Video Summarization
Mid-Term Report

Archit Sharma (14129), **Kanishk Gandhi** (14235),
Nikhil Vanjani (14429), **Animesh Ramesh** (14511), **Shibhansh Dohare** (14644)

1 Task Description

Video summarization aims at producing a summary video, typically 5%-15% of the whole video, consisting of the most informative frames/sequences from the original video. A good video summary depicts the synopsis of the original video, in a compact way depicting all important and relevant scenes/shots.

2 Literature Survey

Selecting the important shots/ frames of a video is done based on algorithms of mainly two types : *Unsupervised algorithms* that use manually defined factors for comparing frames and then subsequently choosing the key ones. *Supervised algorithms* require training examples to train the model to learn which parts of a video are important. This is acheived by using human edited summaries of videos. Recent studies in supervised learning have given some promising results.

2.1 Unsupervised Algorithms

2.1.1 Shot Boundary Detection And Key Frame Extraction

Key frame extraction is one of the primary tasks in summarizing a video. Shot boundary detection is one of the primary steps of segmenting the video temporally. Deleting redundant information is achieved by segmenting the video into shots.

2.1.2 VSUMM

Video Summarization (VSUMM), as described in [Avila et al.2008], is one of the initial algorithms to generate summaries. It is based on clustering video frames based on visual features, particularly, *Color Histograms*. The summaries are evaluated by a set of users. The paper also explores the effect of pre-sampling videos. The algorithm is implemented and tested on SumMe benchmark, as explained below.

2.1.3 Still and moving (STIMO)

STIMO uses the HSV color distribution. This is a fast clustering algorithm and can be used to generate highly custumizable video summaries based on the user's requirements. STIMO has been shown to be efficient in giving video summaries quickly making on-the-fly sage possible. Both still and moving storyboards are possible using STIMO.

2.1.4 VSCAN

VSCAN witis an enhanced Video Summarisation algorithm using Density based Spatial Clustering as proposed by [Mohamed et al. 2014]. It is based on a modified DBSCAN clustering algorithm which uses color and texture features for summarising video content. Color features are extracted using color histogram in HSV color space and texture features are extracted using a two-dimensional Haar wavelet transform in HSV color space. An advantage of such an approach is that one doesn't need to determine the number of clusters beforehand. Also, the algorithm is able to detect noise in frames automatically. The study shows that that VSCAN generates better quality video summaries as compared to other approaches like- Delaunay Triangulation, STIMO, VSUMM.

2.2 Supervised

2.2.1 Video Summarization with Long Short-term Memory

LSTMs (Long Short Term Memory) are a type of recurrent neural networks that perform well for tracking and maintaining dependencies over big temporal ranges, intuitively making them ideal for video summarization. LSTMs have previously been used for video captioning too. The model used in this algorithm as described in [Zhang *et al.* 2016] utilizes two LSTM layers, one in the forward direction and the other in the reverse direction. The two of these layers are essentially not connected. The outputs of these layers are fed to a multilayer perceptron (MLP). This model is called the vsLSTM model. The algorithm can use either importance scores for each frame or selected keyframes encoded as binary indicator vectors. The features used for training the model can be deep or shallow features. We use a combination of deep features from a CNN and shallow

features like color histograms, GIST, HOG, dense SIFT etc. While learning, we first train the MLP and subsequently the LSTM layers and then the two of them together. An extension of the algorithm called Determinantal point processes (DPP) LSTM further improves the performance of the model.

3 Dataset

The dataset chosen for video summarization is SumMe [Gygli et al. 2014]. There are a lot of datasets available for video summarization, particularly notable being the VSUMM dataset and TVSum50 dataset compiled by Yahoo! Labs. Video summarization is a particularly hard task to evaluate, because there can be no unilateral consensus on which summary represents the video best. The general approach in most of the datasets has been to get users, across the complete spectrum of ages and genders, to rate particular summaries, to create an ordering of the summaries. This particular approach does not allow for generalization to any other summary, except those evaluated for the dataset, restricting the utility of the datasets. SumMe dataset, however:

- Contains summaries generated by users themselves. Each video is summarized by multiple users, as there is no well-defined "best" summary. Automatic summaries are compared to the human summaries, allowing a common benchmark for comparison.
- allows for automatic evaluation of the summaries generated by the different algorithms.

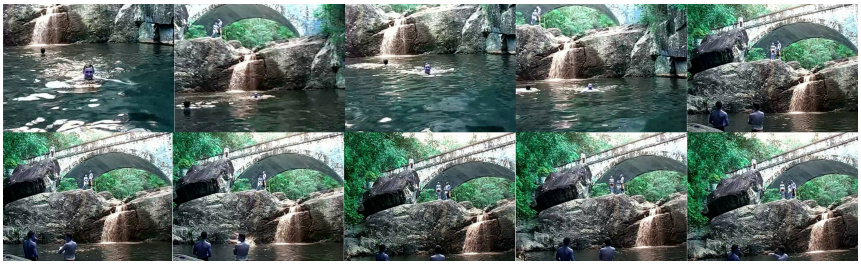
The evaluation, compared to the human generated summaries is given by F-measure, as described in [Gygli *et al.* 2014]

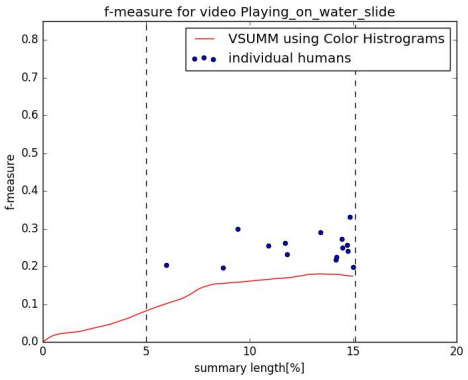
4 Experimental Results

4.1 Evaluation of VSUMM on SumMe Benchmark

VSUMM algorithm was evaluated over the SumMe dataset. The results for clustering algorithm have been pre-tabulated in [Gygli et al.2014]. The algorithm employs the K-means algorithm, which clusters the frames according their color histograms. The color histrogram consisted of pixel values in 16 bins. Ideally, the histogram should be $16 \times 16 \times 16$ tensor. However, due to the computational overhead of computing such a tensor, it was decided that the histograms for the 3 channels would be computed separately resulting in 48×1 feature vector for each frame, when flattened. The results were comparable. The VSUMM algorithm gives comparable performance to the clustering algorithm with $16 \times 16 \times 16$ histograms. For example, the summary produced by this implementation for *Playing on water slide* gets a f-measure score of 0.174, greater than the f-measure score of 0.141 for clustering algorithm in [Gygli et al.2014]. However, in general the performance is worse compared to human summaries and other algorithms developed in recent times. However, this does set the benchmark for automatic summarization. An example summary of 10 frames is shown for the video *Paluma Jump*. A plot comparing VSUMM generated summary to human summaries for *Playing on water slide* is also shown.

The effect of pre-sampling was also observed. One frame out of every 2, 5, 10, 25, 30, 50, 75, 100 frames were chosen in separate steps. The pre-sampling had no or minimal effect on the f-measures of the summaries till frames are chosen every 25 frames. However, the f-measure scores dropped drastically after that. This partly happened because sampling at such high rates did not allow for large enough summaries to be generated, which led to a drastic drop in f-measures of the summary.





5 Future Plans

We plan to do a more detailed analysis of VSUMM algorithm on the SumMe benchmark, evaluating the performance of the algorithms based on their categories. We also plan to implement other algorithms, particularly those based on learning "summarization" using a combination of shallow (histogram, SIFT etc.) and deep features in the supervised setup, such as Video Summarization using LSTMs. Also, we plan to devise an algorithm, possibly an ensemble or a variant of the algorithms we have reviewed, and evaluate its performance on the SumMe benchmark.

References

[Avila et al. 2008] Sandra E. F. de Avila, Antonio da Luz Jr. Arnaldo de A. Ara ujo, Matthieu Cord 2008. *VSUMM: An Approach for Automatic Video Summarization and Quantitative Evaluation*,

[Gygli et al. 2014] Michael Gygli, Helmut Grabner, Hayko Riemenschneider, and Luc Van 2014. *Creating Summaries from User Videos*,

[Zhao et al. 2000] Li Zhao, Wei Qi, Stan Z. Li, Shi-Qiang Yang, H. J. Zhang 2000. *Key-frame Extraction and Shot Retrieval Using Nearest Feature Line (NFL)*,

[Song et al. 2015] Yale Song, Jordi Vallmitjana, Amanda Stent, Alejandro Jaimes Yahoo Labs, New York 2015. *TVSum: Summarizing Web Videos Using Titles*,

[Zhang et al. 2016] Ke Zhang, Wei-Lun, Chao Fei Sha and Kristen Grauman 2016. *Video Summarization with Long Short-term Memory*,

[Sebastian et al. 2015] Tinumol Sebastian, Jiby J. Puthiyidam 2015. *A Survey on Video Summarization Techniques*,

[Khattabi et al. 2015] Zaynab El khattabi, Youness Tabii, Abdelhamid Benkaddour 2015. *Video Summarization: Techniques and Applications*,

[Mehmood et al. 2014] Irfan Mehmood, Muhammad Sajjad, Sung Wook Baik 2014. *Visual attention based extraction of semantic keyframes*,

[Gong et al. 2014] Boqing Gong, Wei-Lun Chao, Kristen Grauman and Fei Sha 2014. *Diverse Sequential Subset Selection for Supervised Video Summarization*,

[Mohamed et al. 2014] Karim M. Mohamed, Mohamed A. Ismail, and Nagia M. Ghanem 2014. *VSCAN: An Enhanced Video Summarization using Density-based Spatial Clustering*,