LEARNING DEEP SEMANTIC ATTRIBUTES FOR USER VIDEO SUMMARIZATION

Ke Sun[1], *Jiasong Zhu*[2], *Zhuo Lei*[1], *Xianxu Hou*[1],*Qian Zhang*[1], *Jiang Duan*[3], *Guoping Qiu*[1]

[1]The University of Nottingham, Ningbo China
[1](zx17889, Xianxu.Hou, Zhuo.Lei, Qian.Zhang, Guoping.Qiu)@nottingham.edu.cn
[2]Shenzhen University, China
[2]zhujiasong@gmail.com
[3]Southwestern University of Finance and Economics, China
[3]duanj_t@swufe.edu.cn

ABSTRACT

This paper presents a Semantic Attribute assisted video SUMmarization framework (SASUM). Compared with traditional methods, SASUM has several innovative features. Firstly, we use a natural language processing tool to discover a set of keywords from an image and text corpora to form the semantic attributes of visual contents. Secondly, we train a deep convolution neural network to extract visual features as well as predict the semantic attributes of video segments which enables us to represent video contents with visual and semantic features simultaneously. Thirdly, we construct a temporally constrained video segment affinity matrix and use a partially near duplicate image discovery technique to cluster visually and semantically consistent video frames together. These frame clusters can then be condensed to form an informative and compact summary of the video. We will present experimental results to show the effectiveness of the semantic attributes in assisting the visual features in video summarization and our new technique achieves state-of-the-art performance.

***Index Terms*—** Video Summarization, Deep Convolution Neural Network, Semantic Attribute, Bundling Center Clustering

1. INTRODUCTION

The rapid proliferation of online video contents in recent years has created a demand for methods to perform effective video management and retrieval. However, the user-defined crowd-sourcing data such as titles, annotations and thumbnails often fail to provide specific representation of the abundant visual content, which could lead to unsatisfactory retrieval performance.

A possible remedy for this problem is to adopt automatic video summarization, which provides a synopsis for the given video content, saving time and money cost both for users and enterprises that provide video based services. Early work on automatic video summarization mainly focuses on certain domains such as sports [1] and news [2] videos, and generates summaries by leveraging domain-specific knowledge during the analysis process. However, most of these approaches only consider visual features, while the high-level semantics are often ignored.

To address this problem, some recent approaches attempt to introduce manually defined semantics to help generate video summaries, for example, interestingness [3], categoric knowledge from web images [4] and titles of user video [5], etc. An implicit assumption under this line of work is that these crowd-sourcing data, such as tags, categories, user titles are correctly given. But due to the subjectivity nature of the problem, different tags and titles are often used to describe the same or similar video contents, and some of them are even irrelevant. Such imperfections make these methods less applicable without proper human supervision.

We address this issue by developing a novel approach to automatically obtain joint (visual and semantic) feature representation from the video themselves without having to explicitly rely on human supervision. We do it by introducing a set of semantic attributes. Each semantic attribute is in the form of a unique word discovered from a database consisting of web images and associated text captions. We then train a deep convolution neural network for extracting visual features as well as predicting the semantic attributes of video segments. Based on the observation that adjacent video frames almost inevitably contain partially duplicate objects or regions, we adapt the partially duplicate image discovery technique, bundling center clustering method [6], for generating the final video summary.

The main contributions of this work are:

We propose to use semantic attributes to help capture high-level semantics in the video contents. These semantic attributes are automatically discovered from a joint image and text corpora.

We present a Semantic Attribute assisted video SUMmarization framework (SASUM) that is able to automatically find important shots from a video by leveraging joint visual and semantic features.

978-1-5090-6067-2/17/$31.00 ©2017 IEEE

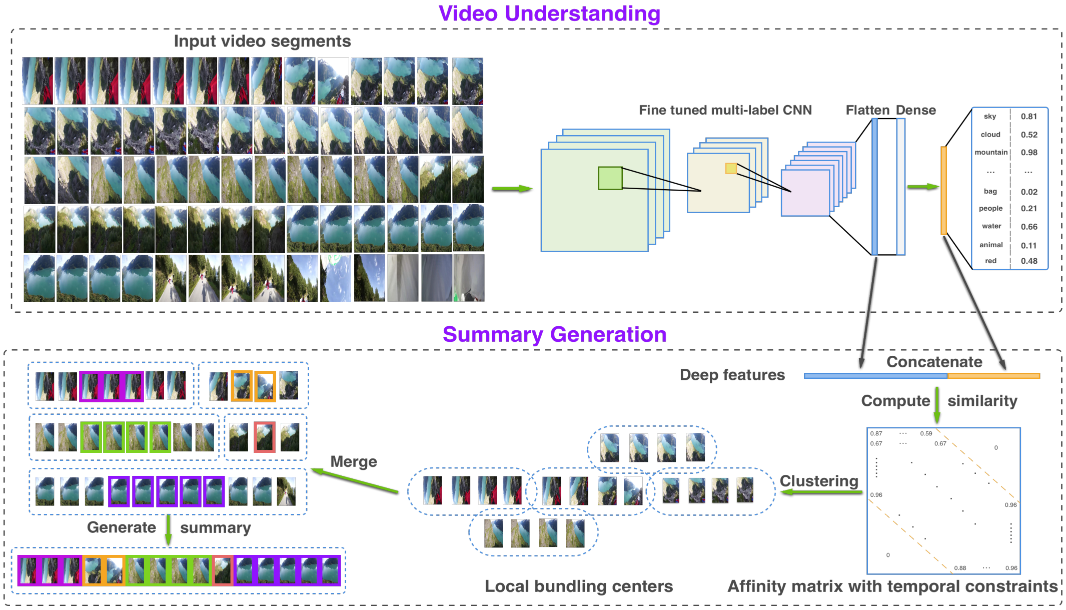

Fig. 1. Overview of our video summarization framework. For video understanding, we train a deep neural network for predicting a set of semantic attributes. We then compute deep features of the input video segments and construct the affinity matrix based on the pairwise similarity and temporal constraints. For summary generation, we cluster the whole sequence of segments into several continuous groups and then concatenate the central part of some segments in each group to obtain the final video summary.

We evaluate our SASUM framework on the popular SumMe [3] benchmark, and results show that our approach achieves state-of-the-art performance. We further show that automatically generated video summaries exhibit good correlation with manually created summaries on some videos.

2. THE SASUM FRAMEWORK

Our video summarization framework is illustrated in Figure 1. Given an input video segment, we first use a trained deep neural network to extract the joint deep features, and then apply a clustering method to divide the whole sequence of segments into several continuous groups. Finally, we select a few segments from each group and concatenate them following the temporal order to form the final video summary.

2.1. Learning Semantic Attributes

As mentioned in Section 1, we want to use semantic attributes to capture the semantics in videos. In stead of using hand-labeled data, our semantic attributes are extracted from image captions, including object classes, appearance & properties and motions. The question is how to discover such attributes. We do it by examining a joint image and text corpora, where images come with human-annotated captions. Different from [7] who directly extracts semantic features from the raw image captions, we use Stanford Corenlp toolkit [8] to automatically extract a set of attributes (words) from the captions of the training images in the Microsoft COCO [9] dataset. We then retain T most frequent words as our semantic attributes. In order to avoid overgrowing the dimension of our attribute vocabulary, we only consider the ”lemma” form of a word, for example, ”spots”, ”spotted” and ”spotting” are all treated as ”spot”.

We retain T most frequently occurred words and use them as our semantic attributes. Given that attribute vocabulary, we can easily replace the original caption of an image with a small set of attributes. We then wish to train a predictor to predict the attributes of a given video segment. Considering that each segment may contain multiple attributes and some attributes may only apply to sub-regions of the segment, we treat the prediction task as a region-based multi-label classification problem [10].

Figure 2 summarizes our deep neural network for attribute prediction. We adopt the powerful ResNet [11] pre-trained on ImageNet [12] as our base model. We modify the structure of ResNet by inserting a fully-connected layer with 1024 neurons before the output layer and then change the target output for multi-label prediction. In the fine-tuning phase, the output of the fully-connected layer are passed to a T-way softmax function. For objective function, we use the popular binary cross-entropy loss. Suppose we have N training samples and corresponding attribute based annotations, we then use $y_n = [y_{n1}, y_{n2}, ...y_{nt}]$ to denote the attribute vector of the n^{th} training image, where $y_{nt} = 1$ if the im-

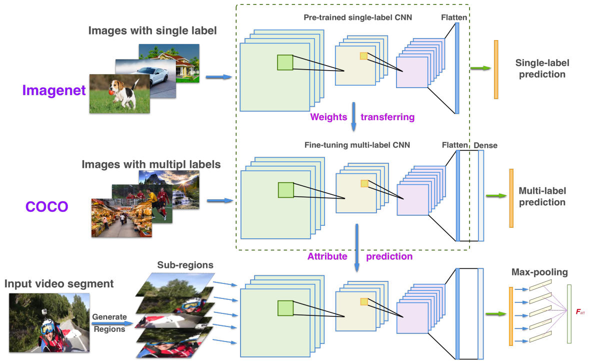

Fig. 2. Our attribute prediction model. We build the CNN based on the ResNet [11] pre-trained on ImageNet [12]. We add a new dense layer after the last convolution block and then fine-tune the model for multi-label prediction purpose. Given a test video segment, a small set of proposed sub-regions are passed into the CNN and the output of each sub-region are then aggregated by max-pooling to generate the final attribute prediction result (denoted as F_{att}).

age contains the t^{th} attribute, and $y_{nt} = 0$ otherwise. If we use $\hat{y_n} = [\hat{y_{n1}}, \hat{y_{n2}}, ...\hat{y_{nt}}]$ to represent the predicted attribute probabilities of the n^{th} training image, then the training objective is to minimize:

$$J = -\frac{1}{N}\frac{1}{T}\sum_{n=1}^{N}\sum_{t=1}^{T}[y_{nt}\log\hat{y_{nt}} + (1 - y_{nt})\log(1 - \hat{y_{nt}})] \quad (1)$$

For implementation, we use the 82783 training images from the MS COCO dataset [9] for the training purpose. We obtained 186 unique words and use them as our semantic attributes. We do not consider a large vocabulary since the distribution of discovered semantic attribute is unbalanced. The statistical information and some examples of these attributes could be found in the supplementary material. The original caption of an image is then replaced with a small set of attributes as the ground-truth annotation. During the fine-tuning phase, the parameters of the new fully-connected layer and the output layer are initialized with random values. The learning rates of these two layers are set to 0.001. All parameter values in other layers are fixed throughout the fine-tuning process. We employ the Stochastic Gradient Descent (SGD) as the optimizer and execute 20 epochs in total. The momentum is set to 0.9, and the dropout rate is set to 0.5.

After the fine-tuning process, we predict each video frame's attributes by selecting and feeding image regions into the trained model. Considering the efficiency of the deep network, we firstly employ Multi-scale Combinatorial Grouping (MCG) [13] to extract hundreds of sub-regions from the given image, we then follow a similar approach in [10] and adopt the normalized cut algorithm to cluster all region proposals into c clusters based on the IoU (Intersection-over-Union) affinity matrix. The top k proposals in terms of predictive scores defined in the MCG method are then selected and passed into the trained CNN. We also add the original image to the proposal group, and obtain $ck + 1$ region proposals for each image. In the final stage, we use the simple max-pooling to aggregate the outputs of all proposals into a compact attribute prediction vector F_{att}.

To evaluate the region-based attribute prediction approach, we design an experiment to predict attributes for a set of images using regions and the whole image respectively. Since no ground-truth captions of test images are provided in the MS COCO dataset, we randomly sample 5000 images from the validation set and use them for testing. The evaluation metric is Average Precision (AP) and mean of AP (mAP), complying with the protocols in [10]. The results indicate that our region based prediction method significantly outperforms the whole image based method on more than 93% of all the attributes. The mAP on the 186 attributes is 0.439 for region-based method and 0.116 for whole image based. This further motivates us to apply the region-based method to video content analysis. Please see the supplementary material for more details.

2.2. Building Deep Features

As summarized in Figure 1, our deep features come with two parts. Given a single video frame, we pass it into the trained CNN and take the output of the last convolution block as visual features. The result of attribute prediction is then treated as semantic features. For feature fusion, we notice that the authors in [14] compute two affinity matrices on visual features and semantic features respectively, and introduce an external parameter α (from 0 to 1) to fuse the two affinity matrices. This method is not applicable to our case, since their visual features and semantic features are extracted from two different models that are trained respectively on different image and text corpora. Moreover, the values of α are often set empirically, and tuning the value of α will introduce extra computational cost.

Considering that our visual and semantic features are computed using the same model, we fuse these two types of features by vector concatenation. Besides, we also investigate the impact of applying dimensionality reduction techniques to our deep features. Please see Section 4.1 for the experimental details.

3. GENERATING VIDEO SUMMARY

Given an input video, we divide it into segments in a uniform length of 10 frames. For example, if a video contains 5400 frames, then the number of segments is 540. Each segment is represented by the first frame of this segment. We then generate the video summary by finding an optimal subset of these video segments. To achieve this, our first step is to group all video segments into M continuous groups by evaluating pairwise similarity of the adjacent segments. We measure such similarity by computing the Cosine distance of their corresponding deep features extracted from our deep CNN. We then obtain a symmetric affinity matrix M_{sim} where each entry $M_{sim}(i, j)$ quantifies the similarity between segment i and segment j. Considering the temporal peculiarity of the video content, we only consider the similarity between a segment with its k temporal neighbors, so we set $M_{sim}(i, j) = 0$ if $|i - j| > k$, and $M_{sim}(i, i) = 0$.

We apply the Bundling Center Clustering (BCC) [6] to help cluster video segments into M groups. Based on the temporal constrained affinity matrix, dense neighbors [15] of each video segment from the matrix are generated, and local dense neighbor clusters with high average similarity score are then identified as local bundling centers. We follow [6] to grow and merge local bundling centers and finally obtain M merged large clusters, which we consider as the video segment groups. An obvious advantage of BCC is that the number of cluster M can be automatically obtained by its dynamic programming approach, which significantly reduce the time for parameter tuning.

After the clustering stage, for each group $G_i(i = 1, ..., N)$, we pick a certain length of continuous segments from its central part to avoid introducing noisy or redundant information near the group boundary (see Figure 1). Denote the video summary length as L_{sum}, the picked video length from each group G_i would be:

$$PickedLength(G_i) = \frac{length(G_i)}{\sum_{i=1}^{N} length(G_i)} * L_{sum} \qquad (2)$$

Hence, a longer group will contribute relatively more video segments for the summary geneartion. The final summary is then obtained by concatenating all the selected video segments $S_m(m = 1, ..., M)$ in the temporal order.

4. EXPERIMENT AND DISCUSSION

To evaluate the proposed framework, we use the SumMe [3] dataset containing 25 videos as our testbed. This dataset features various types of user-generated videos, such as static, moving and egocentric, and most of them are either unedited or minimally edited. Each video in the SumMe dataset contains at least 15 manually-created video summaries, we treat all of them as the ground truth to evaluate our method. Two types of evaluation metric, namely, the maximun-based f-measure [16] and the average-based f-measure [3] are used. More specifically, the maximum-based f-measure is computed based on the most similar human generated summary, while the average-based f-measure is computed on all manually created summaries. We use both of them to compare our approach with other state-of-the-art methods.

We use the code provided by [3] to compute the average f-measure and maximum f-measure. For constructing the temporal affinity matrix M_{sim}, we set $M_{sim}(i, j) = 0$ if $|i - j| > 20$ (we have varied this number and obtained similar results). The summary length L is set to be approximately 15% of the input video's length following [3].

4.1. Evaluating Different Feature Construction Methods

To examine the performance of different feature construction approaches, we design the following five methods: 1) visual features only (VF, 2048-d); 2) semantic features only (SF, 186-d); 3) concatenate visual and semantic features (VSF, 2234-d); 4) concatenate visual and semantic features, then apply PCA to the joint features (PCA-VS, 442-d); 5) apply PCA to the visual features, then concatenate visual and semantic features (PCA-V+S, 256-d + 186-d).

We use these five types of deep features to summarize videos in the SumMe dataset and use the average f-measure as the evaluation metric. The results shown that the PCA-V+S approach outperforms all the other approaches in terms of the average f-measure (0.243) computed on all the 25 videos. We also notice that only using semantic features give the worst performance in this experiment (0.176), this is probably caused by the lack of visual cues in the semantic space. We also learn that directly concatenate visual and semantic features are not applicable, due to the dimensionality difference (2048 for visual part and 186 for semantic part). Hence, we adopt the PCA-V+S approach to compute our deep features for the summarization task. The complete result could be found in the supplementary material.

4.2. Maximum-based Evalution

We compare our SASUM with some recent approaches using the maximum f-measure: 1) **Interestingness** video summarization [3] is a supervised method which uses several manually defined objective function to help summarize videos. 2) **Submodular** [16], in which a submodular function is learned to optimize the objective function for selecting video frames. 3) **DPP** [17] is a supervised approach which use Determinant Point Process (DPP) to help generating video summaries. 4) **dppLSTM** [18] is a supervised method which combines both of DPP and LSTM. 5) **Video MMR** [19] is an unsupervised approach which defines redundancy and representativeness to selecting video frames for summary generation.

As shown in Table 1, our approach achieves the highest overall score of 0.521 on the SumMe dataset (the previous state-of-the-art result published very recently was 0.429 [18]). The results demonstrate that the proposed approach is able to create video summaries closer to the human-level performances than other approaches.

Some examples of video summaries generated using our method is shown in Figure 3. The peaks of blue lines mean that the corresponding video segments enjoy high popularity for being selected by human subjects. From the figure we

Table 1. Quantitative results with maximum-based evaluation. We report the mean maximum f-measure computed on all videos in the SumMe dataset.

	Method	Mean Max F-measure
Supervised	Interestingness	0.394
	Submodular	0.397
	DPP	0.413
	dppLSTM	0.429
Unsupervised	Videov MMR	0.266
	SASUM (Ours)	**0.521**

could observe that the segments selected by our method (orange blocks) show strong correlation to the blue lines. This demonstrate that our approach is consistent with human perception of the visual contents.

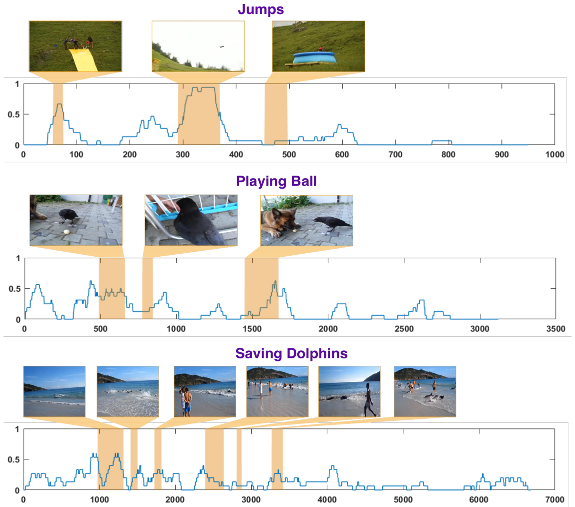

Fig. 3. Some qualitative results of video summarization. The orange blocks represent the video segments selected by our approach. For each segment, we show an image shot at its central part. The blue lines denotes the ratio of human annotators who agree to include each frame in their manually-created summaries.

4.3. Average-based Evaluation

We compare our SASUM with several recent approaches using the average-based evaluation: 1) **Uniform sampling** is a popular baseline method. 2) **Interestingness** [3] manually defines and optimizes some objectives based on the SumMe dataset. 3) **Attention** model focuses on the visual attention [20] to generate video summary. 4) **Title-based** [5] is a semi-supervised method which leverages a set of visual concepts discovered using video titles in the SumMe dataset. 5) **Semantic** approach is an unsupervised technique that maps the visual content to the semantic space [7] for summary generation. 6) **WebPrior** [4] uses web-image based prior information to help generate video summaries. 7) **Quasi** [21] learns a dictionary from the given video using group sparse coding for summary generation.

We only report the mean average f-measure scores of the **WebPrior** [4] and **Quasi** [21] since the f-measure on each video are not available: [4] achieves 0.24 while [21] achieves 0.246. The remaining results of the average-based evaluation is shown in Figure 4. It could be noticed that our SASUM outperforms all the other video summarization approaches except the title-based. Note that this approach leverages the video titles in the SumMe dataset to find some groups of web images, and then a mapping function from images to videos is learned to generate visual concepts for the summarization task. Adding such domain specific knowledge would be helpful for video summarization. Our video summaries are generated using a relatively simple framework without prior knowledge from these videos. Nevertheless, our approach outperformed the title-based method for some videos. We believe the performance of our approach could be enhanced by defining domain-specific semantic attributes and by adopting more sophisticated video segmentation techniques like [3].

5. CONCLUDING REMARKS

In this work, we present the investigation into the value of automatically mining high-level semantic information for the video summarization problem. In the process, we design an algorithm to learn a set of semantic attributes that are automatically discovered from a joint image and text corpora. We then predict attributes on user videos and use the predicted output as an essential part of our deep features. We employ the bundling center clustering method to help generate the final video summary. By comparing our result with several recent computational approaches, we show the advantage of our joint deep features for the video summarization problem.

6. REFERENCES

[1] Engin Mendi, Hélio B Clemente, and Coskun Bayrak, “Sports video summarization based on motion analysis,” *Computers & Electrical Engineering*, vol. 39, no. 3, pp. 790–796, 2013.

[2] Marcus J Pickering, Lawrence Wong, and Stefan M Rüger, “Anses: Summarisation of news video,” in *International Conference on Image and Video Retrieval*. Springer, 2003, pp. 425–434.

[3] Michael Gygli, Helmut Grabner, Hayko Riemenschneider, and Luc Van Gool, “Creating summaries from user videos,” in *European conference on computer vision*. Springer, 2014, pp. 505–520.

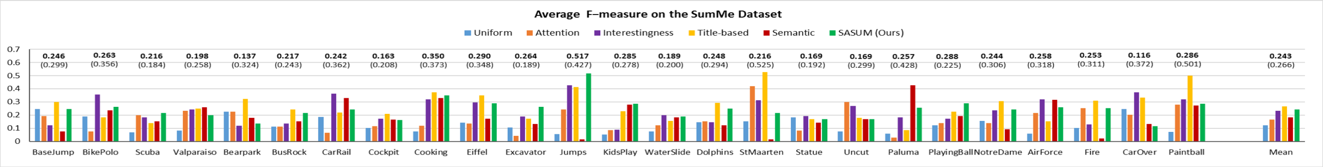

Fig. 4. Quantitative results with the average-based evaluation. Bold-faced numbers represent our results; numbers in parentheses represent the highest score from any approach. The last column shows mean scores computed on all videos.

[4] Aditya Khosla, Raffay Hamid, Chih-Jen Lin, and Neel Sundaresan, "Large-scale video summarization using web-image priors," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2698–2705.

[5] Yale Song, Jordi Vallmitjana, Amanda Stent, and Alejandro Jaimes, "Tvsum: Summarizing web videos using titles," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5179–5187.

[6] Qian Zhang and Guoping Qiu, "Bundling centre for landmark image discovery," *International Journal of Multimedia Information Retrieval*, vol. 5, no. 1, pp. 35–50, 2016.

[7] Mayu Otani, Yuta Nakashima, Esa Rahtu, Janne Heikkilä, and Naokazu Yokoya, "Video summarization using deep semantic features," *arXiv preprint arXiv:1609.08758*, 2016.

[8] Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky, "The stanford corenlp natural language processing toolkit.," in *ACL (System Demonstrations)*, 2014, pp. 55–60.

[9] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick, "Microsoft coco: Common objects in context," in *European Conference on Computer Vision*. Springer, 2014, pp. 740–755.

[10] Yunchao Wei, Wei Xia, Junshi Huang, Bingbing Ni, Jian Dong, Yao Zhao, and Shuicheng Yan, "Cnn: Single-label to multi-label," *arXiv preprint arXiv:1406.5726*, 2014.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," *arXiv preprint arXiv:1512.03385*, 2015.

[12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 248–255.

[13] Jordi Pont-Tuset, Pablo Arbelaez, Jonathan Barron, Ferran Marques, and Jitendra Malik, "Multiscale combinatorial grouping for image segmentation and object proposal generation," 2015.

[14] Zhuo Lei, Ke Sun, Qian Zhang, and Guoping Qiu, "User video summarization based on joint visual and semantic affinity graph," in *Proceedings of the 2016 ACM workshop on Vision and Language Integration Meets Multimedia Fusion*. ACM, 2016, pp. 45–52.

[15] Hairong Liu, Longin J Latecki, and Shuicheng Yan, "Robust clustering as ensembles of affinity relations," in *Advances in neural information processing systems*, 2010, pp. 1414–1422.

[16] Michael Gygli, Helmut Grabner, and Luc Van Gool, "Video summarization by learning submodular mixtures of objectives," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3090–3098.

[17] Ke Zhang, Wei-Lun Chao, Fei Sha, and Kristen Grauman, "Summary transfer: Exemplar-based subset selection for video summarization," *arXiv preprint arXiv:1603.03369*, 2016.

[18] Ke Zhang, Wei-Lun Chao, Fei Sha, and Kristen Grauman, "Video summarization with long short-term memory," *arXiv preprint arXiv:1605.08110*, 2016.

[19] Yingbo Li and Bernard Merialdo, "Multi-video summarization based on video-mmr," in *11th International Workshop on Image Analysis for Multimedia Interactive Services WIAMIS 10*. IEEE, 2010, pp. 1–4.

[20] Naveed Ejaz, Irfan Mehmood, and Sung Wook Baik, "Efficient visual attention based framework for extracting key frames from videos," *Signal Processing: Image Communication*, vol. 28, no. 1, pp. 34–44, 2013.

[21] Bin Zhao and Eric P Xing, "Quasi real-time summarization for consumer videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2513–2520.