

```

from moviepy.editor import VideoFileClip
from PIL import Image, ImageChops
import numpy as np
import matplotlib.pyplot as plt
from PIL import Image, ImageChops, ImageDraw

def load_video_frames(file_path, fps=1):
    clip = VideoFileClip(file_path)
    frames = [frame for frame in clip.iter_frames(fps=fps)]
    return frames

video_file = '/content/MV Recording 2 Assignment 3.mp4'
frames = load_video_frames(video_file, fps=1)
print(f'Total frames extracted: {len(frames)}')

 Total frames extracted: 70

def segment_frame(frame, lower_bound, upper_bound):
    img = Image.fromarray(frame)
    hsv = img.convert('HSV')
    mask = np.zeros(hsv.size[::-1], dtype=np.uint8)

    for x in range(hsv.width):
        for y in range(hsv.height):
            pixel = hsv.getpixel((x, y))
            if lower_bound[0] <= pixel[0] <= upper_bound[0] and \
                lower_bound[1] <= pixel[1] <= upper_bound[1] and \
                lower_bound[2] <= pixel[2] <= upper_bound[2]:
                mask[y, x] = 255

    segmented = Image.fromarray(np.where(mask[:, :, None], frame, [0, 0, 0]).astype(np.uint8))
    return mask, segmented

lower_bound = (30, 50, 50)
upper_bound = (80, 255, 255)

segmented_frames = [segment_frame(frame, lower_bound, upper_bound) for frame in frames]

def detect_scene_cuts(frames, threshold=1000000):
    hard_cuts = []
    soft_cuts = []

    for i in range(1, len(frames)):
        prev_img = Image.fromarray(frames[i - 1])
        curr_img = Image.fromarray(frames[i])

        diff = ImageChops.difference(prev_img, curr_img)
        diff_pixels = np.sum(np.array(diff)) > 50

        if diff_pixels > threshold:
            hard_cuts.append(i)
        else:
            hist_diff = sum(abs(a - b) for a, b in zip(prev_img.histogram(), curr_img.histogram()))
            if hist_diff > 10000:
                soft_cuts.append(i)

    return hard_cuts, soft_cuts

hard_cuts, soft_cuts = detect_scene_cuts(frames)
print(f'Hard cuts detected at frames: {hard_cuts}')
print(f'Soft cuts detected at frames: {soft_cuts}')

 Hard cuts detected at frames: [6, 11, 12, 13, 15, 16, 17, 19, 24, 25, 26, 27, 29, 30, 31, 32, 33, 34, 35, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59]
  Soft cuts detected at frames: [1, 2, 3, 4, 5, 7, 8, 9, 10, 14, 18, 20, 21, 22, 23, 28, 36, 37, 44, 45, 50, 53, 56, 57, 58, 59]

< >

def mark_scene_cuts(frames, hard_cuts, soft_cuts):
    marked_frames = []
    for i, frame in enumerate(frames):
        img = Image.fromarray(frame)
        draw = ImageDraw.Draw(img)

        if i in hard_cuts:
            draw.text((10, 10), "Hard Cut", fill=(255, 0, 0))
        elif i in soft_cuts:
            draw.text((10, 10), "Soft Cut", fill=(0, 0, 255))

    marked_frames.append(img)
    return marked_frames

```

```
marked_frames.append(img)
return marked_frames

cut_frames = mark_scene_cuts(frames, hard_cuts, soft_cuts)

import math
from PIL import ImageFilter

def segment_frame(frame):
    segmented = frame.convert("L").filter(ImageFilter.FIND_EDGES)
    return segmented

def display_results(frames, hard_cuts, soft_cuts, max_columns=3):
    total_hard = len(hard_cuts)
    total_soft = len(soft_cuts)
    total_cuts = total_hard + total_soft

    if total_cuts == 0:
        print("No scene cuts detected.")
        return

    rows_hard = math.ceil(total_hard / max_columns)
    rows_soft = math.ceil(total_soft / max_columns)

    fig, axes = plt.subplots((rows_hard + rows_soft) * 2, max_columns, figsize=(20, (rows_hard + rows_soft) * 5))
    fig.suptitle('Scene Cuts and Segmentation', fontsize=24)
    axes = axes.flatten()

    for i, cut in enumerate(hard_cuts):
        axes[2 * i].imshow(frames[cut])
        axes[2 * i].set_title(f'Hard Cut at Frame {cut}', fontsize=14)
        axes[2 * i].axis('off')

        segmented_frame = segment_frame(frames[cut])
        axes[2 * i + 1].imshow(segmented_frame, cmap='gray')
        axes[2 * i + 1].set_title(f'Segmented Frame {cut}', fontsize=14)
        axes[2 * i + 1].axis('off')

    offset = rows_hard * max_columns * 2
    for j, cut in enumerate(soft_cuts):
        axes[offset + 2 * j].imshow(frames[cut])
        axes[offset + 2 * j].set_title(f'Soft Cut at Frame {cut}', fontsize=14)
        axes[offset + 2 * j].axis('off')

        segmented_frame = segment_frame(frames[cut])
        axes[offset + 2 * j + 1].imshow(segmented_frame, cmap='gray')
        axes[offset + 2 * j + 1].set_title(f'Segmented Frame {cut}', fontsize=14)
        axes[offset + 2 * j + 1].axis('off')

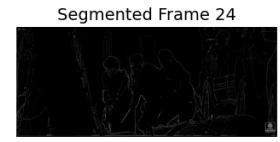
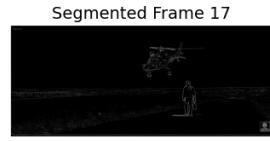
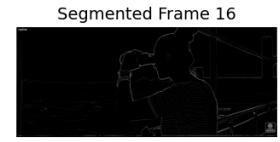
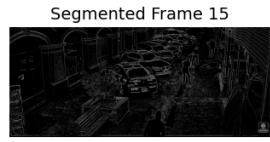
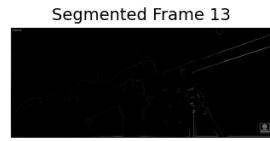
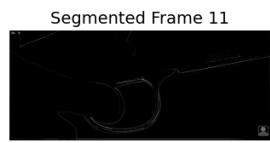
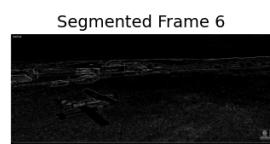
    for ax in axes[total_cuts * 2:]:
        ax.axis('off')

    plt.subplots_adjust(wspace=0.3, hspace=0.5)
    plt.show()

display_results(cut_frames, hard_cuts, soft_cuts, max_columns=3)
```



## Scene Cuts and Segmentation





Segmented Frame 26



Hard Cut at Frame 27



Segmented Frame 27



Hard Cut at Frame 29



Segmented Frame 29



Hard Cut at Frame 30



Segmented Frame 30



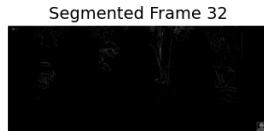
Hard Cut at Frame 31



Segmented Frame 31



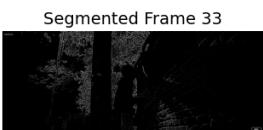
Hard Cut at Frame 32



Segmented Frame 32



Hard Cut at Frame 33



Segmented Frame 33



Hard Cut at Frame 34



Segmented Frame 34



Hard Cut at Frame 35



Segmented Frame 35



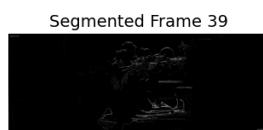
Hard Cut at Frame 36



Segmented Frame 36



Hard Cut at Frame 37



Segmented Frame 37



Hard Cut at Frame 38



Segmented Frame 38



Hard Cut at Frame 39



Segmented Frame 39



Hard Cut at Frame 40



Segmented Frame 40



Hard Cut at Frame 41



Segmented Frame 41



Segmented Frame 42



Segmented Frame 42



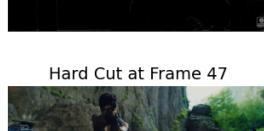
Hard Cut at Frame 43



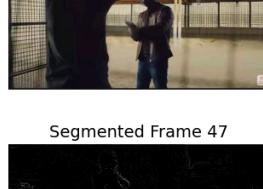
Segmented Frame 43



Segmented Frame 44



Segmented Frame 44



Segmented Frame 45



Hard Cut at Frame 45



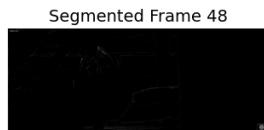
Segmented Frame 45



Segmented Frame 46



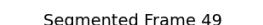
Hard Cut at Frame 46



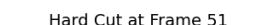
Segmented Frame 46



Segmented Frame 47



Segmented Frame 47



Hard Cut at Frame 48



Segmented Frame 48



Hard Cut at Frame 49



Segmented Frame 49



Hard Cut at Frame 52



Segmented Frame 52



Hard Cut at Frame 54



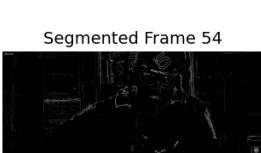
Segmented Frame 54



Hard Cut at Frame 55



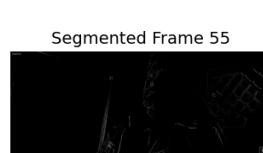
Segmented Frame 55



Hard Cut at Frame 62



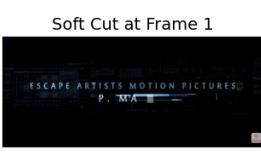
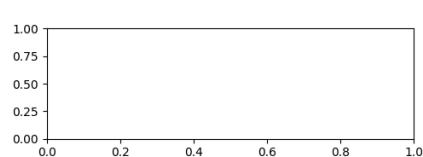
Segmented Frame 62



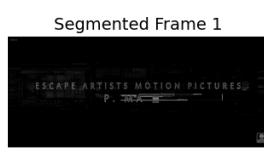
Hard Cut at Frame 63



Segmented Frame 63



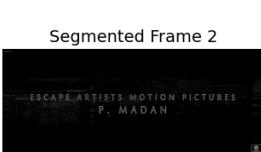
Soft Cut at Frame 1



Segmented Frame 1



Soft Cut at Frame 2



Segmented Frame 2



Soft Cut at Frame 3



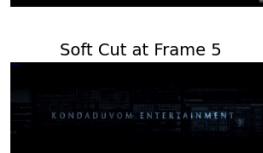
Segmented Frame 3



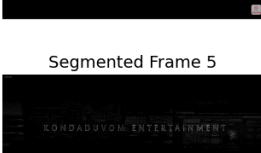
Soft Cut at Frame 4



Segmented Frame 4



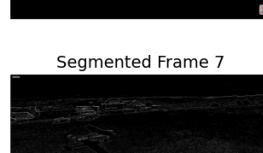
Soft Cut at Frame 5



Segmented Frame 5



Soft Cut at Frame 7



Segmented Frame 7



Soft Cut at Frame 8



Segmented Frame 8



Soft Cut at Frame 9



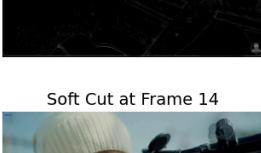
Segmented Frame 9



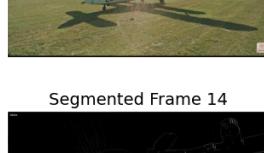
Soft Cut at Frame 10



Segmented Frame 10



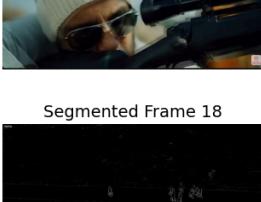
Soft Cut at Frame 14



Segmented Frame 14



Soft Cut at Frame 18



Segmented Frame 18



Soft Cut at Frame 20



Segmented Frame 20

Soft Cut at Frame 21



Segmented Frame 21



Soft Cut at Frame 22



Segmented Frame 22



Soft Cut at Frame 23



Segmented Frame 23



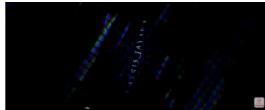
Soft Cut at Frame 28



Segmented Frame 28



Soft Cut at Frame 36



Segmented Frame 36



Soft Cut at Frame 37



Segmented Frame 37



Soft Cut at Frame 44



Segmented Frame 44



Soft Cut at Frame 45



Segmented Frame 45



Soft Cut at Frame 50



Segmented Frame 50



Soft Cut at Frame 53



Segmented Frame 53



Soft Cut at Frame 56



Segmented Frame 56



Soft Cut at Frame 57



Segmented Frame 57



Soft Cut at Frame 58



Segmented Frame 58



Soft Cut at Frame 59



Segmented Frame 59



Soft Cut at Frame 60



Segmented Frame 60



Soft Cut at Frame 61



Segmented Frame 61



Soft Cut at Frame 64



Segmented Frame 64



Soft Cut at Frame 65



Segmented Frame 65



Soft Cut at Frame 66



Segmented Frame 66



Soft Cut at Frame 67





Segmented Frame 67



Soft Cut at Frame 68



Segmented Frame 68



Soft Cut at Frame 69



Segmented Frame 69