired using CNN. This strategy works better than the conventional approaches because of its minimal time complexity. Kızıltepe et al. [15] improved the video classification's accuracy by combining two powerful architectures. In order to extract key-frames, he connected the CNN-RNN architecture with the region of interest. Man and Sun [16] proposed an algorithm for key-frame extraction for commodity-type films that considers both local and global picture qualities together with image description. The e-commerce platform benefits from this work. To identify critical frames, all state-of-the-art methods started with information extraction, such as the HSV color histogram, motion features, texture features, or a combination of these. Through the use of shot detection-based segmentation and deep learning techniques, key-frame extraction enhances performance [17].

2.2 Key-Frame Selection Using Deep Learning

Because video is sequential, several academics have identified sequence-to-sequence modeling as a barrier to summary synthesis. To demonstrate this Mahasseni et al. [18] proposed a technique using LSTM and adversarial LSTM, and produced a summary. And Zhao et al. [19] introduced hierarchical RNN-based techniques, to overcome the drawback caused by RNN; they are unable to establish long-term dependency. In order to counteract the information loss resulting from RNNs, Ghauri et al. [20] combined different feature types along with an attention-based video summarizing framework called MSVA. In order to remove information loss in RNN-based models and to handle long-duration videos Zhao et al. [21] introduced hierarchical RNN. Zhu et al. [22] leveraged temporal consistency by using an anchor-based approach. Li et al. [23] addressed the fundamental structure of video frames by introducing a graph convolutional attention network (GCAN). The main pattern of the video frames is captured by the author using graph embedding, which enables the selection of semantically dependent frames that are then used to deliver summary information.

Another choice for attention-based approaches is Transformer, which has the ability to analyze global dependencies. Zhao et al. in [24] presented a hierarchical transformer to provide scenic video summaries by capturing global cues between shots and frames. The proposed method also has the advantage of using both visual and aural information, which is essential for video summarizing. The transformer's high parameter requirement is one of its drawbacks. To offer a solution to this problem, Chen et al. [25] proposed a method considering a U-shaped transformer that can both detect intermediate frame features and reduce processing cost. Yoon et al. [26] computed the key-frame level relevance score using CNN. Key-frames are equally and efficiently picked, and rewards are computed via reinforcement learning. Tan et al. [27] applied a large model-based sequential key-frame extraction technique to extract shot-level key-frames. In addition, the suggested method extracts characteristics using a deep learning model, which makes it more efficient than conventional methods.

3 Proposed Methodology

The approach for extracting video key-frames based on similarity metrics is presented in this study. High-level video features are extracted using deep feature extraction models that have been pre-trained on the ImageNet dataset. Feature vectors are employed in the following stage to locate pertinent frames. The summary video is obtained by consuming the chosen frames. Key-frame selection utilizing the similarity measure is shown in Fig. 2.

Let video is denoted as V and is divided into N frames as shown in Eq. 1.

$$V = \sum_{i=1}^{N} fi, \tag{1}$$

$$FV = \sum_{i=1}^{N} fvi. \tag{2}$$

Every frame's features that were extracted using deep learning models are contained in the FV as in Eq. 2. By default, the first frame is chosen to be the

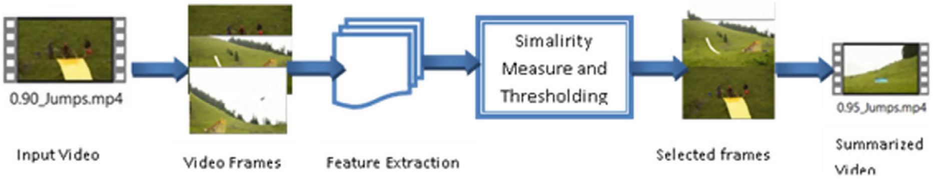


Fig. 2 Key-frame selection using similarity measure

key-frame. The selection of key-frames *fk* is done using Eq. 3.

$$\sum_{k=1}^{M} fk = S(FV), \tag{3}$$

where *S* is the similarity function applied on the feature vector and *M* << *N*.

3.1 Deep Feature Extraction

Deep features have been extracted from segmented frames using the CNN model. These deep learning models are capable of identifying an image's semantic and contextual elements. Simple features are extracted by the first layers, and deep features are captured by the final convolution layer. The number of layers in a feature enhances its complexity.

As feature extractors, we compared three deep learning models: ResNet-50, GoogleNet, and VGG16. These models take in each frame as input and output a feature vector. Pre-training of these models was done at a resolution of 224 × 224 using the ImageNet dataset. Equation 4 is used to acquire the feature vectors.

$$\sum_{i=1}^{N} fvi = \ominus(fi), \tag{4}$$

where Θ denotes deep learning feature extraction model and fvi is the feature vector of each video frame.

3.2 Key-frame Selection

The first step in creating a video summary is choosing key-frames. It handles full video and trims the content for user-requested indexing and other uses. The film's length is decreased while preserving the semantic content of the video by choosing key-frames that are contextually dependent. To do this, we have determined the difference between consecutive frames using the similarity metric. Frames that are smaller than the threshold δ are chosen as key-frames when the computed distance is compared with it. Algorithm 1 illustrates the key-frame selection process.

Algorithm1: Key-frame selection using CNN-SM method	
Requirement:	Video **V**
Output:	Key-frames **F = {** $\mathbf{f_1}$**,** $\mathbf{f_2}$**,………,**$\mathbf{f_M}$**}** and **Summarized video V**

(continued)

(continued)

Step 1:		Segmentation
	1.1	Divide a video **V** into $\{\mathbf{f_1}, \mathbf{f_2}, \ldots\ldots\ldots.., \mathbf{f_N}\}$, where **N > > M**
Step 2:		Feature extraction
	2.1	Extract deep feature vector $[\mathbf{f_{vi}}]^{N}_{\mathbf{i=1}}$ using deep learning model
Step 3:		Similarity measure
	3.1	Calculate similarity **S** between feature vectors of video frames
	3.2	if S < δ, select frame as key-frame
Step 4:		Video summary V is generated using selected frames

Our similarity metric of choice for determining the relationship between the two frames is cosine similarity.

4 Experimental Results

The algorithm is tested using the five sample videos from the SumMe dataset introduced by Gaygli et al. [17]. These videos last between one and three minutes, at a maximum frame rate of 25 to 30 frames per second. Sample videos show situations that shift either swiftly or slowly [5]. *N* frames are extracted from each video. GoogleNet, VGG16, and ResNet-50 are three deep learning models that are used to create feature vectors of segmented frames.

Cosine similarity is applied to the feature vector to choose key-frames. The suggested method selects key-frames whose similarity *S* is smaller than the threshold, where the threshold values are 0.90 and 0.95. Tang et al. [28] conducted a comparison analysis using the compression ratio (CR) in accordance with Eq. 5.

$$CR(V) = (1 - \frac{(Nk}{N)) * 100}. \tag{5}$$

N denotes the total number of frames in the video *V* and *Nk* the number of key-frames that were chosen. Three pre-trained models were used in the experiment. Table 1 displays the outcome that the ResNet model produced. Table 2 displays the GoogleNet result, and Table 3 displays the result obtained by VGG16.

Table 1 Compression ratio of five different videos from SumMe dataset using ResNet

Video		Threshold (0.95)		Threshold (0.90)	
	Total frames	Key-frames	CR	Key-frames	CR
Jumps	950	194	79.57895	70	92.63158
Cooking	1280	240	81.25	87	93.20313
Bikepolo	3064	1380	54.96084	912	70.23499
Bearpark	3341	841	74.8279	216	93.53487
Statue of liberty	3863	443	88.53223	129	96.66063
			Avg = 75.83		Avg = 89.25

Table 2 Compression ratio of five different videos from SumMe dataset using GoogleNet

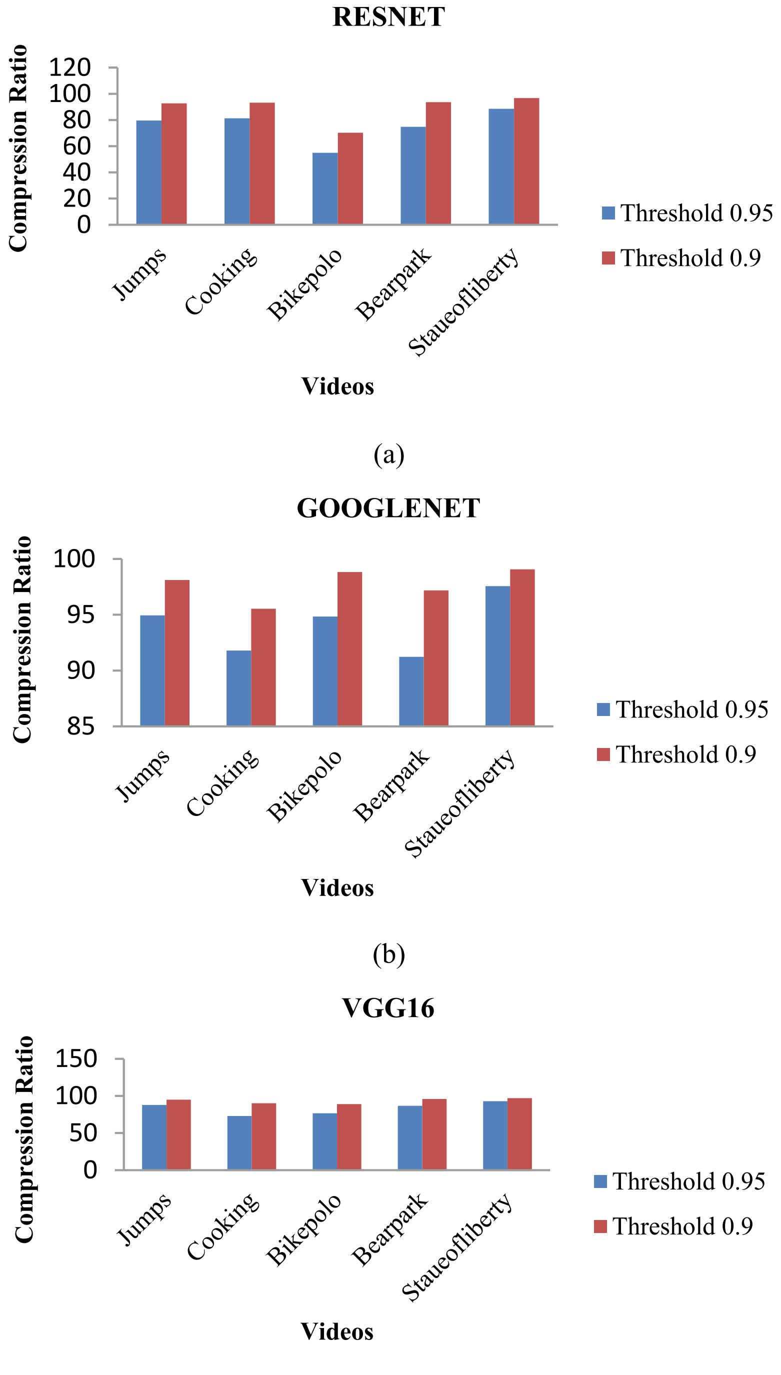
Video		Threshold (0.95)		Threshold (0.90)	
	Total frames	Key-frames	CR	Key-frames	CR
Jumps	950	48	94.94736842	18	98.10526316
Cooking	1280	105	91.796875	57	95.546875
Bikepolo	3064	158	94.84334204	36	98.82506527
Bearpark	3341	293	91.23017061	94	97.18647112
Statue of liberty	3863	94	97.56665804	36	99.0680818
			Avg = 94.08		Avg = 97.75

Table 3 Compression ratio of five different videos from SumMe dataset using VGG16

Video		Threshold (0.95)		Threshold (0.90)	
	Total frames	Key-frames	CR	Key-frames	CR
Jumps	950	116	87.78947368	47	95.05263158
Cooking	1280	346	72.96875	127	90.078125
Bikepolo	3064	712	76.76240209	335	89.06657963
Bearpark	3341	446	86.7	135	96.0
Statue of liberty	3863	277	92.8294072	110	97.15247217
			Avg = 83.41		Avg = 93.47

When compared to other $\delta = 0.95$ values, the suggested technique has a high average CR of 97.75 when taking $\delta = 0.90$ into account. In addition, it is noted that when compared to other models, GoogleNet features a condensed video summary. Figure 3 presents a comparative examination of all the models.

We chose one sample movie, “Jumps,” from the SumMe collection for the other evaluation metric, F-score, and got in touch with three distinct users with technical



(a)

(b)

(c)

Fig. 3 Comparative analysis based on compression ratio using cosine similarity using **a** ResNet, **b** GoogleNet, and **c** VGG16

expertise. They provide the index of significant frames after manually analyzing the video. Equation 6 is used to determine the F-score based on the user-selected frames and the model-generated frames.

$$\frac{F - \text{score} = 2 \cdot (\text{precision} * \text{recall})}{(\text{precision} + \text{recall})}, \tag{6}$$

where precision and recall are calculated as per Eqs. 7 and 8, respectively.

$$\frac{\text{Precision} = \text{TP}}{(\text{TP} + \text{FP})}, \tag{7}$$

$$\frac{\text{Recall} = \text{TP}}{(\text{TP} + \text{FN})}. \tag{8}$$

In this case, the number of frames that match from the user summary and the model summary is represented by TP. The number of frames chosen only by the user summary is contained in FN, while the number of frames in the model summary is represented by FP and is not connected to the user summary. With a threshold of 0.95, the best F1-score that the ResNet model could achieve was an average of 0.21. Tables 4 and 5 provide illustrations of the results, and Fig. 4 presents them graphically.

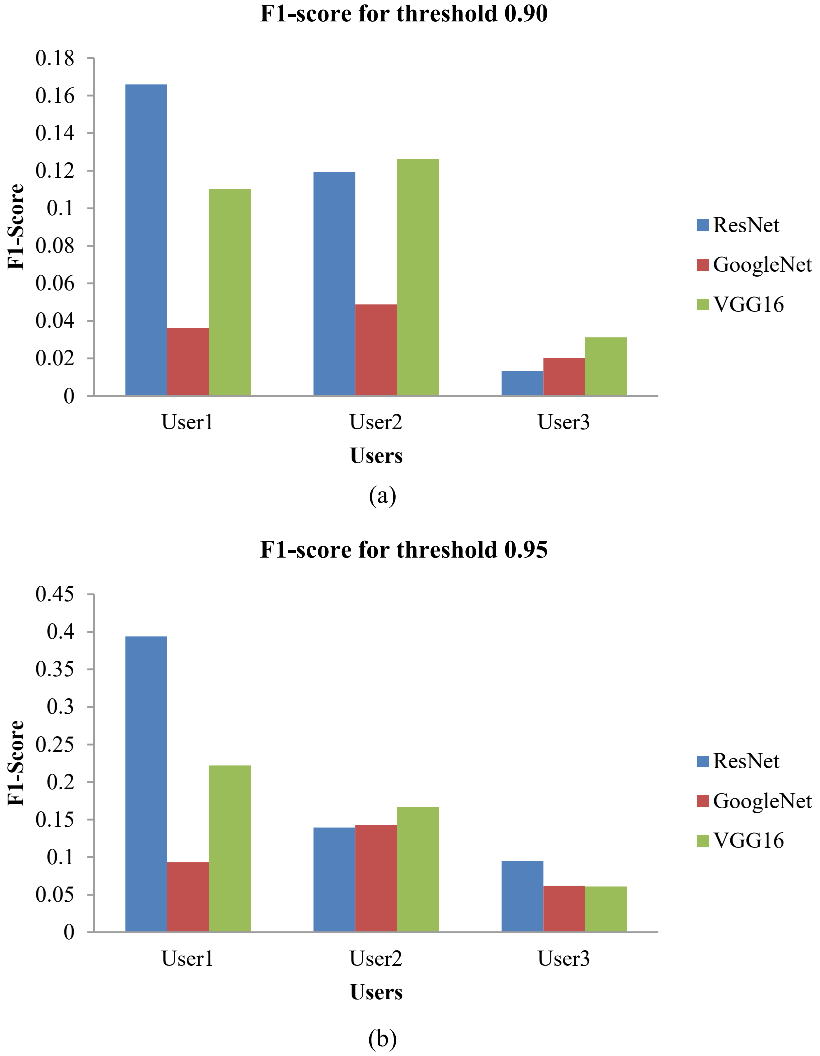
According to testing results, the GoogleNet model yields an ideal compression ratio, suggesting it would be the best option for video summarization; nevertheless, the quality of the summary is affected. When compared to other models, the ResNet model performs better in other evaluation parameters (F-score). The model still has a great deal of room for improvement.

Table 4 F1-score of three deep learning models considering $\delta = 0.90$

	ResNet	GoogleNet	VGG16
User1	0.165991903	0.036199095	0.110403397
User2	0.119402985	0.048780488	0.126126126
User3	0.013245033	0.02020202	0.03125
Average	0.10	0.04	0.09

Table 5 F1-score of three deep learning models considering $\delta = 0.95$

	ResNet	GoogleNet	VGG16
User1	0.393939394	0.093220339	0.222222222
User2	0.139534884	0.142857143	0.166666667
User3	0.094545455	0.062015504	0.060913706
Average	0.21	0.10	0.15



(a)

(b)

Fig. 4 Comparative analysis based on F-measure using cosine similarity for threshold **a** 0.90 and threshold **b** 0.95

5 Conclusion

Video summarization research is getting more and more essential in many different fields. In this work, we attempted to choose the key-frames using the cosine similarity measure, and we assessed our performance using the F1-score and CR metrics. According to the trial, GoogleNet produces the best results in terms of CR; nevertheless, this means that the quality of the summary must be sacrificed. As an evaluation parameter in state-of-the-art methods, the F-score in the ResNet-50 model performs better than that of other models. The suggested work restricts its visual modality and operates at the most fundamental level. The model can improve and generate a useful summary in future by learning multi-modality characteristics from the dataset. To grab the user's attention, a video summary with synchronized audio is a must. Despite the fact that a lot of work has been done in this field, there are still a lot of deficiencies that need to be fixed.

References

1. Ajmal M, Ashraf MH, Shakir M, Abbas Y, Shah FAS (2012) Video summarization: techniques and classification. In: Proceedings of the international conference on computer vision and graphics. Springer, Berlin, pp 1–13
2. https://medium.com/@bdhuma/applications-of-video-summarization-7fdadeb43b2d
3. Asim M, Almaadeed N, Al-maadeed S, Bouridane A, Beghdadi A (2018) A key frame based video summarization using color features. In: Colour and visual computing symposium (CVCS), Gjovik, Norway, pp 1–6. https://doi.org/10.1109/CVCS.2018.8496473
4. Jadon S, Jasim M (2020) Unsupervised video summarization framework using keyframe extraction and video skimming. In: IEEE 5th International conference on computing communication and automation (ICCCA), Greater Noida, India, pp 140–145. https://doi.org/10.1109/ICCCA49541.2020.9250764
5. Chunlei C, Guoliang L, Ruyun W, Chen L, Lei L, Peng Z, Hong L (2021) Graph-based structural difference analysis for video summarization. Inform Sci 57
6. Mendi E, Clemente HB, Bayrak C (2013) Sports video summarization based on motion analysis. Pergamon Press, Inc. United States. Comput Electri Eng 39(3):790–796. https://doi.org/10.1016/j.compeleceng.2012.11.020
7. Jesna M, Madhu N (2018) Dynamic summarization of videos based on descriptors in space-time video volumes and sparse autoencoders. IEEE Access
8. Mini Z, Guang H (2011) Video summarization by motion analysis: using optical flow technique. https://doi.org/10.1109/ICIII.2011.332
9. Sushma B, Aparna P (2021) Summarization of wireless capsule endoscopy video using deep feature matching and motion analysis. IEEE Access 9:13691–13703. https://doi.org/10.1109/ACCESS.2020.3044759
10. Harakannanavar SS, Sameer SR, Kumar V, Behera SK, Amberkar AV, Puranikmath VI (2022) Robust video summarization algorithm using supervised machine learning. Global Transitions Proc 3(1):131–135. ISSN 2666-285X. https://doi.org/10.1016/j.gltp.2022.04.009
11. Jie-Ling L, Yi Y (2012) Key frame extraction based on visual attention model. J Visual Commun Image Represent 23(1):114–125. ISSN 1047-3203. https://doi.org/10.1016/j.jvcir.2011.08.005
12. Shaoshuai L, Xie G, Yan G (2014) A novel key-frame extraction approach for both video summary and video index. The Scient World J 695168:9. https://doi.org/10.1155/2014/695168
13. Huang C, Wang H (2020) A novel key-frames selection framework for comprehensive video summarization. IEEE Trans Circuits and Syst Video Technol 30(2): 577–589. https://doi.org/10.1109/TCSVT.2019.2890899
14. Gawande U, Hajari K, Golhar Y (2020) Deep learning approach to key frame detection in human action videos. In: Recent Trends in Computational Intelligence.https://doi.org/10.5772/intechopen.91188
15. Savran Kızıltepe R, Gan JQ, Escobar JJ (2021) A novel keyframe extraction method for video classification using deep neural networks. Neural Comput Appl. https://doi.org/10.1007/s00521-021-06322-x
16. Man G, Sun X (2022) Interested keyframe extraction of commodity video based on adaptive clustering annotation. Appl Sci 12(3):1502. https://doi.org/10.3390/app12031502

17. Gygli M, Grabner H, Riemenschneider H, Van Gool L (2014) Creating summaries from user videos. In: Fleet D, Pajdla T, Schiele B, Tuytelaars T (eds) Computer vision—ECCV 2014. ECCV 2014, Lecture notes in computer science, vol 8695. Springer, Cham. https://doi.org/10.1007/978-3-319-10584-0_33
18. Mahasseni B, Lam M, Todorovic S (2017) Unsupervised video summarization with adversarial lstm networks. In: IEEE Conference on computer vision and pattern recognition (CVPR), Honolulu, HI, USA, pp 2982–2991. https://doi.org/10.1109/CVPR.2017.318
19. Zhao B, Li X, Lu X (2018) HSA-RNN: hierarchical structure-adaptive RNN for video summarization. In: IEEE/CVF Conference on computer vision and pattern recognition, Salt Lake City, UT, USA, pp 7405–7414. https://doi.org/10.1109/CVPR.2018.00773
20. Ghauri J, Hakimov S, Ewerth R (2021) Supervised video summarization via multiple feature sets with parallel attention. In: IEEE International conference on multimedia and expo (ICME). Shenzhen, China, pp 1–6s. https://doi.org/10.1109/ICME51207.2021.9428318
21. Zhao B, Li X, Lu X (2017) Hierarchical recurrent neural network for video summarization. In: Proceedings of the 25th ACM international conference on multimedia. https://doi.org/10.1145/3123266.3123328
22. Zhu W, Lu J, Li J, Zhou J (2021) DSNet: a flexible detect-to-summarize network for video summarization. In IEEE Trans Image Process 30:948–962. https://doi.org/10.1109/TIP.2020.3039886
23. Ping L, Chao T, Xianghua X (2021) Video summarization with a graph convolutional attention network. Front Inform Technol Electron Eng 22(6):902–913. https://doi.org/10.1631/FITEE.2000429
24. Zhao B, Gong M, Li X (2021) Hierarchical multimodal transformer to summarize videos. ArXiv, abs/2109.10559
25. Yaosen C, Bing G, Yan S, Renshuang Z, Weichen L, Wei W, Xuming W, Xinhua S (2022) Video summarization with u-shaped transformer. Appl Intell 52(15): 17864–17880. https://doi.org/10.1007/s10489-022-03451-1
26. Yoon UN, Hong MD, Jo G-S (2023) Unsupervised video summarization based on deep reinforcement learning with interpolation. Sensors 23(7):3384. https://doi.org/10.3390/s23073384
27. Tan K, Zhou Y, Xia Q, Liu R, Chen Y (2024) Large model based sequential keyframe extraction for video summarization. arXiv preprint arXiv:2401.04962
28. Hao T, Ding L, Wu S, Ren B, Sebe N, Rota P (2022) Deep unsupervised key frame extraction for efficient video classification. ACM (NY). ACM Trans Multimedia Comput Commun Appl. https://doi.org/10.1145/3571735