EasyChair Preprint

№ 1181

Video Summarization using Keyframe Extraction and Video Skimming

Shruti Jadon and Mahmood Jasim

May 24, 2020

Video Summarization using Keyframe Extraction and Video Skimming

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

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

***Abstract*—Video is one of the robust sources of information and the consumption of online and offline videos has reached an unprecedented level in the last few years. A fundamental challenge of extracting information from videos is a viewer has to go through the complete video to understand the context, as opposed to an image where the viewer can extract information from a single frame. In this project, we attempt to employ different Algorithmic methodologies including local features and deep neural networks along with multiple clustering methods to find an effective way of summarizing a video by interesting keyframe extraction.**

***Keywords*—Video Summarization, Vision, Deep Learning.**

I. Introduction

Following the advances of 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 quickly replacing the traditional news and media sharing methods whom themselves are forced to adapt the trend of posting videos instead of written articles to convey stories, news and information. This abundance of videos include new challenges concerning an efficient way to extract the subject matter of the videos in question. It would be frustrating, inefficient, unintelligent and downright impossible to watch all movies thoroughly and catalog them according to their categories and subject matter, which is extremely important when searching for a specific video. Currently, this categorization is dependent on the tags, metadata or titles provided by the video uploaders. But these are highly personalized and unreliable in application and 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 of course computer vision.

Video summarization is the process of compacting a video down to only important components in the video. The process is shown in Fig 1. This compact representation can be useful when browsing a large number of videos and retrieve the desired ones efficiently. The summarized video must have the following properties, firstly, it must contain the high priority entities and events from the video and secondly, the summary should be free of repetition and redundancy. It is essential for the summarized video to capture all the important components, so that it represents the complete story of the video. Failure to exclude these components might lead to misinterpretation of the video from its summarized version. Also, redundant and unimportant components should be removed to make the summary compact and effective in representing the content properly.

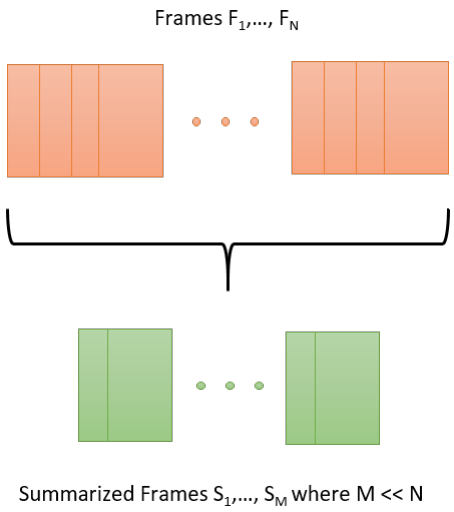

Fig. 1. The process of video summarization. N number of frames in the video is summarized to M number of frames where M is far smaller than N.

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 project, we explore the keyframe extraction. We also propose a clustering method to cluster the summarized videos. We use the SumMe dataset for our experimentation and results. Our contributions for this include suggesting a new unsupervised method of video summarization. We have experimented with a method which includes extracting frame based features using RESNET16 trained on image net, and then clustering them with different algorithms. Later, chosing the keyframes as the points which were closest to the center of each cluster.

The rest of this literature is organized as follows. The related research is presented in section II, followed by our approach in section III. We present our experimental results is section IV. The paper is concluded with discussions and future goals in section V.

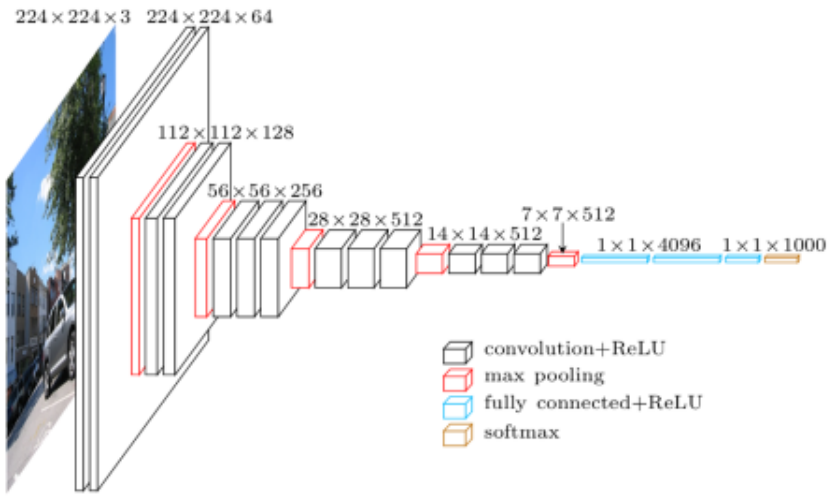

Fig. 2. Sample CNN Architecture for VSUMM and RESNET16

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 [5], shape [6] or motion [7]. The frames containing these 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 [8], [9]. These methods 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 extract features that are worthwhile to be in the summary while uninteresting frames rich with low level features are discarded. Different clustering methods have been used by researchers to find interesting frames [8]. Some methods use web-based image priors to extract the keyframes, for example, [10], [11].

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 will less abrupt frame changes and cuts. The process is more complex than simple keyframe extraction, however, because a continuous flow of semantic information [12] 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.

III. Approach

In this project we use both keyframe extraction and video skimming for video summarization. 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 [cite ImageNet]. We also use different clustering methods including K-means and Gaussian clustering. We use video skims around the selected keyframes to make the summary fore fluid and comprehensible for humans. We take inspiration from the VSUMM method which is a prominent method in video summarization [13].

A. Keyframe extraction

1) Uniform Sampling: Uniform sampling is one of the most common methods for keyframe extraction [cite uniform sampling]. 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 baseline for video summarization.

2) Image histogram: Image histograms represent the tonal 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 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.

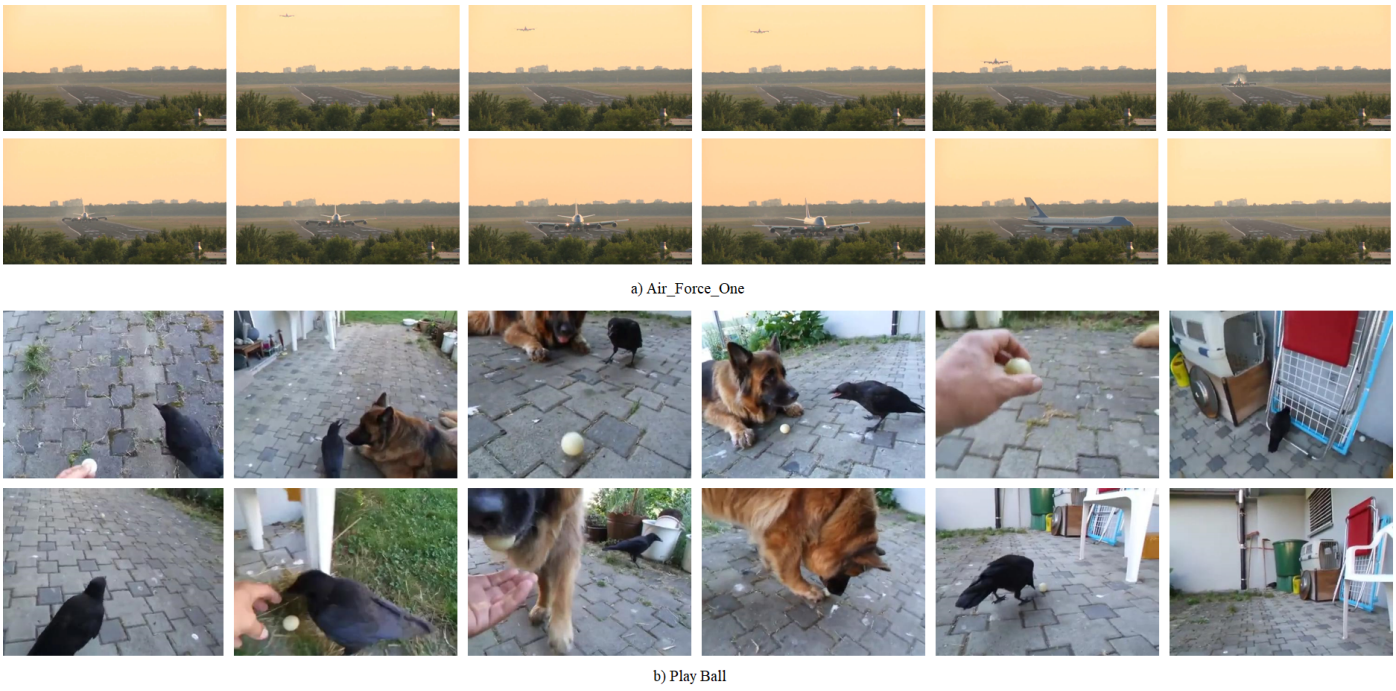

Fig. 3. Two example videos from the SumMe dataset. a) Air Force One has a fixed camera and mostly static background. b) Play Ball - has moving camera with dynamic background.

3) Scale Invariant Feature Transform: Scale Invariant Feature Transform (SIFT) [cite SIFT], 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. 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: This technique has been one of the fundamental techniques in video summarization in the unsupervised setup. The algorithm uses the standard K-means algorithm to cluster features extracted from each frame. Color histograms are proposed to be used in [13]. Color histograms are 3-D tensors, where each pixels values in the RGB channels determines the bin it goes into. Since each channel value ranges in 0 255, usually, 16 bins are taken for each channel resulting in 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 . The nest step suggested for clustering is slightly different. But, the simplified color histograms give comparable performance to the true color histograms. The features extracted from VGG16 at the 2nd fully connected layer [14] were tried, and clustered using kmeans.

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, and forward pass them through ResNet16, and after obtaining the embeddings for each frame in video, we clustered them using 2 algorithms: Kmeans , 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. A sample CNN architecture for VSUMM and RESNET16 is presented in Fig 2.

B. Clustering

1) K-means clustering: K-means clustering is a very popular clustering method. Given a set of image frames extracted by one of the methods mentioned in section III-A, the 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 project, we have used Kmeans 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

TABLE I. RESULTS

Video Name	Human (Avg.)	Uniform Sampling	SIFT	VSUMM(K-means)	VSUMM(Gaussian)	CNN (K-means)	CNN (Gaussian)
Base jumping	0.257	0.085364	0.234	0.083356	0.094	0.239	0.247
Bike Polo	0.322	0.07112	0.196	0.078369	0.065	0.204	0.212
Scuba	0.217	0.0145059	0.144	0.145599	0.172	0.195	0.184
Valparaiso Downhill	0.217	0.19899	0.19	0.201909	0.197	0.207	0.211
Bearpark climbing	0.217	0.160377	0.146	0.156611	0.142	0.196	0.204
Bus in Rock Tunnel	0.217	0.030199	0.177	0.029341	0.033	0.124	0.119
Car railcrossing	0.217	0.363804	0.36	0.386466	0.396	0.197	0.174
Cockpit Landing	0.217	0.089413	0.035	0.906021	0.856	0.965	0.984
Cooking	0.217	0.023748	0.192	0.023172	0.0257	0.205	0.197
Eiffel Tower	0.312	0.119034	0.004	0.123115	0.135	0.157	0.146
Excavators river crossing	0.303	0.328008	0.32	0.326871	0.345	0.342	0.357
Jumps	0.483	0.176244	0.16	0.174919	0.185	0.182	0.176
Kids playing in leaves	0.289	0.426775	0.366	0.424418	0.482	0.372	0.384
Playing on water slide	0.195	0.168675	0.232	0.174321	0.185	0.278	0.297
Saving dolphines	0.188	0.212642	0.121	0.229369	0.257	0.247	0.217
St Maarten Landing	0.496	0.0404343	0.12	0.039482	0.0254	0.059	0.068
Statue of Liberty	0.184	0.068651	0.208	0.070949	0.072	0.095	0.097
Uncut Evening Flight	0.35	0.253156	0.256	0.251676	0.274	0.278	0.295
paluma jump	0.509	0.048565	0.092	0.047268	0.048	0.049	0.049
playing ball	0.271	0.239955	0.222	0.258244	0.237	0.256	0.258
Notre Dame	0.231	0.229265	0.23	0.223917	0.021	0.0230	0.0227
Air Force One	0.332	0.066812	0.07	0.065103	0.061	0.065	0.048
Fire Domino	0.394	0.002603	0.247	0.003367	0.0020	0.0042	0.0035
car over camera	0.346	0.035693	0.04	0.038304	0.035	0.0458	0.0475
Paintball	0.399	0.224322	0.23	0.233006	0.245	0.297	0.304
mean	0.311	0.0152	0.171	0.155	0.1869	0.1765	0.212

15% of the video. After clustering, we chose the key points which was closest to the center of that specific cluster.

2) Gaussian Clustering (Mixture Model): Gaussian mixture models (GMM) [**?**] are often used for data clustering. Usually, fitted GMMs cluster by assigning query data points to the multivariate normal components that maximize the component 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 project, 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 [13]. 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. 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. EXPERIMENTAL RESULTS

A. Dataset

For our experimentation, we use the SumMe dataset [15] 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 3.

B. Evaluation Method

The SumMe dataset provides individual scores to each annotated frames. We evaluate our method by measuring the 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 is a measure that combines precision and recall is the harmonic

mean of precision and recall, the traditional F-measure or balanced F-score:

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

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 1) .Our main goal is to be as much close to human, which we were able to obtain using SIFT, VSUMM, and CNN. We also took mean of scores for all videos, and can see that CNN(Gaussian) was performing good followed by Vsumm. We observed 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 poor even with compared 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 features obtained. Though, SIFT is not able to outperform CNN.

V. Conclusion

Video clustering is one of the hardest task because it depends on person's perception. So, we can never have a good baseline to understand whether our algorithm is working or not. Sometimes, Humans just want 1-2 second of video as summary, whereas machine looks for slightest difference in image intensity and might give us 10 seconds of video.

From what the baseline has been given in SumMe Dataset, we chose the average human baseline as true, as we would like to consider all perspectives. After testing with all different forms of videos, we can conclude that Gaussian Clustering along with Convolutional Networks can give better performance than other methods with moving point camera videos. In fact, the SIFT algorithm seems to perform well on videos with high motion, the reason behind it is that we used deep layered features, thus they consists of important points inside image, followed by Gaussian Clustering, which is specifically made for mixture based components. We have also observed that, even Uniform Sampling is giving better result for videos which have stable camera view point and very less motion.
We can conclude that one single algorithm can't be solution of video summarization, it is dependent of the type of video, the motion inside 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] 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.

[6] 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.

[7] 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

[8] 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

[9] 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.

[10] 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.

[11] 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

[12] 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

[13] 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

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

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