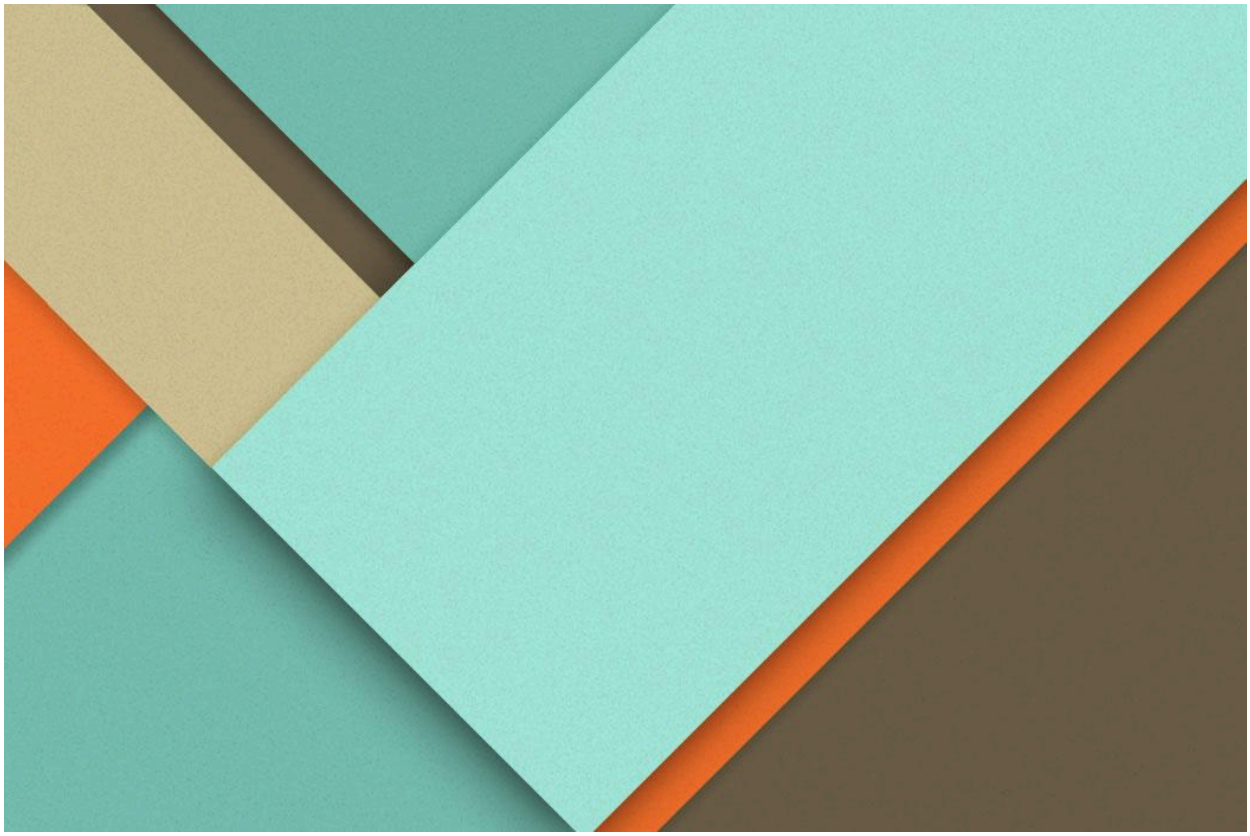


Video Segmentation Tool

03.25.2024

Shubham Laxmikant Deshmukh

MEng Computer Science
Virginia Tech

Overview

The Video Segmentation Tool is developed to segment long videos into smaller clips based on shot boundaries. The tool utilizes computer vision techniques to detect transitions between shots and extract relevant segments from the input video. It provides a user-friendly graphical interface for interaction, allowing users to select input video files, specify output directories, and segment videos effortlessly.

Previous Work

A lot of work is done in the field of Shot boundary detection in computer Vision. You could see many GitHub repos regarding this.

- Helios Zoa has used the technique of finding that the difference of the selected frame is more than 3 times the average difference of the other frames in the window. Basically using simple opencv libraries and making an algorithm around it. [1]
- Another was using traditional AI algorithms like SVM, ANN and KNN, where hoffsupes derived that SVM has better accuracy than other models achieving accuracy of 92.99%. [2]
- The best job is done by TransNet V2 dilated DCNN architecture which has a better F1 score than any other model. [3]

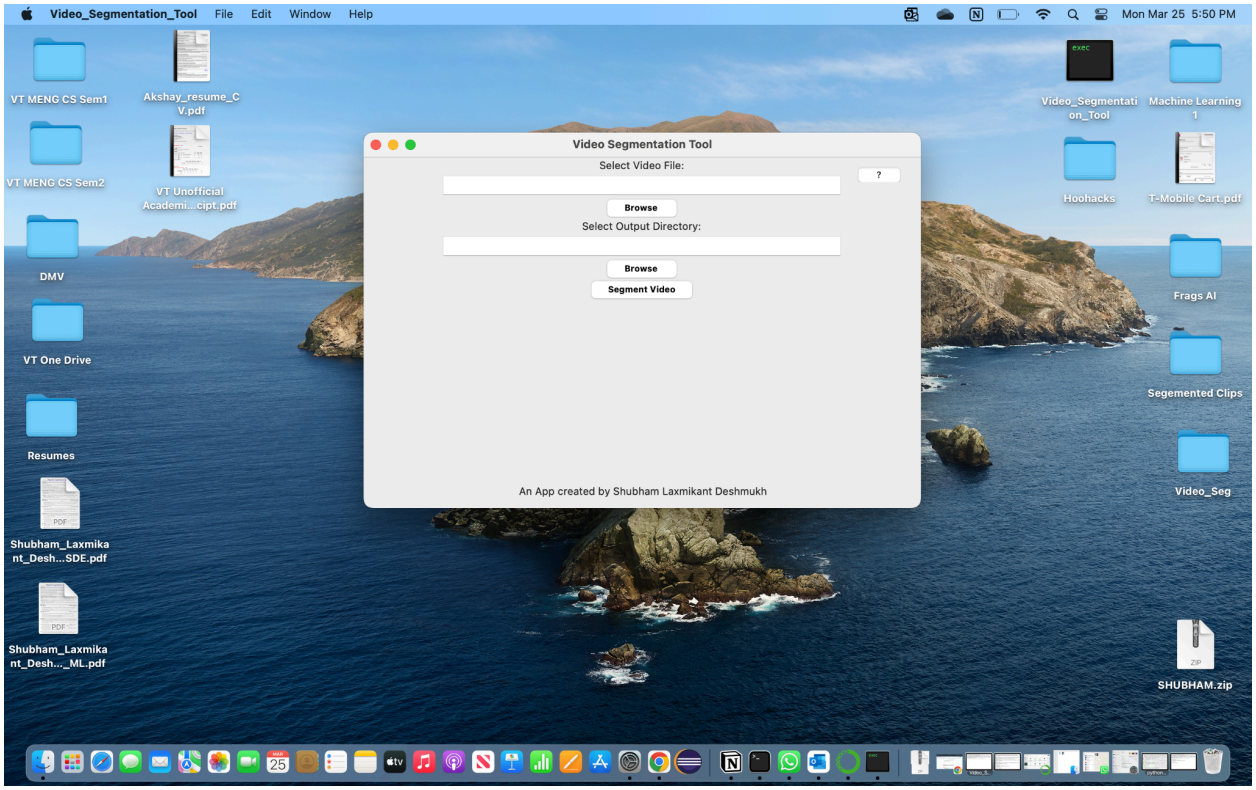
Methodology

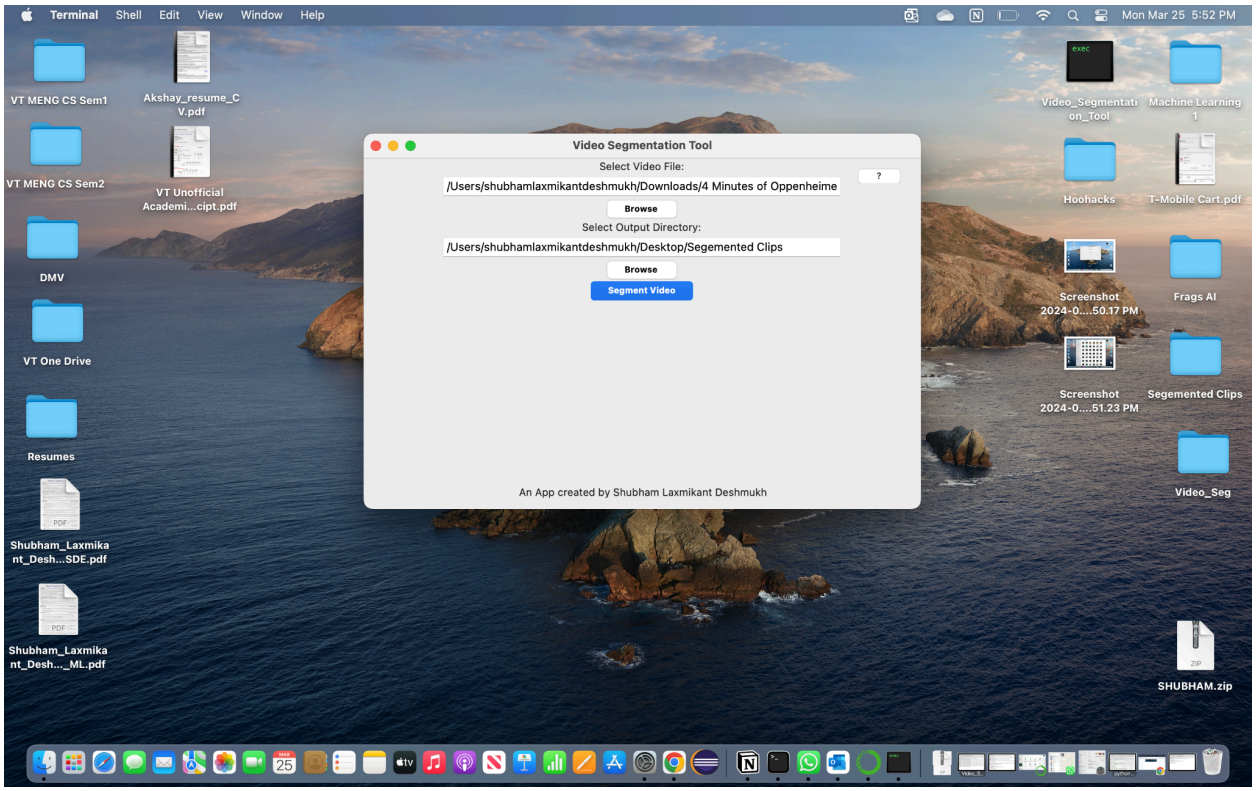
The way I have approached this task is:

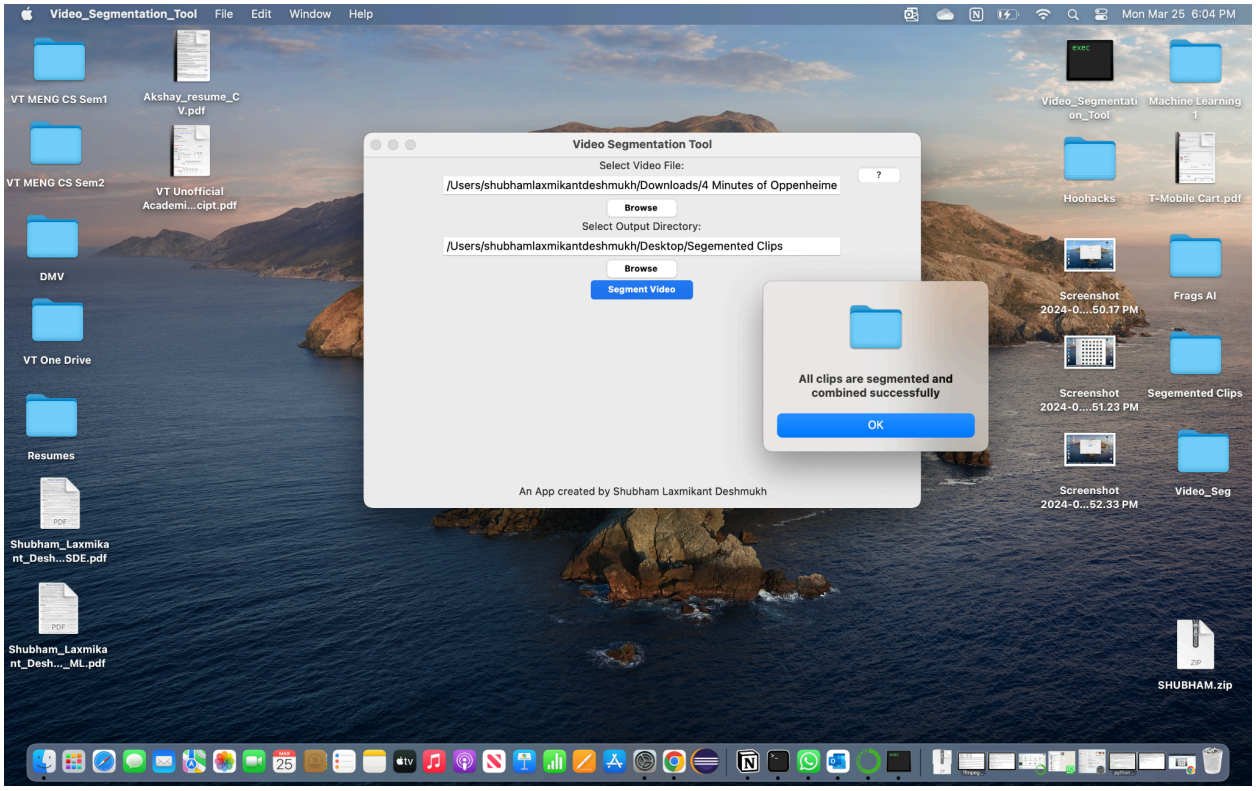
- First, take each frame of the video clip that you want to segment. Then covert it into a grayscale image. Then take out its histogram values with 256 bins, it will output an array. Normalize those values.
- Those values can be used to find out the to find out the difference between the previous frame and the current frame using the **cv2.compareHist function**. If that hist_diff is greater than the threshold (0.32, which I manually figured out with lots of testing) then its **cap.get(cv2.CAP_PROP_POS_MSEC)** or its millisecond time in the video is appended in the shot_boundry_detection array.
- Then the values of the shot_boundry_detection array are used to Video write or save separate clips from the given input video.
- As **OpenCV** only operates on frames and not on audio, I clipped audio too on the same time as I clipped the video files using the **audio.subclip** function from **moviepy** library.
- Then I combined these two video and audio clips into one to get the desired output using the moviepy library.

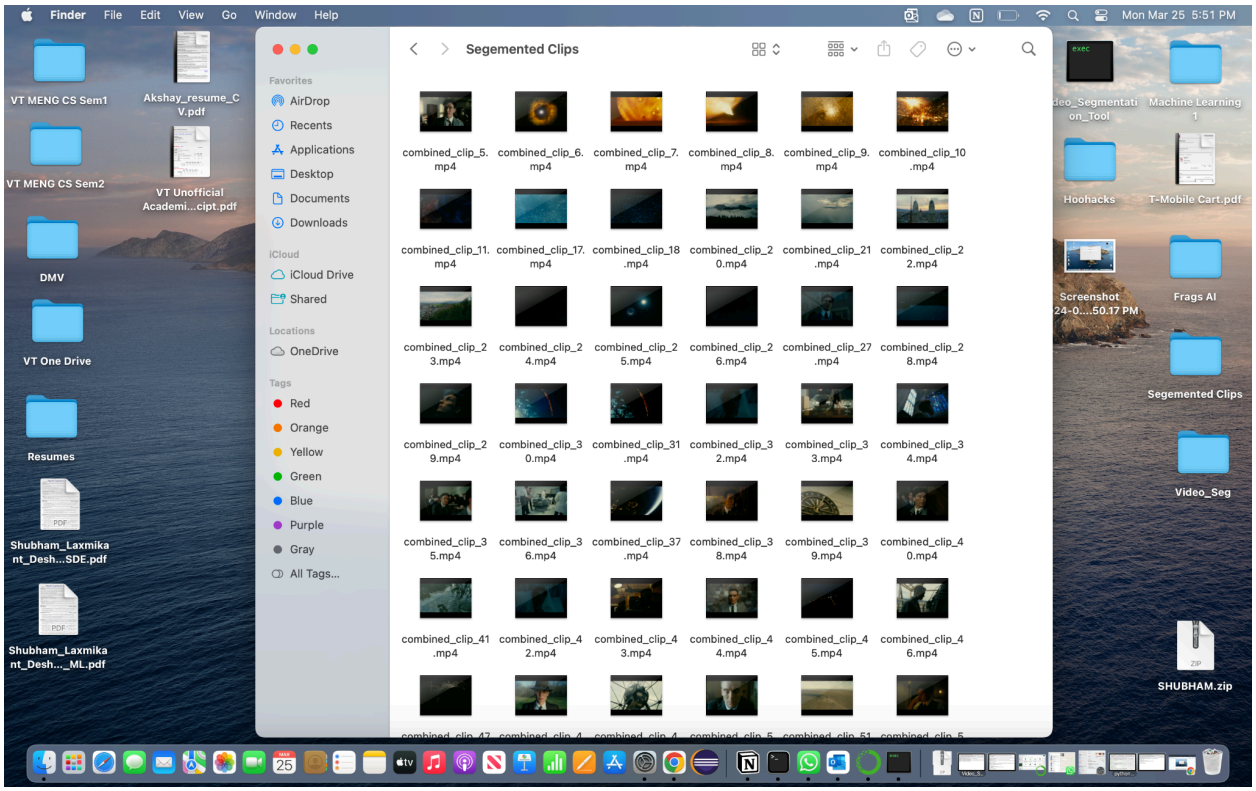
- The GUI was done using the tkinter library.
- I deployed the entire application using the **pyinstaller** library.

Results:




Terminal Shell Edit View Window Help
Mon Mar 25 5:52 PM
Video Segmentation Tool
Select Video File:
/Users/shubhamlaxmikantdeshmukh/Downloads/4 Minutes of Oppenheime
Browse
Select Output Directory:
/Users/shubhamlaxmikantdeshmukh/Desktop/Segemented Clips
Browse
Segment Video
An App created by Shubham Laxmikant Deshmukh
VT MENG CS Sem1
Akshay_resume_CV.pdf
VT MENG CS Sem2
VT Unofficial Academi...cipt.pdf
DMV
VT One Drive
Resumes
Shubham_Laxmikant_Desh...SDE.pdf
Shubham_Laxmikant_Desh..._ML.pdf
Video_Segmentation_Tool
Machine Learning 1
Hoohacks
T-Mobile Cart.pdf
Screenshot 2024-0....50.17 PM
Frags AI
Screenshot 2024-0....51.23 PM
Segemented Clips
Video_Seg
SHUBHAM.zip


Video_Segmentation_Tool File Edit Window Help
Mon Mar 25 6:04 PM
Video Segmentation Tool
Select Video File:
/Users/shubhamlaxmikantdeshmukh/Downloads/4 Minutes of Oppenheime
Browse
Select Output Directory:
/Users/shubhamlaxmikantdeshmukh/Desktop/Segemented Clips
Browse
Segment Video
All clips are segmented and combined successfully
OK
An App created by Shubham Laxmikant Deshmukh
VT MENG CS Sem1
Akshay_resume_CV.pdf
VT MENG CS Sem2
VT Unofficial Academi...cipt.pdf
DMV
VT One Drive
Resumes
Shubham_Laxmikant_Desh...SDE.pdf
Shubham_Laxmikant_Desh..._ML.pdf
Video_Segmentation_Tool
Machine Learning 1
Hoohacks
T-Mobile Cart.pdf
Screenshot 2024-0....50.17 PM
Frags AI
Screenshot 2024-0....51.23 PM
Segemented Clips
Screenshot 2024-0...52.33 PM
Video_Seg
SHUBHAM.zip



All kinds of video files can be inserted as an input in this tool. I have also tested various videos from youtube with varied quality like (720p, 360p, 144p) and I am getting good results when I keep the threshold as 0.32.

Conclusion:

The Video Segmentation Tool offers an efficient solution for segmenting long videos into smaller clips, facilitating easier management and analysis of video content. With its user-friendly interface and robust functionality, the tool serves as a valuable asset for content creators, researchers, and video enthusiasts. I could not evaluate this methodology using metrics like the F1 score, but this gives the desired output. This might be the best approach using traditional algorithms of OpenCV.

Future Work:

I am currently working on the pre-trained model TransNet V2 as I mentioned earlier to detect the shot boundries, I think using a good Deep learning pre-trained model will give better results. The UI of the app can be improved too. This app can be deployed on web using GCP or AWS.

References

[1] https://github.com/HeliosZhao/Shot-Boundary-Detection

[2]https://github.com/hoffsupes/Shot-boundary-detection-using-SVM-s-Aritificial-Neural-Networks-and-kNN
[3] https://arxiv.org/pdf/2008.04838.pdf

GitHub Link

https://github.com/Shubhs0411/Clip-Segmentation-Shubham