Shri Ramdeobaba College of Engineering and Management
Nagpur, 440013

Department of Computer Science Engineering

FDVIP Lab

Name : *Shantanu Mane*
Roll No. : *E63*
Batch : *CSE-AIML*
Date : *19/4/2023*

AIM - To study concepts of video processing and perform video processing using OpenCV

1. Capturing and displaying video
2. Capturing and saving video
3. Displaying properties of video

Importing Dependencies

```
import cv2
import matplotlib.pyplot as plt
import numpy as np
```

Capturing and Displaying Video

```
vid_capture = cv2.VideoCapture(r"../data/video.mp4")

if not vid_capture.isOpened():
    print("Error opening the video file")
else:
    fps = vid_capture.get(5)
    print('Frames per second : ', fps, 'FPS')
    frame_count = vid_capture.get(7)
    print('Frame count : ', frame_count)

while vid_capture.isOpened():
    ret, frame = vid_capture.read()
    if ret:
        cv2.imshow('Frame', frame)
        key = cv2.waitKey(20)

        if key == ord('q'):
            break
    else:
        break

vid_capture.release()
```

Capturing and Saving Video

```
video = cv2.VideoCapture(0)

if not video.isOpened():
    print("Error reading video file")

frame_width = int(video.get(3))
frame_height = int(video.get(4))

size = (frame_width, frame_height)

result = cv2.VideoWriter('../data/saved_vid.avi',
                         cv2.VideoWriter_fourcc(*'MJPG'),
                         10, size)

while (True):
    ret, frame = video.read()

    if ret:
        result.write(frame)

        cv2.imshow('Frame', frame)

        if cv2.waitKey(1) & 0xFF == ord('s'):
            break

    else:
        break

video.release()
result.release()
```

Displaying Properties of Video

```
capture = cv2.VideoCapture(0)
```

```
print("CV_CAP_PROP_FRAME_WIDTH: '{}'".format(capture.get(cv2.CAP_PROP_FRAME_WIDTH)))
print("CV_CAP_PROP_FRAME_HEIGHT : '{}'".format(capture.get(cv2.CAP_PROP_FRAME_HEIGHT)))
print("CAP_PROP_FPS : '{}'".format(capture.get(cv2.CAP_PROP_FPS)))
print("CAP_PROP_POS_MSEC : '{}'".format(capture.get(cv2.CAP_PROP_POS_MSEC)))
print("CAP_PROP_FRAME_COUNT  : '{}'".format(capture.get(cv2.CAP_PROP_FRAME_COUNT)))
print("CAP_PROP_BRIGHTNESS : '{}'".format(capture.get(cv2.CAP_PROP_BRIGHTNESS)))
print("CAP_PROP_CONTRAST : '{}'".format(capture.get(cv2.CAP_PROP_CONTRAST)))
print("CAP_PROP_SATURATION : '{}'".format(capture.get(cv2.CAP_PROP_SATURATION)))
print("CAP_PROP_HUE : '{}'".format(capture.get(cv2.CAP_PROP_HUE)))
print("CAP_PROP_GAIN  : '{}'".format(capture.get(cv2.CAP_PROP_GAIN)))
print("CAP_PROP_CONVERT_RGB : '{}'".format(capture.get(cv2.CAP_PROP_CONVERT_RGB)))

while capture.isOpened():
    ret, frame = capture.read()
    if ret:
        cv2.imshow('Frame', frame)
        key = cv2.waitKey(20)

        if key == ord('q'):
            break
    else:
        break

capture.release()
```

Output

```
CV_CAP_PROP_FRAME_WIDTH: '640.0'
CV_CAP_PROP_FRAME_HEIGHT : '480.0'
CAP_PROP_FPS : '30.0'
CAP_PROP_POS_MSEC : '0.0'
CAP_PROP_FRAME_COUNT  : '-1.0'
CAP_PROP_BRIGHTNESS : '128.0'
CAP_PROP_CONTRAST : '32.0'
CAP_PROP_SATURATION : '64.0'
CAP_PROP_HUE : '0.0'
CAP_PROP_GAIN  : '-1.0'
CAP_PROP_CONVERT_RGB : '1.0'
```