2020 IEEE 5th International Conference on Computing Communication and Automation (ICCCA)
Galgotias University, Greater Noida, UP, India. Oct 30-31, 2020

Unsupervised video summarization framework using keyframe extraction and video skimming

Shruti Jadon
College of Information and Computer Science
University of Massachusetts
Amherst, MA 01002
Email: sjadon@umass.edu

Mahmood Jasim
College of Information and Computer Science
University of Massachusetts
Amherst, MA 01002
Email: mjasim@cs.umass.edu

***Abstract*—Video is a robust sources of information and the consumption of online and offline videos has reached an unprecedented level in the last few years. However, the extraction of information from a video presents more challenges than the extraction of information from a picture. To extract the context of the video, a viewer has to go through the whole video. Apart from context understanding, it almost impossible to create a universal summarized video for everyone as everyone has their own bias of keyframe. Example, in a soccer game, a coach might prefer frames which consist of information on player placement and techniques. However, a person with less knowledge about soccer will focus more on frames which consist of goals and score-board. Therefore, if we were to tackle problem video summarization through a supervised learning path, it will require extensive personalized data labeling. In this paper, we attempt to solve video summarization through unsupervised learning by employing traditional vision-based algorithmic methodologies for accurate feature extraction from video frames. We have also proposed a deep learning based feature extraction followed by multiple clustering methods to find an effective way of summarizing a video by interesting keyframe extraction. We have compared the performance of these approaches on the SumMe dataset and showcased that using deep learning-based feature extraction has been proven to perform better in dynamic viewpoint videos.**

Keywords*—*Video Summarization, Vision, Deep Learning, Clustering, Image processing.

I. Introduction

With the recent advances in efficient data storage and streaming technologies, videos have become arguably the primary source of information in today's social media heavy culture and society.

Video streaming sites like YouTube are replacing the traditional methods of news and media sharing, whom, themselves are forced to adapt the trend of sharing videos on these streaming platforms rather than written articles. This change in information sharing methods brings forth challenges of developing an efficient way to extract information about subject matter video data. It's almost impossible to watch all video data thoroughly and catalog them according to their categories and subject matter. However, this metadata is extremely important when searching for a specific video. Currently, this categorization is dependent on the tags, titles etc. provided by the video owners. However, they are highly personalized and unreliable. Hence, a better way is required to create a summarized representation of the video that is easily comprehensible in a short amount of time. This is an open research problem in a multitude of fields including information retrieval, networking and computer vision.

Video Summarization can be defined as the process of compacting a video down to only important components in the video. The process is shown in Fig 1. This compact representation is useful in retrieving the desired videos from a large video library. A summarized video must have the following properties:

- It must contain the high priority entities and events from the video.
- The summary of the video should be free of repetition and redundancy.

Failure to exclude these components might lead to misinterpretation of the video from its summarized version. A summary of video also varies from person to person. A supervised video summarization can be seen as personalized recommendation approach, but, it requires extensive amount of data labeling by a person and yet the trained model isn't a generalized model among mass.

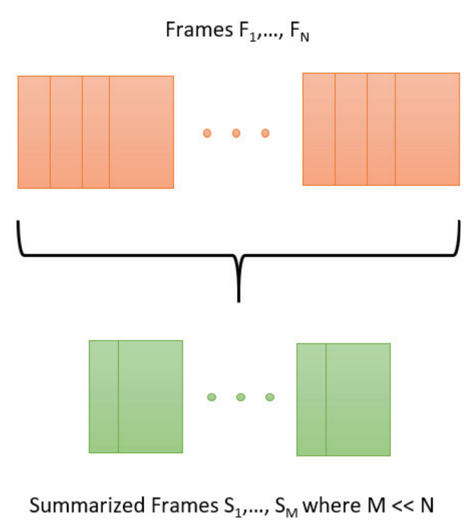

Fig. 1. The process of video summarization. N number of frames in the video is summarized to M number of frames where M is far smaller than N.

This work has been done as part of Computer Vision Coursework at UMass Amherst.

Various approaches have been taken to solve this problem by different researchers. Some of most prominent approaches include keyframe extraction using Visual Features [1] [2] and Video Skimming [3],[4]. In this paper, we will be exploring a generalized video summarization approach through unsupervised keyframe extraction. We propose a framework to extract vision-based keyframes and use them to create summarized videos with help of clustering and video skimming. We have used the SumMe dataset [5] for our experiments. Our key contributions include suggesting a new unsupervised framework for video summarization. We have first extracted features on basis of traditional computer vision filters and RESNET16 trained on image net. We used these extracted features for clustering to obtain keyframes. After choosing the keyframes, we just skimmed the video by adding 1.5 second worth of video around it.

This paper proposed an unsupervised video summarization framework to address the large scale video-based data management. The rest of the paper is organized as follows:

In Section II, the related research work, is discussed. Sections III & IV discuss our proposed technique for keyframe extraction and video skimming, followed by experiments. The paper is concluded with discussions and future goals in section V. Our code is available at Github (Link).

II. Related Research

The most difficult challenge of video summarization is determining and separating the important content from the unimportant content. The important content can be classified based on low level features like texture [6], shape [7] or motion [8]. The frames containing the important information are bundled together to create the summary. This manner of finding key information from static frames is called keyframe extraction. These methods are used dominantly to extract a static summary of the video. Some of the most popular keyframe extraction methods include [9], [10]. They use low level features and dissimilarity detection with clustering methods to extract static keyframes from a video. The clustering methods are used to extract the features that are worthwhile to be in the summary while irrelevant frames rich with low level features are discarded. Different clustering methods have been used by researchers to find interesting frames [9]. Some methods use web-based image priors to extract the keyframes, for example, [11], [12]. In recent years, deep reinforcement learning approaches has also been used to tackle video summarization. [13] proposed a deep summarization network with novel reward function that takes into account the diverse features. Similarly, with the popularity and wide use case of Generative Adversarial Networks, [14] proposed Attentive Conditional Generative Adversarial Networks to extract latent features for feature extraction part of video summarization.

While extracting static keyframes to compile a summary of the video is effective, the summary itself might not be pleasant to watch and analyze by humans as it will be discontinuous and with abrupt cuts and frame skips. This can be solved by video skimming which appears more continuous and the frame changees and cuts will be less abrupt. However, the process is more complex than simple keyframe extraction because a continuous flow of semantic information [15] and relevance is needed to be maintained for videos skimming. Some of the video skimming approaches include [1], which utilizes the motion of the camera to extract important information and calculates the inter-frame dissimilarities from the low level features to extract the interesting components from the video. A simple approach to video skimming is to augment the keyframe extraction process by including a continuous set from frames before and after the keyframe up to a certain threshold and include these collection frames in the final summary of the video to create an video skim.

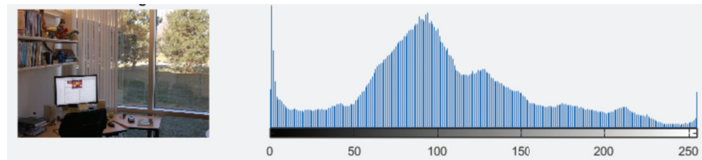

Fig. 2. A simple example of Image Histogram.

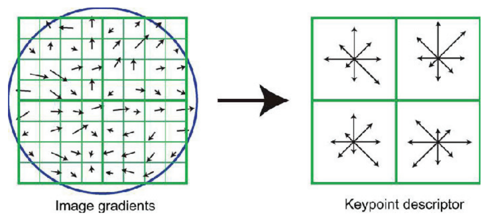

Fig. 3. A pictorial representation of SIFT key points extraction from 8X8 Image.

III. Approaches

In this paper, for static keyframe extraction, we extract low level features using uniform sampling, image histograms, SIFT and image features from Convolutional Neural Network (CNN) trained on ImageNet [16]. We also used two clustering methods: K-means and Gaussian clustering and chose the number of clusters on basis of summarized video size. We have used video skims around the selected keyframes to make the summary more fluid and comprehensible for humans. We take inspiration from the VSUMM method which is a prominent method in video summarization [17] [5].

A. *keyframe Extraction Techniques*

1) Uniform Sampling: Uniform sampling is one of the most common methods for keyframe extraction [18]. The idea is to select every kth frame from the video where the value of k is dictated by the length of the video. A usual choice of length for a summarized video is 5% to 15% of the original video, which means every 20th frame in case of 5% or every 7th frame in case of 15% length of the summarized video is chosen. For our experiment, we have chosen to use every 7th frame to summarize the video. This is a very simple concept which does not maintain semantic relevance. Uniform sampling is often considered as a generalized baseline for video summarization.

2) Image histogram: Image histograms [19] represent the total distribution of an image. It gives us the number of pixels for a specific brightness values rated from 0 to 256. Image histograms contain important information about images and they can be utilized to extract keyframes. We extract the

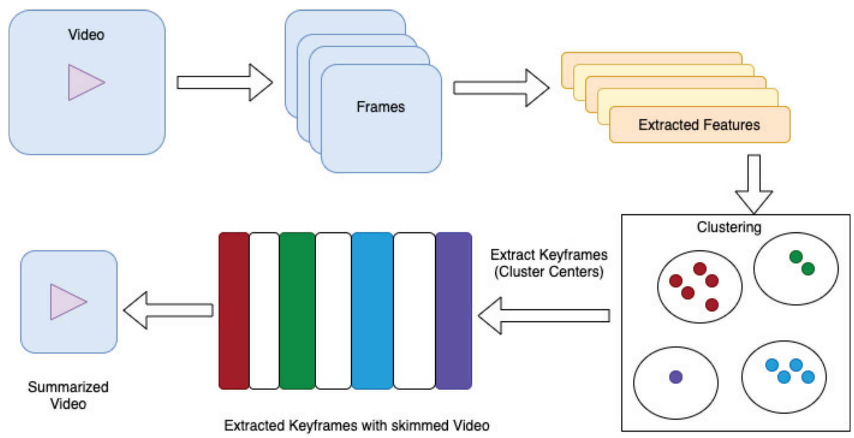

Fig. 4. Unsupervised Video Summarization Framework. Here, first video is converted into frames, followed by feature extraction. After feature extraction, they are plotted in an embeddings space, where they are clustered with % of summarized video as number of clusters. Last, we take center of these clusters as keyframes and apply video skimming to make a continuous summarized video.

histogram from all frames. Based on the difference between histograms of two frames, we decide whether the frames have significant dissimilarities among them. We infer that, a significant inter-frame image histogram dissimilarity indicates a rapid change of scene in the video which might contain interesting components. For our experiments, if histograms of two consecutive frames are 50% or more dissimilar, we extract that frame as a keyframe.

3) Scale Invariant Feature Transform: Scale Invariant Feature Transform (SIFT) [20], has been one of the most prominent local features used in computer vision is applications ranging from object and gesture recognition to video tracking. We use SIFT features for keyframe extraction. SIFT descriptors are invariant to scaling, translation, rotation, small deformations, and partially invariant to illumination, making it a robust descriptor to be used as local features. In SIFT, important locations are first defined using a scale space of smoothed and resized images and applying difference of Gaussian functions on these images to find the maximum and minimum responses. Non maxima suppression is performed and putative matches are discarded to ensure a collection of highly interesting and distinct collection of keypoints. Histogram of oriented gradients is performed by dividing the image into patches to find the dominant orientation of the localized keypoints. These keypoints are extracted as local features. In our experiment, we have extracted HOGs for each frame in video, and then put a threshold which could take 15% of video.

4) VSUMM: VSUMM has been one of the fundamental techniques in video summarization in the unsupervised setting. [17] proposed to extract features on basis of color histograms and use K-means algorithm to cluster features extracted from each frame. Color histograms are widely used in traditional vision, and a lot of other feature extraction methods are based out of them. These histograms are essentially a pixel classification approach where each pixels value of the image in the RGB channels gets stored to a bin of respective RGB channels, forming the histogram to be 3-D Tensor. Since each RGB channel based pixel values ranges in 0 255, generally, 16 bins are taken for each channel forming a 16X16X16 tensor. Due to computational reasons, a simplified version of this histogram was computed, where each channel was treated separately, resulting in feature vectors for each frame belonging to R 48 . This simplified color histograms give comparable performance to the true color histograms. Following this, we also experimented on VSUMM approach by altering the features extraction method from Color Histogram to transfer learning approach using VGG16 [21], and clustered the extracted features using k-means to analyze video based features. Our experiment results are shown in table I

5) ResNet16 on ImageNet: While reading about approach of VSUMM, we decided to test a different approach. We chose ResNet16 trained on image net with different range of filters and chopped of last loss layer so as to obtain the embeddings of each image (512 dimension). We extracted frames out of the videos, forward pass them through ResNet16 and after obtaining the embeddings for each frame in video, we clustered them using 2 algorithms: K-means and Gaussian Mixture Models. The number of cluster has been take as 15% of the video frame numbers. We later chose the frames closest to the center of clusters as the keyframes.

B. Clustering

1) K-means clustering: K-means clustering is a very popular clustering method. This algorithm consists of 2 steps: Expectation step, where we determine the mean of clusters, and Maximization step, where we map the points to the recently determined cluster. For our experiments, we had access to a set of image frames extracted by one of the methods mentioned in section III-A. Our goal is to partition these frames into different clusters, so that the within-cluster sum of squared difference is minimum. This is equivalent to minimizing the pairwise squared deviation of points in the same cluster. With this clustering we find the interesting frames to be included in the summarization and discard the ones that are rich in local features but contains less informative or interesting content.
For our paper, we have used K-means for clustering the features obtained from RESNET16 ImageNet trained method. We obtained 512 dimension vector for each frame in video and clustered them. We have set the number of cluster to be 15% of the video. After clustering, we chose the key points which was closest to the center of that specific cluster.

TABLE I. F1 SCORE VALUES OF 25 SUMME DATASET [5] VIDEOS WITH HUMAN'S ENTRY AS BASELINE FOLLOWED BY DIFFERENT APPROACHES OF KEYFRAME EXTRACTION

Video Name	Human (Avg.)	Uniform Sampling	SIFT	VSUMM(K-means)	VSUMM(Gaussian)	CNN (K-means)	CNN (Gaussian)
Base jumping	0.257	0.085	0.234	0.083	0.094	0.239	0.247
Bike Polo	0.322	0.071	0.196	0.078	0.065	0.204	0.212
Scuba	0.217	0.0145	0.144	0.145	0.172	0.195	0.184
Valparaiso Downhill	0.217	0.198	0.19	0.201	0.197	0.207	0.211
Bearpark climbing	0.217	0.161	0.146	0.156	0.142	0.196	0.204
Bus in Rock Tunnel	0.217	0.030	0.177	0.029	0.033	0.124	0.119
Car railcrossing	0.217	0.363	0.36	0.386	0.396	0.197	0.174
Cockpit Landing	0.217	0.089	0.035	0.906	0.856	0.965	0.984
Cooking	0.217	0.024	0.192	0.0231	0.0257	0.205	0.197
Eiffel Tower	0.312	0.119	0.004	0.123	0.135	0.157	0.146
Excavators river crossing	0.303	0.328	0.320	0.327	0.345	0.342	0.357
Jumps	0.483	0.176	0.161	0.174	0.185	0.182	0.176
Kids playing in leaves	0.289	0.426	0.366	0.424	0.482	0.372	0.384
Playing on water slide	0.195	0.168	0.232	0.174	0.185	0.278	0.297
Saving dolphines	0.188	0.212	0.121	0.229	0.257	0.247	0.217
St Maarten Landing	0.496	0.040	0.12	0.039	0.0254	0.059	0.068
Statue of Liberty	0.184	0.068	0.208	0.070	0.072	0.095	0.097
Uncut Evening Flight	0.35	0.253	0.256	0.251	0.274	0.278	0.295
paluma jump	0.509	0.048	0.092	0.047	0.048	0.049	0.049
playing ball	**0.271**	0.239	0.222	0.258	0.237	0.256	**0.258**
Notre Dame	0.231	0.229	0.23	0.223	0.021	0.0230	0.0227
Air Force One	**0.332**	**0.066**	0.065	0.063	0.061	0.045	**0.048**
Fire Domino	0.394	0.002	0.247	0.003	0.0020	0.0042	0.0035
car over camera	0.346	0.035	0.04	0.038	0.035	0.0458	0.0475
Paintball	0.399	0.224	0.23	0.233	0.245	0.297	0.304
Mean	**0.311**	**0.0152**	**0.171**	**0.155**	**0.1869**	**0.1765**	**0.212**

2) Gaussian Clustering (Mixture Model): Gaussian mixture models (GMM) are widely used method for higher-dimensional data clustering. Similar to KMeans, GMMs also uses Expectation-Maximization algorithm to obtain accurate clusters. In GMMs, EM algorithm optimization is also done in iterative manner of 2 steps: Expectation , and Maximization. In first step, it learns the mean and variance of the multivariate normal components and in second step it assigns the data points to the components. Using it, EM maximizes the posterior probability given the data, that is, given a fitted GMM, a cluster assigns query data to the component yielding the highest posterior probability. This method of assigning a data point to exactly one cluster is called hard clustering.
However, GMM clustering is more flexible because you can view it as a fuzzy or soft clustering method. Soft clustering methods assign a score to a data point for each cluster. The value of the score indicates the association strength of the data point to the cluster. As opposed to hard clustering methods, soft clustering methods are flexible in that they can assign a data point to more than one cluster.
In this paper, we used clustering on the embeddings obtained using RESNET16 trained network. we set the number of clusters to be 15% of the video, then chose the points which were closest to the center of the cluster.

C. Video Summarization

Our approach for video summarization is influenced by the VSUMM method [17]. Firstly, keyframes containing important information is extracted using one of the methods mentioned in section III-A. To reduce the computation time for video segmentation, a fraction of the frames were used. Considering the sequence of frames are strongly correlated, the difference from one frame to the next is expected to be very low when sampled at high frequencies, such as, 30 frames per second. Instead using a low frequency rate of 5 frames per second had insignificant effect on the results but it increased the computation speed by a significant margin. We used 5 frames per second as a sampling rate for our experiments and discarded the redundant frames.

After extracting all the keyframes, we perform a clustering on the frames to categorized them into interesting and uninteresting frames using one of the methods mentioned in section III-B. The cluster with the interesting frames were used to generate the summary of the video. The summary of the video was chosen to have the length of approximately 15% of the original video (Refer figure 4). But this summary was discontinuous and thus different from the way a human observer would evaluate the summary leading to poor scores as our evaluation method coincides with how a human being scores the summary. This problem was overcome by using a 1.8 second skims from the extracted interesting frame. This makes the summary continuous and easy to comprehend. The low frequency sampling of frames helps keep the size if the video in check.

IV. EXPERIMENTS AND RESULTS

A. Dataset

For our experimentation, we use the SumMe dataset [22] which was created to be used as a benchmark for video summarization. The dataset contains 25 videos with the length ranging from one to six minutes. Each of these videos are annotated by at least 15 humans with a total of 390 human summaries. The annotations were collected by crowd sourcing. The length of all the human generated summaries are restricted to be within 15% of the original video. Frames from two example videos, a) Air Force One and b) Play Ball is presented in Fig 5.

B. Evaluation Method

The SumMe dataset provides individual scores to each annotated frames. We evaluate our method by measuring the

a) Air_Force_One

b) Play Ball

Fig. 5. Two example videos from the SumMe dataset [5]. a) Air Force One has a fixed camera and mostly static background. b) Play Ball - has moving camera with dynamic background.

F-score from the set of frames that have been selected by our method. We compare the F-score to the human generated summaries to validate the effectiveness of our method. F-score can be defined as harmonic mean of precision and recall metrics. Both precision and recall captures the true positive rate with respect to false positive and false negative respectively. It can be written as :

$$F = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

This measure is approximately the average of the two when they are close, and is more generally the harmonic mean, which, for the case of two numbers, coincides with the square of the geometric mean divided by the arithmetic mean. There are several reasons that the F-score can be criticized in particular circumstances due to its bias as an evaluation metric. This is also known as the F_1 measure, because recall and precision are evenly weighted.

C. Results

We ran mentioned methods on the SumMe Dataset, and compared the F-scores obtained by them (as shown in Table I). Any algorithm is said to be effective if it is can achieve F1 score close to human, which we were able to obtain using SIFT, VSUMM, and CNN. To analyse on a bigger scale, we also took mean of scores for all videos, and observed that CNN(Gaussian) was performing the best followed by VSUMM(Gaussian). We have also noted that the videos which had dynamic view point was performing good with VSUMM and CNN, whereas, the videos with stable view point was performing very poorly, even in comparison to Uniform Sampling. This is where we can find difference in a human's method of summarizing vs an algorithm method. We can also see that SIFT's and CNN's have positive correlation in terms of F-scores this is due to the type of features obtained. Though, SIFT is not able to outperform CNN as CNN can extract more complex features.

V. Conclusion

Video Summarization is a challenging problem because it depends on a person's perception. We can never have a good baseline to understand whether our algorithm is working or not. Sometimes, a person prefers 1-2 second of video as summary, whereas an automated summary system might look for slightest difference in image intensity and give 10 seconds video as summary. In this work, we have take the baseline given in SumMe Dataset. We chose the average human baseline as true, as we would like to consider all perspectives. After experimenting with above mentioned unsupervised approaches on different videos, we can conclude that Gaussian Clustering along with Convolutional Networks can give better performance than other methods with moving point camera videos. We have also observed that the SIFT algorithm seems to perform well on videos with high motion as they extract transformation invariant features. On the other hand, Uniform Sampling performed better for videos which have stable camera view point and very less motion. Based on our experiments, we can conclude that there is no universal unsupervised summarization approach for every type of video.

References

[1] Y. Gong and X. Liu, "Video summarization using singular value decomposition," in *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No.PR00662)*, vol. 2, 2000, pp. 174–180 vol.2.

[2] B. Gong, W.-L. Chao, K. Grauman, and F. Sha, "Diverse sequential subset selection for supervised video summarization," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2014, pp. 2069–2077. [Online]. Available: http://papers.nips.cc/paper/5413-diverse-sequential-subset-selection-for-supervised-video-summarization.pdf

[3] K. Chorianopoulos, "Collective intelligence within web video," *Human-centric Computing and Information Sciences*, vol. 3, no. 1, p. 10, Jun 2013. [Online]. Available: https://doi.org/10.1186/2192-1962-3-10

[4] Z. Liu, E. Zavesky, B. Shahraray, D. Gibbon, and A. Basso, "Brief and high-interest video summary generation: Evaluating the at&t labs rushes summarizations," in *Proceedings of the 2Nd ACM TRECVid Video Summarization Workshop*, ser. TVS '08. New York, NY, USA: ACM, 2008, pp. 21–25. [Online]. Available: http://doi.acm.org/10.1145/1463563.1463565

[5] M. Gygli, H. Grabner, H. Riemenschneider, and L. Van Gool, "Creating summaries from user videos," in *ECCV*, 2014.

[6] T. Ahonen, A. Hadid, and M. Pietikainen, "Face description with local binary patterns: Application to face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2037–2041, Dec 2006.

[7] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, June 2005, pp. 886–893 vol. 1.

[8] A. Bruhn, J. Weickert, and C. Schnörr, "Lucas/kanade meets horn/schunck: Combining local and global optic flow methods," *International Journal of Computer Vision*, vol. 61, no. 3, pp. 211–231, Feb 2005. [Online]. Available: https://doi.org/10.1023/B:VISI.0000045324.43199.43

[9] N. Ejaz, I. Mehmood, and S. W. Baik, "Efficient visual attention based framework for extracting key frames from videos," *Signal Processing: Image Communication*, vol. 28, no. 1, pp. 34 – 44, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0923596512001828

[10] Y. J. Lee, J. Ghosh, and K. Grauman, "Discovering important people and objects for egocentric video summarization," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, June 2012, pp. 1346–1353.

[11] A. Khosla, R. Hamid, C. J. Lin, and N. Sundaresan, "Large-scale video summarization using web-image priors," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, June 2013, pp. 2698–2705.

[12] G. Kim, L. Sigal, and E. P. Xing, "Joint summarization of large-scale collections of web images and videos for storyline reconstruction," in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, ser. CVPR '14. Washington, DC, USA: IEEE Computer Society, 2014, pp. 4225–4232. [Online]. Available: http://dx.doi.org/10.1109/CVPR.2014.538

[13] K. Zhou, Y. Qiao, and T. Xiang, "Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[14] X. He, Y. Hua, T. Song, Z. Zhang, Z. Xue, R. Ma, N. Robertson, and H. Guan, "Unsupervised video summarization with attentive conditional generative adversarial networks," in *Proceedings of the 27th ACM International Conference on Multimedia*, ser. MM 19. New York, NY, USA: Association for Computing Machinery, 2019, p. 22962304. [Online]. Available: https://doi.org/10.1145/3343031.3351056

[15] S. JADON, *HANDS-ON ONE-SHOT LEARNING WITH PYTHON: A Practical Guide to Implementing Fast And... Accurate Deep Learning Models with Fewer Training*. PACKT PUBLISHING LIMITED, 2019. [Online]. Available: https://books.google.com/books?id=mRfDxQEACAAJ

[16] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[17] S. E. F. de Avila, A. P. B. a. Lopes, A. da Luz, Jr., and A. de Albuquerque Araújo, "Vsumm: A mechanism designed to produce static video summaries and a novel evaluation method," *Pattern Recogn. Lett.*, vol. 32, no. 1, pp. 56–68, Jan. 2011. [Online]. Available: http://dx.doi.org/10.1016/j.patrec.2010.08.004

[18] M. Nixon and A. Aguado, *Feature extraction and image processing for computer vision*. Academic press, 2019.

[19] Y. Wang, Q. Chen, and B. Zhang, "Image enhancement based on equal area dualistic sub-image histogram equalization method," *IEEE Transactions on Consumer Electronics*, vol. 45, no. 1, pp. 68–75, 1999.

[20] J.-M. Morel and G. Yu, "Is the scale invariant feature transform (sift) really scale invariant ?" 2010.

[21] S. Jadon, "Introduction to different activation functions for deep learning," *Medium, Augmenting Humanity*, vol. 16, 2018.

[22] M. Gygli, H. Grabner, H. Riemenschneider, and L. Van Gool, "Creating summaries from user videos," in *ECCV*, 2014.