Semantic Text Summarization of Long Videos

Shagan Sah[1*] Sourabh Kulhare[1] Allison Gray[1] Subhashini Venugopalan[2]
Emily Prud'hommeaux[1] Raymond Ptucha[1]
[1]Rochester Institute of Technology [2]University of Texas at Austin
[*]E-mail: sxs4337@rit.edu

Abstract

Long videos captured by consumers are typically tied to some of the most important moments of their lives, yet ironically are often the least frequently watched. The time required to initially retrieve and watch sections can be daunting. In this work we propose novel techniques for summarizing and annotating long videos. Existing video summarization techniques focus exclusively on identifying keyframes and subshots, however evaluating these summarized videos is a challenging task. Our work proposes methods to generate visual summaries of long videos, and in addition proposes techniques to annotate and generate textual summaries of the videos using recurrent networks. Interesting segments of long video are extracted based on image quality as well as cinematographic and consumer preference. Key frames from the most impactful segments are converted to textual annotations using sequential encoding and decoding deep learning models. Our summarization technique is benchmarked on the VideoSet dataset, and evaluated by humans for informative and linguistic content. We believe this to be the first fully automatic method capable of simultaneous visual and textual summarization of long consumer videos.

1. Introduction

Ease of use, instant sharing, and high image quality have resulted in abundant amounts video capture not only on social media outlets like Facebook and Youtube, but also personal devices including cell phones and computers. Several solutions are available to manage, organize, and search still images. Applying similar techniques to video works well for short snippets, but breaks down for videos over a few minutes long. While computer vision techniques have significantly helped in organizing and searching still image data, these methods do not scale directly to videos, and are often computationally inefficient. Videos that are tens of minutes to several hours long remain a major technical challenge. Ensuring that important moments are preserved, a proud parent may record long segments of their babys first birthday party. While the videos may have captured cherished moments, they may also include substantial amounts of transition time and irrelevant imagery.

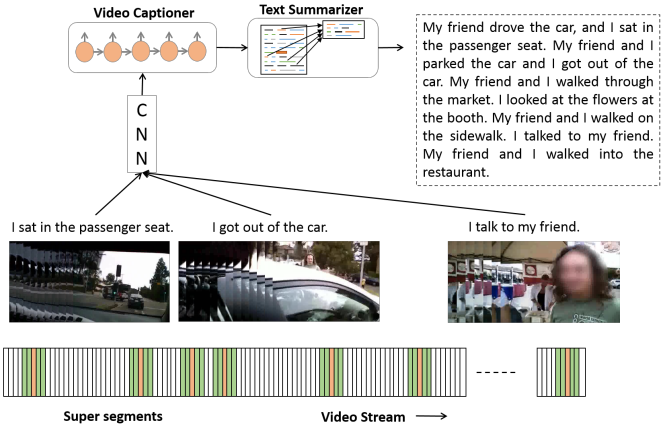

Figure 1. Overview of video summarization. Interesting regions identify key superframe segments. Each key segment is annotated. All annotations are fed into a text summarization module.

To mitigate these problems, we propose techniques that leverage recent advances in video summarization [21][7][15][40][16], video annotation [34][37][33], and text summarization [28][5], to summarize hour long videos to a substantially short visual and textual summary.

Our proposed method uniquely identifies interesting segments from long videos using image quality and consumer preference. Key frames are extracted from interesting segments whereby deep visual-captioning techniques generate visual and textual summaries. Captions from interesting segments are fed into extractive methods to generate paragraph summaries from the entire video. The paragraph summary is suitable for search and organization of videos, and the individual segment captions are suitable for efficient seeking to proper temporal offset in long videos. Because boundary cuts of interesting segments follow cinematography rules, the concatenation of segments forms a shorter summary of the long video. Our method provides

knobs to increase and/or decrease both the video and textual summary length to suit the application. While we evaluate our methods on egocentric videos and TV episodes, similar techniques can also be used in commercial and government applications such as sports event summarization or surveillance, security, and reconnaissance.

The novel contributions of this paper include: 1) The ability to split a video into superframe segments, ranking each segment by image quality, cinematography rules, and consumer preference; 2) Advancing the field of video annotation by combining recent deep learning discoveries in image classification, recurrent neural networks, and transfer learning; 3) Adopting textual summarization methods to produce human readable summaries of video; and 4) providing knobs such that both the video and textual summary can be of variable length.

The paper is organized as follows- Section 2 lists the related work, Section 3 describes the proposed methodology and Section 4 discusses the results.

2. Related Work

Video summarization research has been largely driven by parallel advancements in video processing methods, intelligent selection of video frames, and start-of-the-art text summarization tools. [25] generates story driven summary from long unedited egocentric videos. They start with a *static-transit* procedure to extract subshots from a longer egocentric video and extract entities that appear in each subshot to maximize a order of k selected subshots while preserving influence over time and individual important events. In contrast, [15] works with any kind of video (static, egocentric or moving), generates superframe cuts based on motion and further estimates interestingness of each superframe based on attention, aesthetic quality, landmark, person and objects. [31] uses video titles to find most important video segments. [40] explores a *nonparametric* supervised learning approach for summarization and transfers summary structure to novel input videos. Determinantal Point Process has also often been used in video summary methods [13],[41], [40].

Using key frames to identify important or interesting regions of video has proven to be a valuable first step in video summarization. For example, [7] used temporal motion to define a visual attention score. Similarly, [21] utilized spatial saliency at the frame level. [15] introduced cinematographic rules which pull segment boundaries to locations with minimum motion. [23] favored frames with higher contrast and sharpness, [3] favored more colorful frames, [10] studied people and object content, while [30] studied the role facial content plays in image preference. [10] further tracked objects across a long video to discover story content.

Large supervised datasets along with advances in recurrent deep networks have enabled realistic description of still images with natural language text [4][22][2][35]. The extension of this to video can be done by pooling over frames [34] or utilizing a fixed number of frames [37]. [37] uses a temporal attention mechanism to understand the global temporal structure of video, in addition they also use appearance and action features through a 3-D Convolutional Neural Network (CNN) which encode local temporal structure. Most recently, [33] described a technique, S2VT, to learn a representation of a variable sequence of frames which are decoded into natural text. Recently, [39] demonstrated a hierarchical recurrent neural network to generate paragraph summaries from relatively long videos. These videos were still limited to a few minutes long. We use a variation of the S2VT captioning approach in our work.

Given descriptive captions at key frame locations, we explore extractive methods for summarization. Extractive methods analyze a collection of input text to be summarized, typically sentences. These sentences are selected to be included in the summary using various measurements of sentence importance or centrality. Early seminal summarization research by Luhn [26] used word frequency metrics to rank sentences for inclusion in summaries, while Edmundson [6] expanded this approach to include heuristics based on word position in a sentence, sentence position in a document, and the presence of nearby key phrases. More recent extensions of the word frequency models, including SumBasic [29] and KL-Sum [32], typically incorporate more sophisticated methods of combining measures of word frequency at the sentence level and using these composite measures to rank candidate sentences. Other approaches, such as LexRank [8] and TextRank [27] focus on centroid-based methods of sentence selection, in which random walks on graphs of words and sentences are used to measure the centrality of those sentences to the text being summarized. A good review of these techniques and others can be found in [17][28]. The latest research on single document summarization has utilized both dependency based discourse tree trimming [19] as well as compression and anaphoricity constraints [5].

3. Methodology

Our proposed approach consists of four main components:

1. Identification of interesting segments from the full video;
2. Key frame extraction from these interesting segments;
3. Annotations for these key frames are generated using a deep video-captioning network; and
4. The annotations are summarized to generate a paragraph description of the sequence of events in the video.

The annotations from the key frames form powerful search descriptors, both for finding the appropriate video, and for quickly jumping to the appropriate frame location in the video. The selected interesting segments form a visual summary of the long video. The generated paragraph is the textual summary of the long video. Next, we describe each of these modules in detail.

3.1. Superframe Segmentation Framework

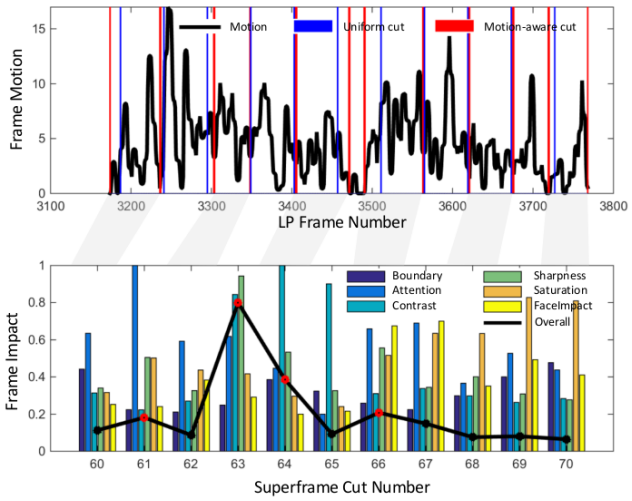

Figure 2. (top) The black trace shows frame-to-frame motion, the blue bars show evenly spaced boundaries, and red bars show the final selected superframe boundary cuts. (bottom) The corresponding superframes impact scores as bar graphs, overall interestingness score as a black line, and red pentagrams indicate selected superframes.

Most work on extracting key segments from video has been done on extracting aesthetically pleasing, informative, or interesting regions. Realizing these key segments will ultimately be stitched, we additionally observe cinematographic rules which prefer segment boundaries with minimum motion [15], which are termed superframe cuts.

As videos used in this research are several hours long, every ten frames are first averaged. The resulting low pass filtered and shortened video is split into s fixed length segments. Optical flow motion estimates are generated, then using cinematographic rules from Gygli et al. [15], the segment boundaries gravitate towards areas of local minimum motion. Figure 2 (top) shows eleven superframe cuts from a typical video.

3.1.1 Generating Superframe Cut Fitness Scores

Given s superframe cuts, we need to decide which are worthy of inclusion in the final summary, and which will be edited out. Worthiness will be determined by a non-linear combination of scores measuring a superframe cut's fitness regarding Boundary, Attention, Contrast, Sharpness, Saturation, and Facial impact. Each of these will be described next.

3.1.1.1. Boundary Score

A Boundary score, B is computed for each superframe region, where the score is inversely proportional to the motion at each boundary neighborhood. Similar to [12], we stack the optical flow between consecutive frames in the x- and y- directions. Motion is computed as $M(t)$ (see key frame selection section below), then given $M(t)$, $B = 1/M(t)$.

3.1.1.2. Attention Score

Each of these superframe regions are evaluated for aesthetic and interesting properties. Similar to [7][21], an Attention score, based on temporal saliency is first used. The Attention score, A is a combination of the superframe motion, m and variance, v, where m and v correspond to the mean and variance of all non-boundary frames motion in a superframe cut. The final Attention score $A = \alpha * m + (1 - \alpha) * v$, with $\alpha = 0.7$.

3.1.1.3. Contrast Score

The measures of Contrast, Sharpness, Saturation, and Facial impact are computed for all frames in each superframe cut and then averaged to report four values for each superframe cut. Similar to [23], a Contrast score is computed. To calculate the Contrast score, C, each frame in a superframe cut is converted to luminance, low pass filtered, and resampled to $64 \times width$, where 64 is the new height and $width$ is selected to preserve the aspect ratio of the frame. The Contrast score, C, is the standard deviation of luminance pixels.

3.1.1.4. Sharpness Score

Similar to [23], a Sharpness score is computed. To calculate a Sharpness score, E, the frames are converted to luminance, then divided up into 10×10 equally spaced regions. Using the center 7×7 regions, the standard deviation of luminance pixels is calculated three times centered on each region, where each of the three times a random shift is added, and the median of the three standard deviation values is reported for each of the 49 regions. The Sharpness score, E is the maximum of the 49 standard deviation values.

3.1.1.5. Colorfulness Score

Similar to [3], a Colorfulness score, S is computed. The frames are converted to HSV space, low pass filtered, resampled to $64 \times width$, where 64 is the new height and $width$ is selected to preserve the aspect ratio of the frame, then the mean saturation value from the frame is reported.

3.1.1.6. Facial Impact Score

Ptucha et al. [30] reported on the importance of facial content in imagery, and described a method for generating aesthetically pleasing crops of images containing facial information. Similar to Gygli et al. [15], but following the rules from [30], we compute a Face impact score, F which favors larger and more centrally located faces. Each face is assigned an impact score and the sum of all face scores is reported as a Face impact score, F.

To convert from pixels to a universal unit of measure, the size of a face, *FS* is normalized to the size of the image using:

$$FS = \frac{faceWidth^2}{(imageWidth \times imageHeight)} \quad (1)$$

where $faceWidth$ is the width of the face bounding box in pixels, or 2 $\times$ intraocular distance if bounding boxes are not square. Finally, following [30], the face size attribute, *FSA* is normalized to 0:1, centered on 0.5 for a typical face:

$$FSA = -72.4 * FS^3 + 27.2 * FS^2 - 0.26 * FS + 0.5. \quad (2)$$

For the face location, faces centered left-right and just above top-bottom center line are favored. The face centrality attribute, *FCA* is measured with respect to the 2D Gaussian:

$$FCA = \frac{1}{2\pi\sigma_x\sigma_y} e^{\frac{\frac{\delta_x^2}{\sigma_x^2} + \frac{\delta_y^2}{\sigma_y^2}}{x}} \quad (3)$$

where:

$\sigma_x = 2 \times imageWidth/3$;
$\sigma_y = imageHeight/2$;
$\delta_x = abs(faceCentroidX - imageWidth/2)$;
$\delta_y = abs(faceCentroidY - 3 \times imageHeight/5)$;
$faceCentroidX$ is the centroid column of the face region; and $faceCentroidY$ is the centroid row of the face region.

For high impact, faces need to have both high *FSA* and *FCA*. The face impact score for the entire image, F is $\sum FSA \times FCA$ for all detected faces in the image.

3.1.2 Fusing Scores

Empirical testing has shown that Attention (A), Contrast (C), and Sharpness (E) are essential elements to the usefulness and fidelity of a superframe region. After normalization, the product of these three scores are used to form a baseline score for each superframe region. Boundary motion (B), Saturation (S), and Face impact (F) increase this baseline score by $\eta(B + F) + \gamma(S)$, where $\eta = 0.35$ and $\gamma = 0.2$. The final measure of superframe cut interestingness score is computed as:

$$I_{score} = A \cdot C \cdot E + \eta(B + F) + \gamma(S) \quad (4)$$

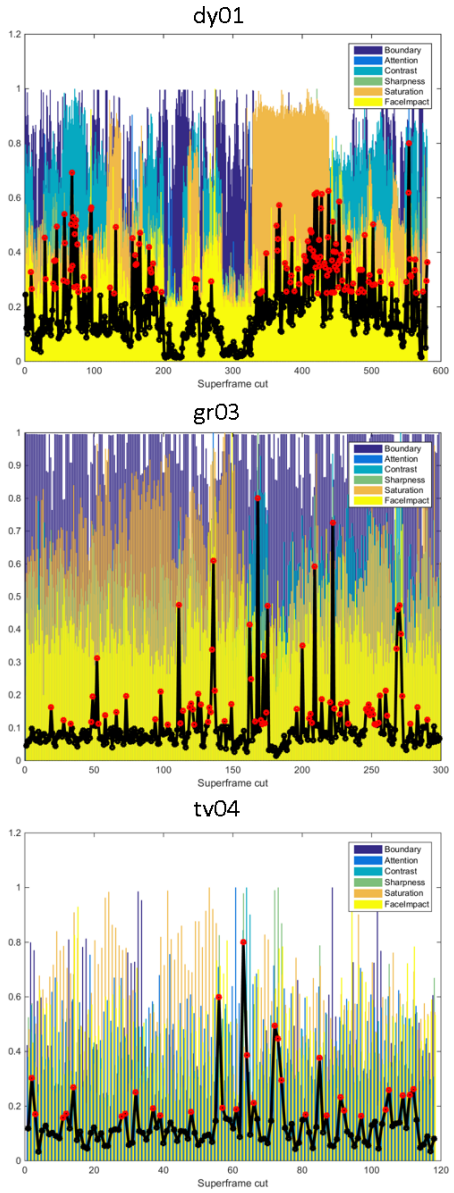

Figure 3. Impact scores for superframe cuts in the three test videos. Different colors represent contribution of features- Boundary, Attention, Contrast, Sharpness, Saturation and Face impact. X-axis is the superframe cut number and Y-axis is the normalized impact score. Solid black is I_{score}, red pentagrams show selected superframe cuts with $\omega = 50\%$. (Figure best viewed at 200%).

Figure 2 (bot) shows the corresponding superframe segments from Figure 2 (top), but with the individualized fitness scores and the overall I_{score} in solid black. After I_{score} is calculated for an entire video, the top superframe cuts (red pentagrams in Figure 2 (bot)) are selected by only using superframe cuts which comprise $\omega\%$ of the total energy. These selected superframe cuts define the region in

the original video which are used for visual and annotation summaries. Video summary duration can be altered by changing ω.

3.2. Key Frame Selection

For each selected superframe cut, we use optical flow displacement fields between consecutive frames to identify key frames [36]. A hierarchical time constraint ensures that fast movement activities are not omitted. The first step in identifying key frames is the calculation of optical flow for the entire superframe cut and estimate the magnitude of motion as a function of time. We use an OpenCV implementation [9] of optical flow to estimate motion. The function is calculated by aggregating the optical flow in the horizontal and vertical direction over all the pixels in each frame to calculate a magnitude of motion-

$$M(t) = \sum_i \sum_j \mid OF_x(i,j,t) \mid + \mid OF_y(i,j,t) \mid \quad (5)$$

where $OF_x(i,j,t)$ is the x component of optical flow at pixel i,j between frames t and $t-1$, and similarly for y component. As optical flow tracks all points over time, the sum is an estimation of the amount of motion between frames. The gradient of this function is the change of motion between consecutive frames and hence the local minimas and maximas represent important activities between sequences of actions. For capturing fast moving activities, a temporal constraint between two selected frames is applied during selection [11]. Frames are dynamically selected depending on the content of the video. Hence, complex activities or events would have more key frames, whereas simpler ones may have less.

3.3. Video Clip Captioning

Video clip captioning is achieved by modifying S2VT [33] with new frame features and introduction of key frame selection. Each key frame is passed through the 152-layer ResNet CNN model [18] pre-trained on ImageNet data, where the $[1 \times 2048]$ vector from the last pooling layer is used as a frame feature. These key frame feature vectors are passed sequentially into a Long Short Term Memory (LSTM) network [20], a recurrent neural network approach used in the speech recognition, language translation, as well as visual annotation. The S2VT framework first encodes f frames, one frame at a time to the first layer of a two layer LSTM, where f is of variable length. This latent representation is then decoded into a natural language sentence one word at a time, feeding the output of one time step into the second layer of the LSTM in the subsequent time step.

During training, a video sequence and corresponding text annotation pairs are input to the network. During testing, f key frames around a superframe video segment are

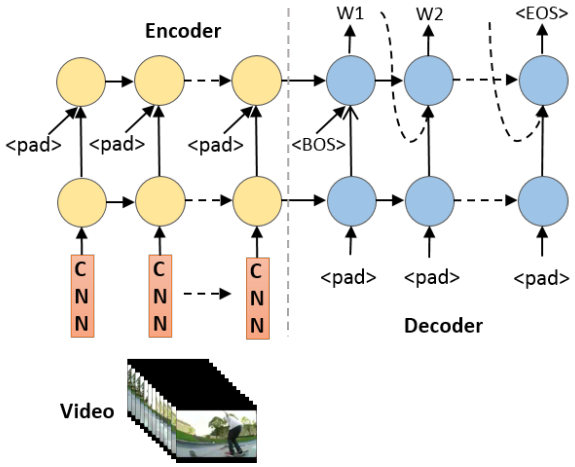

Figure 4. S2VT: A two layer LSTM model to learn video representation in the encoder and word representation in the decoder.

encoded into the trained neural network. Once all frames are processed, a begin of sentence keyword is fed into the network, triggering word generation until and end of sentence keyword is produced. The two layer LSTM is fixed to 80 time steps, which include both the input frames for each clip as well as its associated caption.

3.4. Text Summarization

The *sumy 0.4.1* python framework along with NLTK libraries were used to evaluate Luhn's algorithm, Edmundson's heuristic method, Latent Semantic Analysis (LSA), LexRank, TextRank, SumBasic and KL-Sum text summarization techniques. Before passing video clip captions into the text summarizers, duplicate captions were filtered out. The temporal order of each caption was preserved, and the summary length was fixed to 24 sentences for this paper, but can be changed to any length greater than the number of input captions.

In order to evaluate the summaries produced in this way, we turned to ROUGE [24], a set of objective metrics of summarization quality that can be calculated automatically, making them ideal for development and comparison of summaries generated by multiple summarization models. These metrics rely on methods of measuring word overlap between the output of a summarization system and one or more human generated reference summaries. Although simple, the ROUGE metrics correlate very highly with human evaluations. Here we use ROUGE-2, which measures the number of bigrams (i.e., two-word sequences) appearing in the summarization output that also appear in the reference summaries. ROUGE-2 is one of the more commonly used variation of the ROUGE metric in the text summarization research community and is the variant of ROUGE-N with the highest correlation with human evaluation. Using Lin's notation, ROUGE-2 is formulated as follows: where

$Refs$ is the set of reference summaries, $Count(bigram)$ is the count of a bigram, and $Count_{match}(bigram)$ is the number of matching bigrams in the summarization output:

$$ROUGE2 = \frac{\sum_{S \in Refs} \sum_{bigram \in S} Count_{match}(bigram)}{\sum_{S \in Refs} \sum_{bigram \in S} Count(bigram)} \quad (6)$$

3.5. Datasets

We demonstrate summarization on the VideoSet [38] dataset. This dataset is comprised of eleven long (45 minutes to over 5 hours) videos in three categories: Disney, egocentric, and TV episodes. Eight videos are used for training and three (DY01, GR03, TV04) for testing. The captioning model was pre-trained on the training split of the MSVD dataset [1] as the training data form VideoSet is not deemed sufficient.

4. Results and Discussion

Table 1 compares ROUGE 2 scores from the ground truth captions and summaries provided with the VideoSet dataset using several text summarization methods. The ground truth annotations for each five/five/ten second segments for the egocentric/Disney/TV videos, respectively, were compared to a single ground truth summary for each video. These results can be considered as the *upper bound* of the summarization methods, which suggest that the LexRank, LSA, and SumBasic methods are generally performing best.

Table 1. ROUGE 2 scores (higher is better) for VideoSet dataset. (lu= Luhn, ed=Edmundson, lsa=LSA, tr = text-rank, lr = LexRank, sb = SumBasic)

Video	lu	ed	lsa	tr	lr	sb	kl
DY01	0.32	0.26	**0.42**	0.20	0.29	0.36	0.18
GR03	0.21	0.20	0.22	0.15	0.16	**0.23**	0.16
TV04	0.35	0.14	**0.38**	0.22	0.18	0.16	0.11

After training, text summarization was applied to the three VideoSet test videos: DY01 a 5.5 hour video recorded by a Walt Disney World tourist; GR03 a 3 hour video depicting everyday activities; and TV04 a 45 minute episode of the TV show *Numb3rs*. Table 2 indicates strong benefits to using our key superframe segments. The TV04 was the shortest video and the summary contained numerous unique reference to names which cannot be learned from the training set. The summary of this video had numerous character and character usage errors, most likely due to the lack of training data to learn faces and appearances.

4.1. Human Evaluations

We created a task in which ten human judges rated our machine generated text summaries for overall summary semantics, sentence syntax, and sentence semantics on a 1 (very poor) - 5 (very good) Likert-type scale. The questions asked to the human judges were-

- After reading the summary, would you be able to describe the video to another person.
- Rate the quality of the syntax/grammar of the summary sentences (missing words, word order, incorrect words, unknown words, punctuation, upper/lower case, duplicate words/sentences).
- Rate the quality of the semantics/clarity/understanding of the summary sentences.

For overall summary and sentence syntax, the LSA and LexRank methods were preferred. For sentence semantics, all methods performed comparably. Judges rated the TV04 summaries much lower than DY01 and GR03.

Table 2. ROUGE 2 scores for machine generated vs. ground truth on VideoSet test videos. (LSA/LexRank/SumBasic methods)

Test Video	All Clips	Key Clips
DY01	0.25 / 0.17 / 0.21	0.31 / 0.30 / 0.31
GR03	0.15 / 0.07 / 0.14	0.15 / 0.11 / 0.15
TV04	0.02 / 0.02 / 0.02	0.01 / 0.01 / 0.01

Table 3. Example of a machine generated summary for DY01 video using LSA. (<en_unk> indicates that the model generated a word representation not found in the trained dictionary.)

I used my phone while waiting for the tram to depart. I looked through the attendant and i rode the tram. My friends and i waited for the tram to depart. My friends and i stood around the tour guide. My friends and i posed for a group picture. My friends and i talked about our day while walking around the park. My friends and i waited in the <en_unk> <en_unk> talking to the theater. My friends and i listened to the tour guide. I talked on my phone while walking around the park. My friends and i talked while moving along the line. I stood with a group of my friends talking. My friends and i walked through a dark room. My friends and i talked about our food while walking around the park. My friend and i talked about the camera while walking around the park. My friends and i talked about our camera while waiting around the park. My friends and i walked with our group leader through the park while talking. I stood in a dark place and talked to my friends. I walked through a dark room talking with my friends. I watched a mascot entertain i waiting. I grabbed some food while moving along the line. My friends and i sat at the table and had dinner. My friends and i waited at the table and had dinner. I watched a mascot entertain another group. My friends and i sat at the table and talked.

Table 4. Human evaluation scores on machine generated video summaries using LSA.

	DY01	GR03	TV04
Summary Semantics	3.65	2.35	1.40
Sentence Syntax	3.55	2.40	1.65
Sentence Semantics	3.80	2.35	1.45
Average	3.67	2.37	1.50

4.2. Evaluating Superframe Cut Selection

We use the SumMe Dataset [15] to evaluate the effectiveness of our features in superframe cut selection. The SumMe Dataset consists of 25 videos, ranging from one to seven minutes (950 to 9721 frames). An ablation analysis across the six features of Boundary, Attention, Contrast, Sharpness, Saturation, and Face impact was performed across all 25 videos. A five frame averaging filter was used, and then every 10th frame was extracted and resampled so frame width=480 pixels. The mean value for each feature in each superframe cut along with the mean ground truth relevance score was passed into the ablation analysis. A mean squared error from a linear regression model was used as a fitness criterion.

Table 5. Feature evaluation on SumMe dataset. Mean rank position (lower is better) ; number of times feature was selected 1st; 1st or 2nd; and 1st, 2nd, or 3rd.

Feature	Mean rank	top-1	top-2	top-3
Contrast	2.72 +/- 2.19	7	8	12
Saturation	2.80 +/- 2.16	6	8	10
Boundary	2.92 +/- 1.75	1	6	12
Face impact	2.92 +/- 1.89	1	9	11
Sharpness	3.12 +/- 2.01	3	6	11
Attention	3.24 +/- 2.01	3	7	9

Both the mean rank and top-k ranked columns of Table 5 show all features have significant usefulness in superframe cut selection. Although the Contrast and Saturation features have the lowest rank, the top-3 column shows the balanced nature of all the features. While the Boundary feature was an average performer, the human annotators rated each frame independently, not taking into account cinematographic rules. While the Face impact was found to be one of the most important factors in [30], only 12 out of 25 videos contained faces in this dataset. The low performance of Attention is surprising, and follow-on research finds the frame averaging is critical towards achieving high importance of the Attention score. For the SumMe dataset, the six features had an overall RMSE of 0.0271 as compared to the ground truth, showing this suite of features are excellent indicators of frame relevance.

4.3. Evaluating Key Frame Selection

We use the Keyframe-Sydney (KFSYD) Dataset [14] to evaluate the motion magnitude based key frame election. This dataset consists of ten videos, each with three independent sets of ground truth frame summaries. Table 6 reports the ratio of selected key frames that match with ground truth. A frame is considered a match if it is within n-neighborhood of a ground truth frame. top-k refers to matching k-highest probability frames with ground truth. Results reported in the table are averaged over all videos and all ground truth summaries.

Table 6. Evaluation scores for key frame selection. High ratio is better.

top-k	15-neighbor	25-neighbor
top-8	0.50	0.66
top-16	0.54	0.69
top-24	0.60	0.72
top-32	0.60	0.72

5. Conclusion

This paper introduces a novel method for both video summarization and annotation. Frame to frame motion, frame image quality, as well cinematographic and consumer preference are uniquely fused together to determine interesting segments from long videos. Key frames from the most impactful segments are converted to textual annotations using an encoder-decoder recurrent neural network. Textual annotations are summarized using extractive methods where LSA, LexRank and SumBasic approaches performed best. Human evaluations of video summaries indicate promising results. Independent experiments validate both superframe cuts as well as key frame selection. A key limitation is passing of incorrect superframe or key frame information to the captioning framework. A potential solution would be availability of datasets with ground truth on both key segments and associated captions/summaries.

Acknowledgement

We would like to thank NVIDIA for donating some of the GPUs used in this research.

References

[1] D. L. Chen and W. B. Dolan. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 190–200. Association for Computational Linguistics, 2011.

[2] X. Chen and C. L. Zitnick. Learning a recurrent visual representation for image caption generation. 2015.

[3] R. Datta, D. Joshi, J. Li, and J. Z. Wang. Studying aesthetics in photographic images using a computational approach. In *ECCV 2006*, pages 288–301. Springer, 2006.

[4] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE CVPR*, pages 2625–2634, 2015.

[5] G. Durrett, T. Berg-Kirkpatrick, and D. Klein. Learning-based single-document summarization with compression and anaphoricity constraints. *arXiv preprint arXiv:1603.08887*, 2016.

[6] H. P. Edmundson. New methods in automatic extracting. *Journal of the ACM (JACM)*, 16(2):264–285, 1969.

[7] N. Ejaz, I. Mehmood, and S. W. Baik. Efficient visual attention based framework for extracting key frames from videos. *Signal Processing: Image Communication*, 28(1):34–44, 2013.

[8] G. Erkan and D. R. Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, pages 457–479, 2004.

[9] G. Farnebäck. Two-frame motion estimation based on polynomial expansion. In *Image analysis*, pages 363–370. Springer, 2003.

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

[11] A. Girgensohn and J. Boreczky. Time-constrained keyframe selection technique. In *Multimedia Computing and Systems, 1999. IEEE International Conference on*, volume 1, pages 756–761. IEEE, 1999.

[12] G. Gkioxari and J. Malik. Finding action tubes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 759–768, 2015.

[13] 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*, pages 2069–2077, 2014.

[14] G. Guan, Z. Wang, S. Lu, J. D. Deng, and D. D. Feng. Keypoint-based keyframe selection. *IEEE Tran on Circuits and Systems for Video Technology*, 23(4):729–734, 2013.

[15] M. Gygli, H. Grabner, H. Riemenschneider, and L. Van Gool. Creating summaries from user videos. In *European conference on computer vision*, pages 505–520. Springer, 2014.

[16] M. Gygli, H. Grabner, and L. Van Gool. Video summarization by learning submodular mixtures of objectives. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3090–3098, 2015.

[17] A. Haghighi and L. Vanderwende. Exploring content models for multi-document summarization. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 362–370. Association for Computational Linguistics, 2009.

[18] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.

[19] T. Hirao, Y. Yoshida, M. Nishino, N. Yasuda, and M. Nagata. Single-document summarization as a tree knapsack problem. In *EMNLP*, volume 13, pages 1515–1520, 2013.

[20] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[21] X. Hou, J. Harel, and C. Koch. Image signature: Highlighting sparse salient regions. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(1):194–201, 2012.

[22] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE CVPR*, pages 3128–3137, 2015.

[23] Y. Ke, X. Tang, and F. Jing. The design of high-level features for photo quality assessment. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 419–426. IEEE, 2006.

[24] C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out: Proceedings of the ACL-04 Workshop*, volume 8. Spain, 2004.

[25] Z. Lu and K. Grauman. Story-driven summarization for egocentric video. In *Proceedings of the IEEE CVPR*, pages 2714–2721, 2013.

[26] H. P. Luhn. The automatic creation of literature abstracts. *IBM Journal of research and development*, 2(2):159–165, 1958.

[27] R. Mihalcea and P. Tarau. Textrank: Bringing order into texts. 2004.

[28] A. Nenkova, S. Maskey, and Y. Liu. Automatic summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts of ACL 2011*, page 3. Association for Computational Linguistics, 2011.

[29] A. Nenkova and L. Vanderwende. The impact of frequency on summarization. *Microsoft Research, Redmond, Washington, Tech. Rep. MSR-TR-2005-101*, 2005.

[30] R. Ptucha, D. Kloosterman, B. Mittelstaedt, and A. Loui. Automatic image assessment from facial attributes. In *IS&T/SPIE Electronic Imaging*, pages 90200C–90200C. International Society for Optics and Photonics, 2014.

[31] Y. Song, J. Vallmitjana, A. Stent, and A. Jaimes. Tvsum: Summarizing web videos using titles. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5179–5187, 2015.

[32] L. Vanderwende, H. Suzuki, C. Brockett, and A. Nenkova. Beyond sumbasic: Task-focused summarization with sentence simplification and lexical expansion. *Information Processing & Management*, 43(6):1606–1618, 2007.

[33] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko. Sequence to sequence – video to text. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015.

[34] S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko. Translating videos to natural language using deep recurrent neural networks. 2015.

[35] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3156–3164, 2015.

[36] W. Wolf. Key frame selection by motion analysis. In *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on*, volume 2, pages 1228–1231. IEEE, 1996.

[37] L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, and A. Courville. Describing videos by exploiting temporal structure. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4507–4515, 2015.

[38] S. Yeung, A. Fathi, and L. Fei-Fei. Videoset: Video evaluation through text. *arXiv preprint arXiv:1406.5824*, 2014.

[39] H. Yu, J. Wang, Z. Huang, Y. Yang, and W. Xu. Video paragraph captioning using hierarchical recurrent neural networks. *arXiv preprint arXiv:1510.07712*, 2015.

[40] K. Zhang, W.-L. Chao, F. Sha, and K. Grauman. Summary transfer: Exemplar-based subset selection for video summarizatio. *arXiv preprint arXiv:1603.03369*, 2016.

[41] K. Zhang, W.-L. Chao, F. Sha, and K. Grauman. Video summarization with long short-term memory. *arXiv preprint arXiv:1605.08110*, 2016.