TEMPORAL VIDEO SEGMENTATION

MEHMET MURAT EROL

DECEMBER 2019

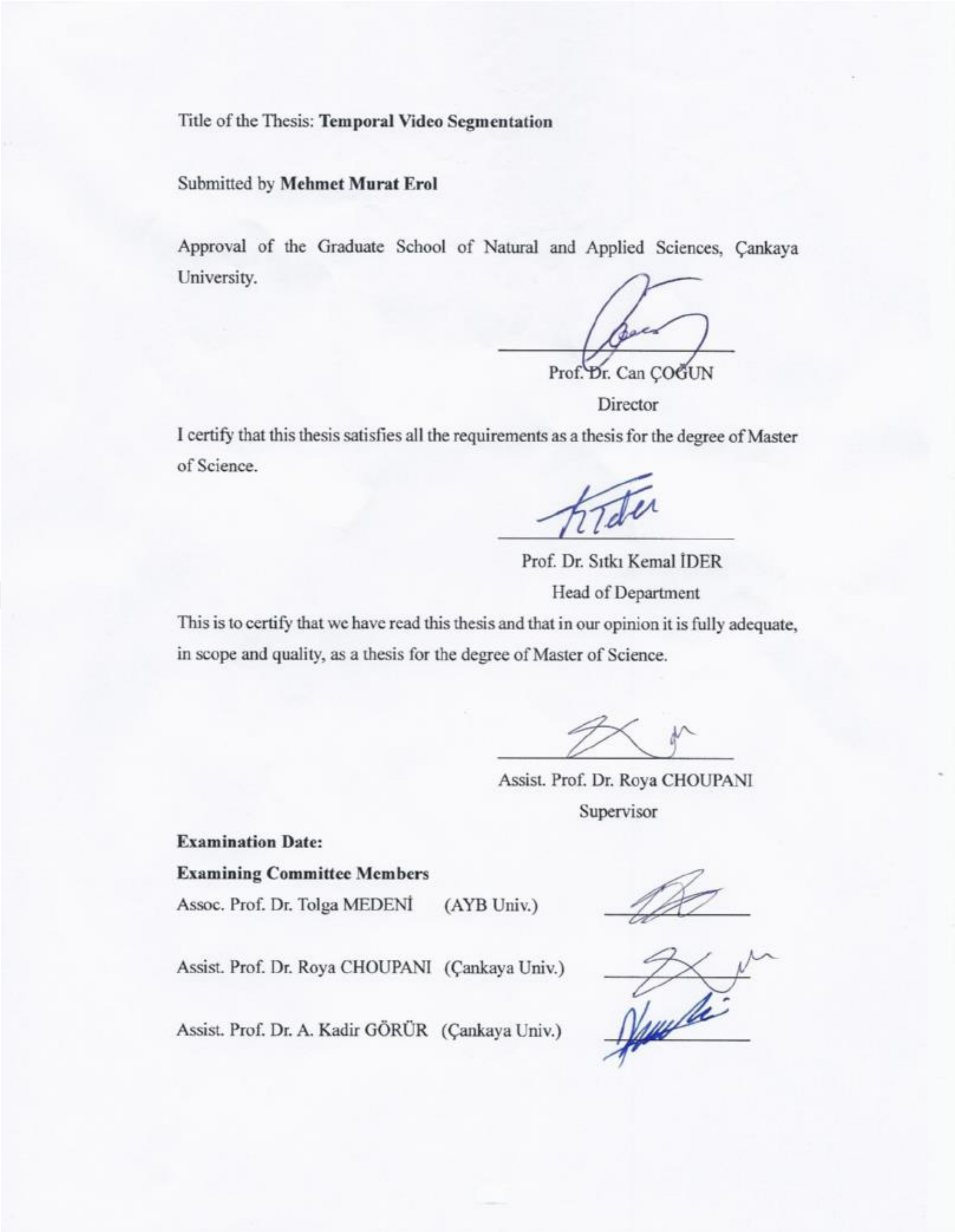
TEMPORAL VIDEO SEGMENTATION

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED
SCIENCES OF
ÇANKAYA UNIVERSITY

BY
MEHMET MURAT EROL

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE
DEGREE OF
MASTER OF SCIENCE
IN
COMPUTER ENGINEERING
DEPARTMENT

DECEMBER 2019



Title of the Thesis: **Temporal Video Segmentation**

Submitted by **Mehmet Murat Erol**

Approval of the Graduate School of Natural and Applied Sciences, Çankaya University.

Prof. Dr. Can ÇOĞUN
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Prof. Dr. Sıtkı Kemal İDER
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. Roya CHOUPANI
Supervisor

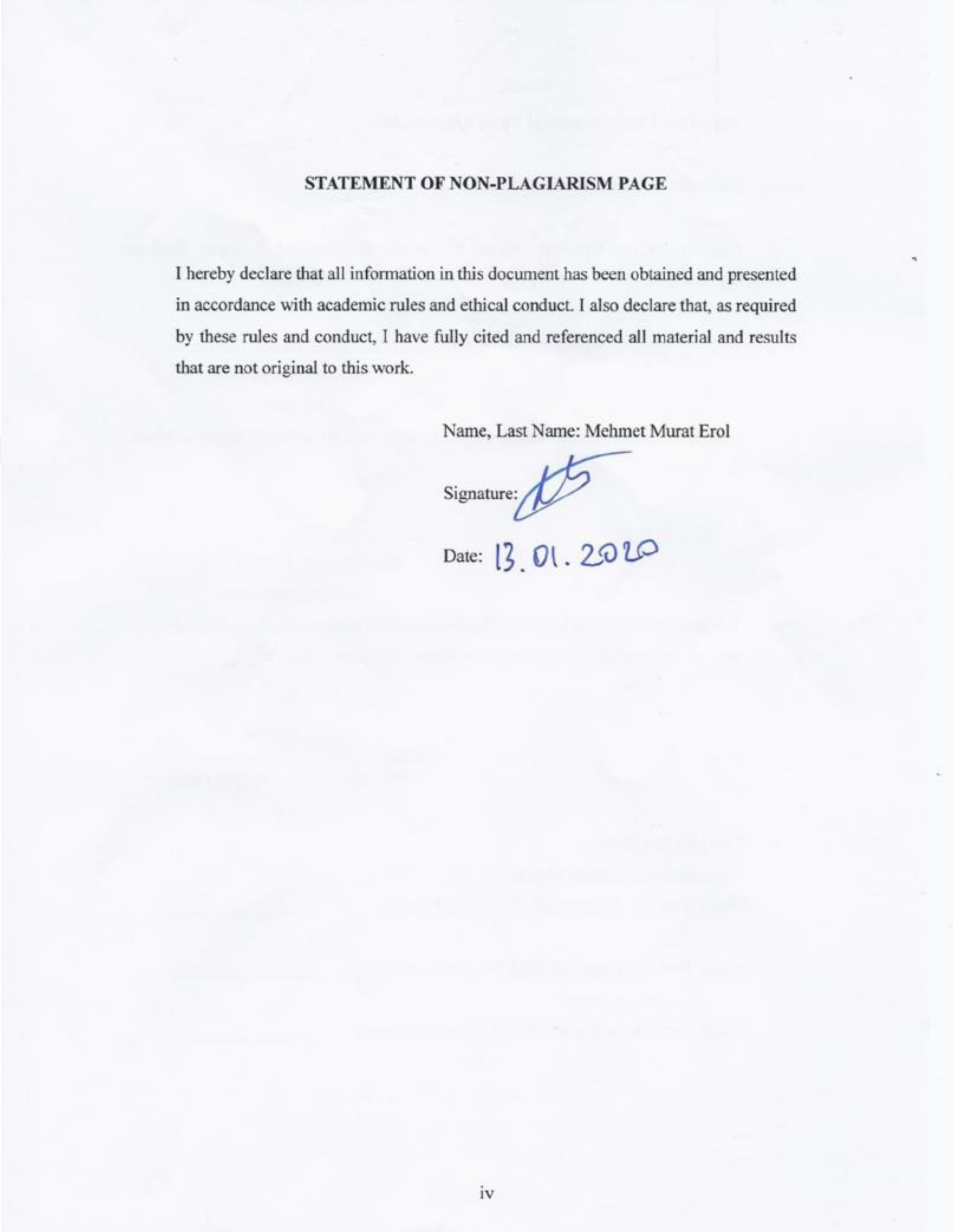
Examination Date:

Examining Committee Members

Assoc. Prof. Dr. Tolga MEDENİ (AYB Univ.)

Assist. Prof. Dr. Roya CHOUPANI (Çankaya Univ.)

Assist. Prof. Dr. A. Kadir GÖRÜR (Çankaya Univ.)



STATEMENT OF NON-PLAGIARISM PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: Mehmet Murat Erol

Signature:

Date: 13.01.2020

ABSTRACT

TEMPORAL VIDEO SEGMENTATION

EROL, Mehmet Murat
M.Sc., Department of Computer Engineering
Supervisor: Assist. Prof. Dr. Roya CHOUPANI

DECEMBER 2019, 26 pages

With the advancement in technology video content generation increased rapidly. This advancement of technology also increased the consumption of video information. The increase in both generation and consumption of video content has created the need of segmenting, summarizing and indexing video with high efficiency.

Video segmentation is the first step to summarize and index videos. Video segmentation aims to segment a video into meaningful, consistent shots. After segmenting video into shots with consistent content then we can apply semantic segmentation techniques to further analysis of a video.

In this thesis Temporal Video Segmentation is examined in both compressed and uncompressed domain and presented a new method using artificial neural networks that has improved performance over methods presented in the related work.

Keywords: Video, Segmentation, Compressed Video, Uncompressed Video

ÖZ

ZAMANSAL VİDEO BÖLÜMLEME

EROL, Mehmet Murat
Yüksek Lisans, Bilgisayar Mühendisliği Anabilim Dalı
Tez Yöneticisi: Dr. Öğr. Üyesi Roya CHOUPANI

ARALIK 2019, 26 sayfa

Teknolojinin ilerlemesi ile birlikte video içerik üretimi de hızlı bir şekilde artmıştır. Teknolojideki bu ilerleme aynı zamanda görsel bilgi kullanımını da arttırdı. Video içeriğinin üretimindeki ve tüketimindeki bu artışlar, videoların etkili bir şekilde bölümlenmesi, özetlenmesi ve sıralanması ihtiyacını doğurmuştur.

Video bölümle, video özetleme ve sıralama işlemlerinin ilk adımıdır. Video bölümleme bir video anlamlı ve kendi içinde bütüncül parçalara ayırmayı hedefler. Videoyu kendi içinde bütüncül parçalara böldükten sonra videonun daha ileri analizi için anlamsal bölümleme tekniklerini uygulayabiliriz.

Bu tezde, Zamansal Video Bölümleme sıkıştırılmış ve sıkıştırılmamış alanlarda incelenmiştir ve yapay sinir ağları kullanılarak ilgili çalışmada geçen metotlardan daha iyi performans gösteren bir metot sunulmuştur.

Anahtar Kelimeler : Video, Bölümleme, Sıkıştırılmış Video, Sıkıştırılmamış Video

To My Parents

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my advisor Asist. Prof. Dr. Roya Choupani for her guidance and support.

I would also like to thank Çankaya University Computer Engineering Department for the opportunities provided.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

TVS	Temporal Video Segmentation
TSoUV	Temporal Segmentation of Uncompressed Video
TSoCV	Temporal Segmentation of Compressed Video
CNN	Central Neural Networks
DCT	Discrete Domaine Transform
DC	Discrete Cosine
MB	Macro Block
MV	Motion Vectors
CM	Coding Mode
CNN	Convolutional Neural Network
HLFPN	High-Level Fuzzy Petri Net
DWT	Discrete Wavelet Transform
SURF	Speeded Up Robust Feature
DBF	Deceived Bilateral Filter

CHAPTER 1

INTRODUCTION

1.1. Overview

With the advancement in technology video content generation increased rapidly [1]. This advancement of technology also increased the consumption of video information [2]. The increase in both generation and consumption of video content has created the need of segmenting, summarizing and indexing video with high efficiency.

Video segmentation is the first step to summarize and index videos. Video segmentation aims to segment a video into meaningful, consistent shots. After segmenting video into shots with consistent content then we can apply semantic segmentation techniques to further analysis of a video.

1.2. Problem Statement

Temporal Video Segmentation (TVS) is highly contested in academia, but because of the high variety of elements of a video there is still not a definitive method to segment a video into meaningful segments.

Moreover, most of the related work present in the literature is bound to the datasets they are using while there are different datasets present today. Because of the nature of boundaries of related work most of them are not transferrable to the commercial products.

1.3 Significance of the Study

Significance of the study is to generate results of presented methods in this thesis using multiple datasets to implement a method that is generic in a sense the method produces results that is higher than the average for every dataset and also highly effective.

1.4 Limitations of the Study

This thesis aims only to create a method that is generic and also highly effective.

1.5 Organization of the Thesis

The thesis is presented in 4 chapters.

In Chapter 1, we introduce our thesis by giving an overview of the thesis. Then we present Problem Statement, Significance of the Study and limitations.

In Chapter 2, we present TVS in detail and related work. This section provides a better understanding about the variety of the elements inside a video and how they are used by the related work.

In Chapter 3, we present our method for TVS and also provide detailed information of the elements of a video.

In Chapter 4, we present our tests and results. The datasets are used also described in detail in this chapter.

CHAPTER 2

RELATED WORKS

2.2 What is Temporal Video Segmentation

Temporal Video Segmentation (TVS) is the first step towards semantic analysis of a video which is required to label, index and retrieve video data. TVS is done by detecting shot boundaries. A shot consists of frames recorded by a camera without any interruption [3].

Shot transitions can be instant or eventual. Instant shots do not contain extra frames between two shots while eventual transitions contain multiple frames between two shots in such cases like fade away. Fade away is a method to make transition between shots while first shot becomes less and less visible while the following shot becomes more and more visible over time with each frame [4].

Below Figure 1. and Figure 2. distinguishes the differences between two different transitions between shots.

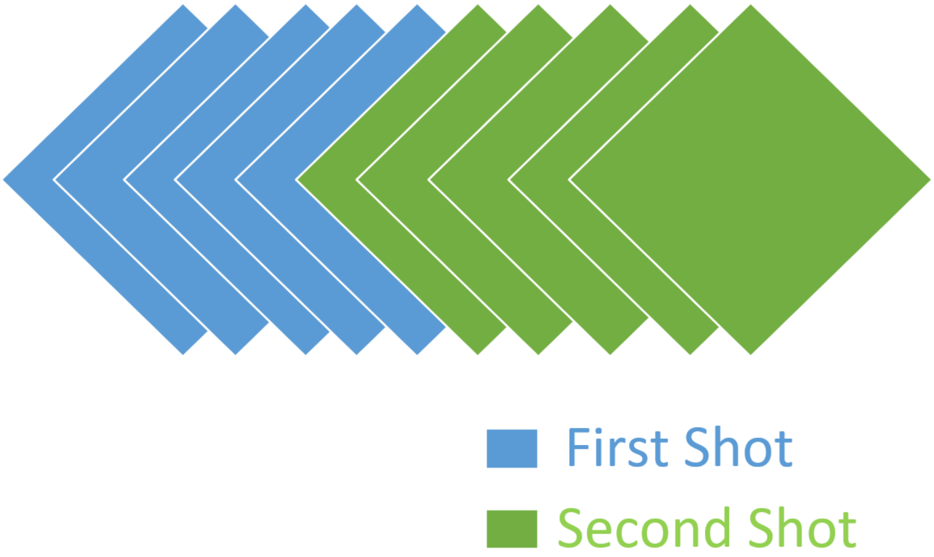


Figure 1 Instant Transition Between 2 Shots

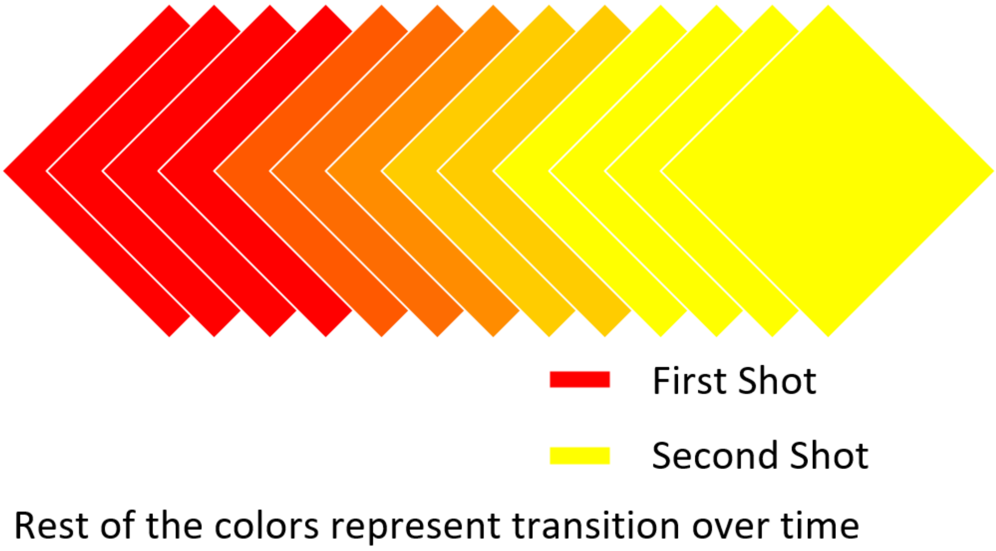


Figure 2 Eventual Transition Between 2 Shots

TVS can be further explained in two different domains. These are TVS of Uncompressed Video and TVS of Compressed Video.

2.2.1 Temporal Segmentation of Uncompressed Video

Temporal Segmentation of Uncompressed Video (TSoUV) takes advantage of pixels, blocks, histograms, clustering techniques, features and mathematical models. Figure 3 shows the basic elements of TSoUV methods.

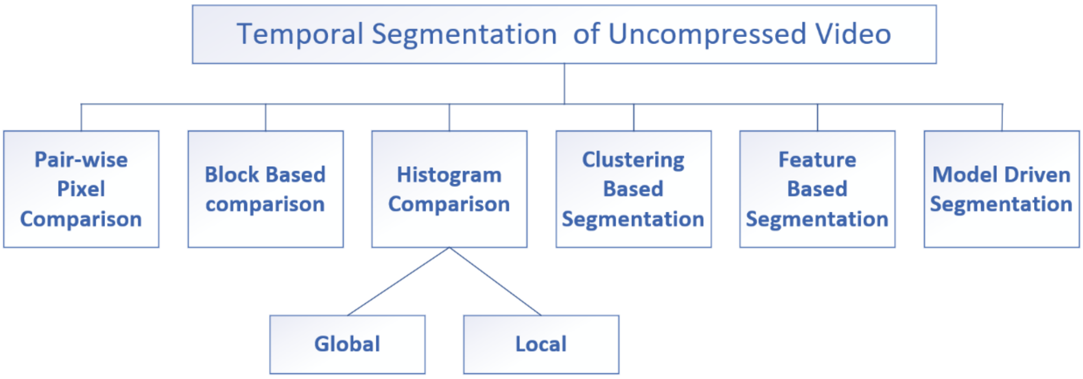

Temporal Segmentation of Uncompressed Video
Pair-wise Pixel Comparison
Block Based comparison
Histogram Comparison
Clustering Based Segmentation
Feature Based Segmentation
Model Driven Segmentation
Global
Local

Figure 3 Temporal Video Segmentation of Uncompressed Video

An uncompressed video consists of raw frames with only pixel information within each frame. Pixels can be represented in several different color formats.

In context of video, a block means m*n fixed sized group of nonoverlapped pixels within a single frame.

Histograms are obtained either locally or globally. Local histogram comparison is made by creating a histogram within a window of fixed sized frames instead of creating a histogram of the entire video in such case it is called a global histogram.

Clustering based segmentation approaches to the problem from a viewpoint that is TVS is a clustering problem and implements clustering methods to segment a video into meaningful shots.

Feature based segmentation ignores the fact that the most atomic element of a video is a pixel and takes the problem into consideration from a wider viewpoint. To find features of a frame to compare with others Central Neural Networks (CNN) are widely used.

Model Driven Segmentation is the widest viewpoint of all the other methods. In this category mathematical models are used to detect shot boundaries.

The state of the art methods usually consists of many different steps that take into consideration more than one segmentation technique. The pixels are usually represented with different color formats, different block sizes are used, while a global or local histogram can be chosen, the bin sizes differentiates between methods, there are wide variety of clustering methods, feature extraction techniques and mathematical models.

The variety of the options available presented in the previous paragraph states that comparing the methods themselves without diving into the details of the steps and choices to come to a conclusion is trivial.

2.2.1.1 Pair-wise pixel comparison

Pair-wise pixel comparison is accomplished by comparing every pixel of N x M sized frame with the corresponding pixel of the following N x M sized frame. The difference between every pixel in a frame and the corresponding pixel is summed up and after summation a predefined or adaptive threshold is applied.

This process is dependent on the resolution of a video and if the context changes above predefined threshold within a single shot between consecutive frames it is prone to false positives. Because a shot might consist a dynamic context in which consecutive frames can have higher difference in pixel wise comparison in comparison within other shots in a video.

A dynamic context consists of frames within a shot when there is high camera movement or high object movement or both. Camera movements can be in three axes naming Pan, Tilt and Zoom and if the movement of a camera is more than the other shots this can cause false positives.

A false positive in this context is if an algorithm finds a shot transition when there is not a shot transition instead a high object movement or high camera movement or both.

2.2.1.2 Block based comparison

Block based comparison is accomplished by comparing every block of an N x M sized frame with the corresponding block of the following N x M sized frame.

To compare the blocks, we first calculate the summation or average of a block's pixel's color values and then compare the result with the following frames block's pixel's color values.

The calculations of a block's pixel values are done with either summation or average of pixel's color values within a block.

2.2.1.3 Global histogram comparison

Global histogram comparison is done by creating histograms of consecutive frames by the color space. There are different color spaces such as HSV, RGB, YCbCr. The complexity of the histogram comparison is based on the color space which dictates the number of bins in a histogram. Though size of the bins can be lowered with the initiative of the implementor to gain speed over correctness of the result.

After creating histograms, the differences of each bins are calculated and summed. If the result of the summation is higher than a threshold T, then it is said that a shot boundary is detected. This is the case where T is static and decided before execution. In such cases that will be discussed in Chapter 2.3 the T is calculated by the metrics obtained from the entire video. In this case it is called a dynamic threshold which is more robust against wrong positives.

2.2.1.4 Local histogram comparison

Local histogram comparison is done by comparing histograms of relative blocks of two consecutive frames. It follows the same principals as Global Histogram Comparison, except local histogram comparison is more robust against two consecutive frames having similar histograms but with different contexts. Local histogram comparison is achieved by comparing non overlapping blocks of the relative block of the consecutive frame.

Histogram comparison is less prone to error because they ignore the change in object movement and camera movement. Instead these algorithms focus on the similarity between entirety of the consecutive frames.

2.2.2 Temporal Segmentation of Compressed Video

Temporal Segmentation of Compressed Video (TSoCV) takes advantage of Discrete Cosine Transform (DCT) coefficients, Discrete Cosine (DC) terms, Macro Block (MB), Coding Mode (CM), Motion Vectors (VT), bitrate.

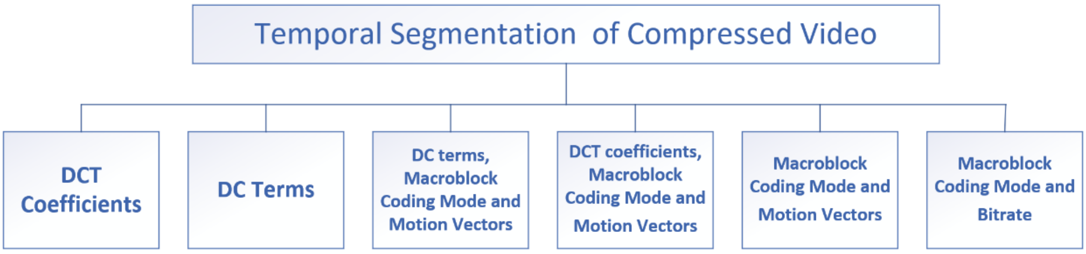

Temporal Segmentation of Compressed Video
DCT Coefficients
DC Terms
DC terms, Macroblock Coding Mode and Motion Vectors
DCT coefficients, Macroblock Coding Mode and Motion Vectors
Macroblock Coding Mode and Motion Vectors
Macroblock Coding Mode and Bitrate

Figure 4 Temporal Video Segmentation of Compressed Video

DCT coefficients are the equivalent of pixels in uncompressed video. That is the most atomic element of a compressed video. They are obtained by applying DCT to raw pixel values of a frame.

In compressed videos a MB usually has size of 8x8 bytes but can also have different sizes commonly 4x4 or 16x16 bytes.

DC terms are the first DCT coefficient in a MB and they have 8 times higher intensity values in that MB. [5]

CM can be either backward, forward, bidirectional or intra.

Backward coding mode indicates that the frame is constructed with the information exists in a future frame while forward coding mode indicates that the frame is constructed with the information that exists in previous frame. If it is (size wise) cheaper to code the frame by using both a previous and a future frame it is called bidirectional coding mode. If none of the previous coding modes provide efficiency in compressing the video the MB is coded using intra coding mode.

MV is used as a complimentary information provided by compressed videos to increase robustness against camera movements namely pan tilt and zoom. Motion vectors are the information about the change of location of macroblocks between two frames that provides compression if one of the next or previous frames contain similar blocks.

2.3 Previously Related Works

Liang et al. [6] proposes a method that lets a pretrained Convolutional Neural Network (CNN) AlexNet [7] to extract features instead of using features presented in the literature.

CNN is one of the most suitable artificial neural network architectures because of its robustness against rotation and transition variance which occurs in a video because of camera movements named Pan, Tilt, Zoom and object movements.

In their proposed method to compare frames they use cosine similarity after getting results from pretrained AlexNet. If the cosine similarity is low that means the two frames high difference and if the cosine similarity is high it means the frames have less difference.

Cosine similarity is especially a better similarity measurement option because CNNs produces vectors and Cosine similarity measurement is based on the cosine angle between two vectors.

After extracting features, they apply three different levels of thresholds for the two different cases namely sudden transition and eventual transition between shot. The first threshold is used to detect sudden transitions. The other two thresholds are used for the start and end of the eventual transition.

To obtain results Liang et al. uses three random videos from internet. These videos' genres are sport, movie and cartoon.

Shen et al. [8] proposes a method using global histogram comparison.

Instead of comparing two consecutive frames Shen et al. also compare the current frame with the future 2nd and 4th frames separately. After the histogram comparison they merge the results using High-Level Fuzzy Petri Net (HLFPN).

Shen et al. uses news stream presented in [9].

In their paper Majumdar et al. [10] provides a different view of point in their paper. They apply global histogram based comparison and pixel wise comparison in different transform domains instead of raw color space.

Majumdar et al. compares 5 different transform domains to compare their performance in histogram based and pixel wise comparison. These are Weibull, Exponential, Normal, Laplacian and Cauchy transform domains.

Majumdar et al. uses two random videos of a football and a cricket match to perform tests.

Kar et al. [11] proposes a method that works on the transform domain as well in uncompressed domain. First, they apply Hilbert Transform to obtain complex and simple parts of frames. After the Hilbert Transform Kar et al. applies another transform namely Discrete Wavelet Transform (DWT) on the complex part of the output of Hilbert Transform. After two different transformations Kar et al. applies threshold to the result.

Kar et al. provides results using TRECVid 2001 [12] dataset.

e Santos et. al [13] proposes a method in uncompressed video domain that first applies Weber Local Descriptor (WLD) to filter the frames and then applies Battacharyya distance to the local histogram of the output from filtered frames. After the comparison they apply a sliding window to determine threshold instead of predefining it to be more robust against eventual transitions.

To provide results e Santos et. al uses two different datasets. These are VIDEOSEG'2004 [14] and TRECVID'2002 [15].

Son et al. [16] proposes a method in their paper that first applies global histogram comparison in hue and saturation instead of RGB color space. After the histogram comparison Son et al. applies static thresholding to the comparison to determine shot boundary detection.

Son et al. uses six videos from a Korean TV show presented in [16] to provide results of their presented method in their paper.

Tippaya et al. [17] proposes a method in their paper that uses both RGB histogram comparison and also Speeded Up Robust Feature (SURF).

Tippaya et al. also uses dynamically calculated threshold after obtaining comparison results from histogram comparison and SURF.

Tippaya et al. performs their tests using videos from Professional Golf Association of Thailand [18].

Kaavya et. al [19] proposes a method in their paper that applies Local Binary Pattern (LBP) on uncompressed video frames, twice. This is done to reduce the feature size of the frames. After LBP local maxima is calculated for each frame to apply threshold.

Kaavya et al. uses various videos from Open Video Project [20] to obtain results of their proposed method.

Chacòn-Quesada et al. [21] presents an improvement in their paper by applying Deceived Bilateral Filter (DBF) to increase the differences between shots.

Chacòn-Quesada et al. uses FIFA's World Cup 2014 videos to obtain their results.

Yang et. al [22] proposes a method in their paper that calculates histogram comparison in HSV color space in a 21 frame window and then apply OTSU's method [23] which is a method to determine threshold.

Yang et al. obtains results of their proposed method by using CC_WEB_VIDEO dataset [24].

Chugh et. al [25] proposes a method in their paper using a block based comparison between frames and the difference is calculated by mean deviation and standard deviation between relative blocks of consecutive frames.

Chugh et al. obtains results by using random videos.

Xu et. al [26] proposes a method in their paper that pre-processes the input video. In this pre-processing step they use adaptive thresholding to the segmented videos into 21 frames that compares the frames withing the 21 frame blocks if they are similar enough to be in the same shot or not. This way they can able to remove unnecessary calculations in the following steps.

After removing unnecessary frames Xu et al. uses an ILSVRC-2012 winner CNN [7] trained with ImageNet to extract features. The output of the CNN is then used to calculate differences between frames to find shot boundaries.

To obtain results of their method Xu et al. uses TRECVID 2001 [12] dataset.

CHAPTER 3

METHODOLOGY

TVS is a highly contested field in academia due to the increasing need to separate a video into meaningful shots. Due to the nature of a video consisting many different elements inside its structure namely, pixels, DCT coefficients, Macro Blocks, Motion Vectors, different color spaces, different coding modes, different transform domains it is prone to produce wrong positives when not done manually. Also due to the variety of different video genres even if the field is highly contested there is still not an off the shelf commercial product for TVS.

To overcome this problem, we propose a method consists of steps shown in Figure 5.

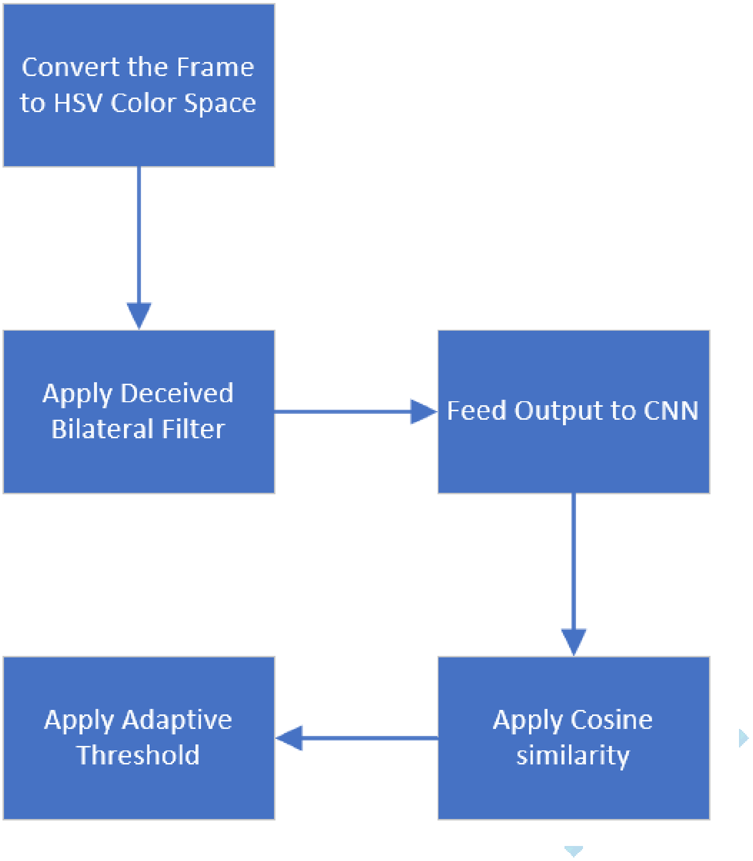

Convert the Frame to HSV Color Space
Apply Deceived Bilateral Filter
Feed Output to CNN
Apply Cosine similarity
Apply Adaptive Threshold

Figure 5 Proposed Method

3.1 Convert the Frame into HSV Color Space

First, we convert the frames into HSV color space to get hue and saturation information while keeping color information.

While traditional RGB color space represents a pixel's color value in Red, Green and Blue components, HSV is more similar to how humans perceive color and represents a pixel's color in Hue, Saturation and Value components.

3.2 Apply Deceived Bilateral Filter

Deceived Bilateral Filter (DBF) presented in [27] decreases the similarity between frames by increasing contrast and sharpening the edges. This is most effective if the resolution or frame rate of a video is lower.

The lower the resolution of a video the more obscure the objects become. This is due to the fact that frames are represented as a dot pixel frames.

Frame rate of a video corresponds to the frames within each second. If the object movement or the camera movement is faster than the frame rate of a video objects again become more obscure because of the fact that the camera movement speed and the object movement speed combined happens move than once between two frames.

We apply DBF to frames so that the color differences are more distinct between frames and the obscurity of objects are decreased.

3.3 Convolutional Neural Network

In our proposed method we use CNN to decrease the feature space and obtain output vectors that will be used in the next step.

CNN is one of the most suitable artificial neural network architectures because of its robustness against rotation and transition variance which occurs in a video because of camera movements named Pan, Tilt, Zoom and object movements.

We use AlexNet [7] as our CNN. It has been widely used and implementation is readily available in MATLAB.

To train our CNN we used Open Video Scene Detection Dataset [28][29].

After training our CNN, output of the DBF filtered HSV frames used with CNN to obtain reduced features.

3.4 Cosine Similarity

Cosine similarity is used to compare the consecutive frames’ features obtained from previously trained CNN.

Cosine similarity is especially a better similarity measurement option because CNNs produces vectors and Cosine similarity measurement is based on the cosine angle between two vectors.

3.5 Adaptive Threshold

To reduce false positives caused by camera movements and object movements, instead of using predefined fixed threshold we used adaptive threshold.

After defining a threshold, we adjusted the threshold by similarities and dissimilarities of 10 previous and 10 feature frames of the current frame during calculations.

If the similarity between two consecutive frames is higher than the threshold it is called a shot boundary.

CHAPTER 4

RESULTS

To test our proposed method, we have used videos from TRECVID 2007 [30] dataset. The details of the videos are shown in Table 1.

Table 1 Dataset

Name of the Video	Number of segments	Duration	Number of Frames
20051231_182800_NBC_NIGHTLYNEWS_ENG (Video 1)	179	2350 seconds	28502
20051208_182800_NBC_NIGHTLYNEWS_ENG (Video 2)	385	4153 seconds	58142
20051213_185800_PHOENIX_GOODMORNCN_CHN (Video 3)	458	4153 seconds	58142
20051227_105800_MSNBC_NEWSLIVE_ENG (Video 4)	293	4153 seconds	58142
20051227_125800_CNN_LIVEFROM_ENG (Video 5)	83	698 seconds	9772
20051209_125800_CNN_LIVEFROM_ENG (Video 6)	97	919 seconds	12864

The result of our proposed method is compared with Liang et al. [6], Lu et al. [31] and Sun et al. [32] with evaluation criteria Recall, Precision and F1 parameters.

Our method and the algorithms of Liang et al. [6], Lu et al. [31] and Sun et al. [32] are implemented to the best of our knowledge in MATLAB to get the results.

Recall is the rate of true positives to the actual positives. Given that a video has T number of segments and Y detected segments, recall is calculated as:

$$Recall = \frac{\mathrm{T}}{\mathrm{Y}}$$

Precision is the rate of true positives to the summation of true and false positives. Given a method produces X true positives and Z false positives, precision is calculated as:

$$Precision = \frac{\mathrm{X}}{\mathrm{X} + \mathrm{Z}}$$

F1 is calculated by taking the weighted average of precision and recall and it is calculated as:

$$F1 = \frac{2 * \mathrm{Precision} * \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}}$$

The results of compared methods are showed in Table 2, Table 3, Table 3, Table 4, Table 5, Table 6 and Table 7.

Table 2 Recall Results of Cut Transitions

	Video 1	**Video 2**	**Video 3**	**Video 4**	**Video 5**	**Video 6**
Sun's	0.872	0.926	0.915	0.875	0.896	0.853
Lu's	0.879	0.883	0.900	0.894	0.865	0.840
Liang's	0.895	0.890	0.868	0.901	0.870	0.900
Ours	0.904	0.916	0.920	0.899	0.886	0.915

Table 3 Precision Results of Cut Transitions

	Video 1	**Video 2**	**Video 3**	**Video 4**	**Video 5**	**Video 6**
Sun's	0.874	0.914	0.891	0.917	0.873	0.870
Lu's	0.864	0.881	0.870	0.903	0.899	0.852
Liang's	0.883	0.907	0.872	0.940	0.863	0.925
Ours	0.925	0.918	0.911	0.947	0.875	0.958

Table 4 F1 Results of Cut Transitions

	Video 1	Video 2	Video 3	Video 4	Video 5	Video 6
Sun [32]	0.873	0.920	0.903	0.896	0.885	0.862
Lu [31]	0.872	0.882	0.885	0.899	0.882	0.846
Liang [6]	0.889	0.899	0.870	0.921	0.867	0.913
Ours	0.915	0.917	0.916	0.923	0.881	0.937

Table 5 Recall Results of Gradual Transitions

	Video 1	Video 2	Video 3	Video 4	Video 5	Video 6
Sun [32]	0.898	0.921	0.890	0.902	0.878	0.918
Lu [31]	0.875	0.919	0.873	0.859	0.850	0.869
Liang [6]	0.882	0.915	0.882	0.872	0.866	0.900
Ours	0.903	0.918	0.894	0.919	0.901	0.905

Table 6 Precision Results of Gradual Transitions

	Video 1	Video 2	Video 3	Video 4	Video 5	Video 6
Sun [32]	0.900	0.906	0.882	0.853	0.903	0.897
Lu [31]	0.873	0.867	0.879	0.856	0.909	0.856
Liang [6]	0.914	0.884	0.934	0.859	0.878	0.879
Ours	0.917	0.904	0.855	0.873	0.900	0.901

Table 7 F1 Results of Gradual Transitions

	Video 1	Video 2	Video 3	Video 4	Video 5	Video 6
Sun [32]	0.899	0.914	0.886	0.878	0.891	0.908
Lu [31]	0.874	0.893	0.876	0.858	0.880	0.863
Liang [6]	0.898	0.900	0.908	0.866	0.872	0.890
Ours	0.910	0.911	0.875	0.896	0.900	0.903

Table 8 Overall Results

	Video 1	**Video 2**	**Video 3**	**Video 4**	**Video 5**	**Video 6**
Sun [32]	0.886	**0.917**	0.895	0.887	0.888	0.885
Lu [31]	0.873	0.888	0.881	0.879	0.881	0.855
Liang [6]	0.894	0.899	0.889	0.894	0.870	0.902
Ours	**0.913**	0.914	**0.896**	**0.910**	**0.891**	**0.920**

After applying our method to the dataset shown in Table-1, our proposed method is able to produce better results overall.

CHAPTER 5

CONCLUSION

With the advancement in technology video content generation increased rapidly. This advancement of technology also increased the consumption of video information. The increase in both generation and consumption of video content has created the of segmenting, summarizing and indexing video with high efficiency.

Video segmentation is the first step to summarize and index videos. Video segmentation aims to segment a video into meaningful, consistent shots. After segmenting video into shots with consistent content then we can apply semantic segmentation techniques to further analysis of a video.

In this thesis we have analyzed Video Segmentation techniques in both compressed and uncompressed domain. Moreover, we have discussed the state of the art methods to segment a video into shots. This research has enabled us to create a new method using CNN to efficiently segment a video.

REFERENCES

[1] Choupani, R., Wong, S., & Tolun, M. (2014). Multiple description coding for SNR scalable video transmission over unreliable networks. *Multimedia Tools and Applications*, *69*(3), 843-858.

[2] Choupani, R., Wong, S., & Tolun, M. (2015, December). Drift-free video coding for privacy protected video scrambling. In *2015 10th International Conference on Information, Communications and Signal Processing (ICICS)* (pp. 1-5). IEEE.

[3] Gunsel, B., Ferman, A. M., & Tekalp, A. M. (1998). Temporal video segmentation using unsupervised clustering and semantic object tracking. *Journal of Electronic Imaging*, *7*(3), 592-605.

[4] Kar, T., & Kanungo, P. (2017, December). Video shot boundary detection based on Hilbert and wavelet transform. In *2017 2nd International Conference on Man and Machine Interfacing (MAMI)* (pp. 1-6). IEEE.

[5] Koprinska, I., & Carrato, S. (2001). Temporal video segmentation: A survey. *Signal processing: Image communication*, *16*(5), 477-500.

[6] Liang, R., Zhu, Q., Wei, H., & Liao, S. (2017, December). A video shot boundary detection approach based on CNN feature. In *2017 IEEE International Symposium on Multimedia (ISM)* (pp. 489-494). IEEE.

[7] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097-1105).

[8] Shen, R. K., Lin, Y. N., Juang, T. T. Y., Shen, V. R., & Lim, S. Y. (2017). Automatic Detection of Video Shot Boundary in Social Media Using a Hybrid Approach of HLFPN and Keypoint Matching. *IEEE Transactions on Computational Social Systems*, *5*(1), 210-219.

[9] Shen, V. R., Tseng, H. Y., & Hsu, C. H. (2014, October). Automatic video shot boundary detection of news stream using a high-level fuzzy Petri net. In *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (pp. 1342-1347). IEEE.

[10] Majumdar, J., Aniketh, M., Abhishek, B. R., & Hegde, N. (2017, April). Video shot detection in transform domain. In *2017 2nd International Conference for Convergence in Technology (I2CT)* (pp. 161-168). IEEE.

[11] Kar, T., & Kanungo, P. (2017, December). Video shot boundary detection based on Hilbert and wavelet transform. In *2017 2nd International Conference on Man and Machine Interfacing (MAMI)* (pp. 1-6). IEEE.

[12] National Institute of Standards and Technology (2018). TRECVid 2001. Retrieved May 2019, from https://trecvid.nist.gov/trecvid.data.html

[13] e Santos, A. C. S., & Pedrini, H. (2017, October). Shot boundary detection for video temporal segmentation based on the weber local descriptor. In *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (pp. 1310-1315). IEEE.

[14] Whitehead, A., Bose, P., & Laganiere, R. (2004, July). Feature based cut detection with automatic threshold selection. In *International Conference on Image and Video Retrieval* (pp. 410-418). Springer, Berlin, Heidelberg.

[15] National Institute of Standards and Technology (2018). TRECVid 2002. Retrieved May 2019, from https://trecvid.nist.gov/trecvid.data.html

[16] Son, J. W., Park, W., Han, M., & Kim, S. J. (2017, February). Scene boundary detection with graph embedding. In *2017 19th International Conference on Advanced Communication Technology (ICACT)* (pp. 451-453). IEEE.

[17] Tippaya, S., Sitjongsataporn, S., Tan, T., Khan, M. M., & Chamnongthai, K. (2017). Multi-modal visual features-based video shot boundary detection. *IEEE Access*, *5,* 12563-12575.

[18] Professional Golf Association of Thailand (2019). Retrieved May 2019, from https://trecvid.nist.gov/trecvid.data.html

[19] Kaavya, S., & Priya, G. L. (2017, February). Local Binary Pattern based Shot Boundary Detection for Video Summarization. In *2017 Second International Conference on Recent Trends and Challenges in Computational Models (ICRTCCM)* (pp. 165-169). IEEE.

[20] Interaction Design Laboratory. The Open Video Project. Retrieved May 2019, from https://open-video.org/

[21] Chacón-Quesada, R., Calderón-Ramírez, S., & Siles, F. (2016, November). Improving the temporal segmentation in digital videos using the deceived bilateral filter. In *2016 IEEE 36th Central American and Panama Convention (CONCAPAN XXXVI)* (pp. 1-6). IEEE.

[22] Yang, Z., Tian, L., & Li, C. (2017, December). A Fast Video Shot Boundary Detection Employing OTSU's Method and Dual Pauta Criterion. In *2017 IEEE International Symposium on Multimedia (ISM)* (pp. 583-586). IEEE.

[23] Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, *9*(1), 62-66.

[24] Wu, X., Hauptmann, A. G., & Ngo, C. W. (2007, September). Practical elimination of near-duplicates from web video search. In *Proceedings of*

the 15th ACM international conference on Multimedia (pp. 218-227). ACM.

[25] Chugh, I., Gupta, R., Kumar, R., & Sahay, P. (2016, January). Techniques for key frame extraction: Shot segmentation and feature trajectory computation. In *2016 6th International Conference-Cloud System and Big Data Engineering (Confluence)* (pp. 463-466). IEEE.

[26] Xu, J., Song, L., & Xie, R. (2016, November). Shot boundary detection using convolutional neural networks. In *2016 Visual Communications and Image Processing (VCIP)* (pp. 1-4). IEEE.

[27] Ramírez, S. C., & Canales, F. S. (2014, July). Deceived bilateral filter for improving the classification of football players from tv broadcast. In *3rd IEEE International Work-Conference on Bioinspired Intelligence* (pp. 98-105). IEEE.

[28] Rotman, D., Porat, D., & Ashour, G. (2016, December). Robust and efficient video scene detection using optimal sequential grouping. In *2016 IEEE International Symposium on Multimedia (ISM)* (pp. 275-280). IEEE.

[29] Rotman, D., Porat, D., & Ashour, G. (2017, October). Robust video scene detection using multimodal fusion of optimally grouped features. In *2017 IEEE 19th International Workshop on Multimedia Signal Processing (MMSP)* (pp. 1-6). IEEE.

[30] National Institude of Standards and Technology (2018). TRECVid 2007. Retrieved July 2019, from https://trecvid.nist.gov/trecvid.data.html

[31] Lu, Z. M., & Shi, Y. (2013). Fast video shot boundary detection based on SVD and pattern matching. *IEEE Transactions on Image processing*, *22*(12), 5136-5145.

[32] Sun, J., & Wan, Y. (2014, December). A novel metric for efficient video shot boundary detection. In *2014 IEEE Visual Communications and Image Processing Conference* (pp. 45-48). IEEE.