

Practical-6

Student Name: Gauri Prabhakar

UID: 18BCS6201

Branch: 18AITAIML-2

Section/Group: B

Semester: 7

Date of Performance: 12th October, 2021

Subject Name: Computer Vision Lab

Subject Code: CSF - 432

1. Aim/Overview of the practical:

To implement pose detection using mediapipe in python and OpenCV.

2. Task to be done:

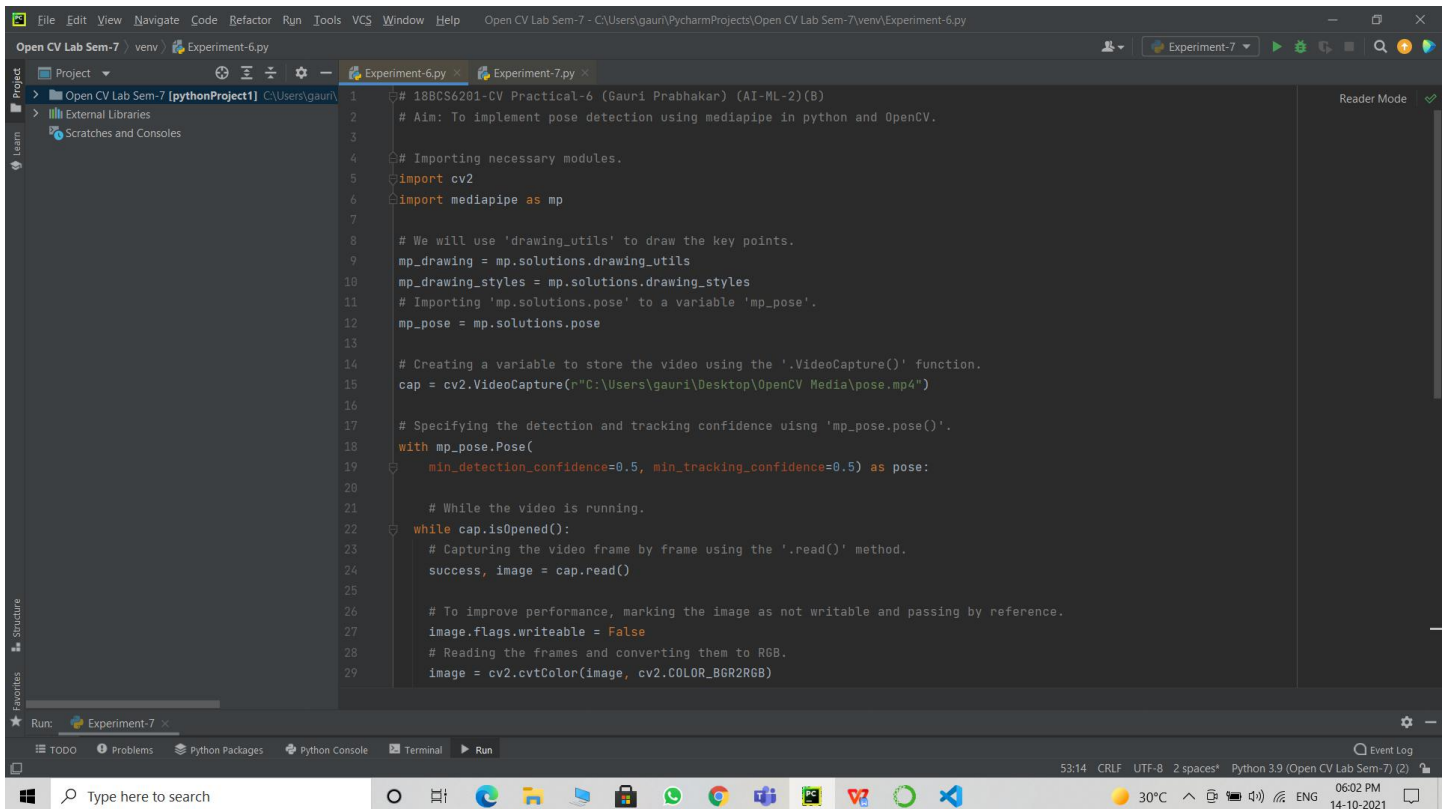
To implement pose detection using mediapipe in python and OpenCV.

3. Steps to be followed:

1. Importing necessary modules.
2. We will use 'drawing_utils' to draw the key points.
3. Importing 'mp.solutions.pose' to a variable 'mp_pose'.
4. Creating a variable to store the video using the '.VideoCapture()' function.
5. Specifying the detection and tracking confidence using 'mp_pose.pose()'.
6. While the video is running.
7. Capturing the video frame by frame using the '.read()' method.
8. To improve performance, marking the image as not writable and passing by reference.
9. Reading the frames and converting them to RGB.
10. Detecting poses in the frame using the function 'pose.process()'.
11. Drawing the pose annotations on the image.
12. Reading the frames and converting them to RGB.
13. Connecting the key points using the function 'mp_drawing.draw_landmarks()'.

14. Rendering the video with effective Posetracking to the console by using the function `'.imshow()'`.
15. Setting up `'.waitkey()'` to wait for a specific time until any key is pressed and break the loop.
16. `'.waitkey(1)'` displays a frame for 1ms after which it moves to the next frame in the video.
17. Setting 'x' as the quitting button.
18. Releasing the variable/object 'cap'.

4. Result/Output/Writing Summary:



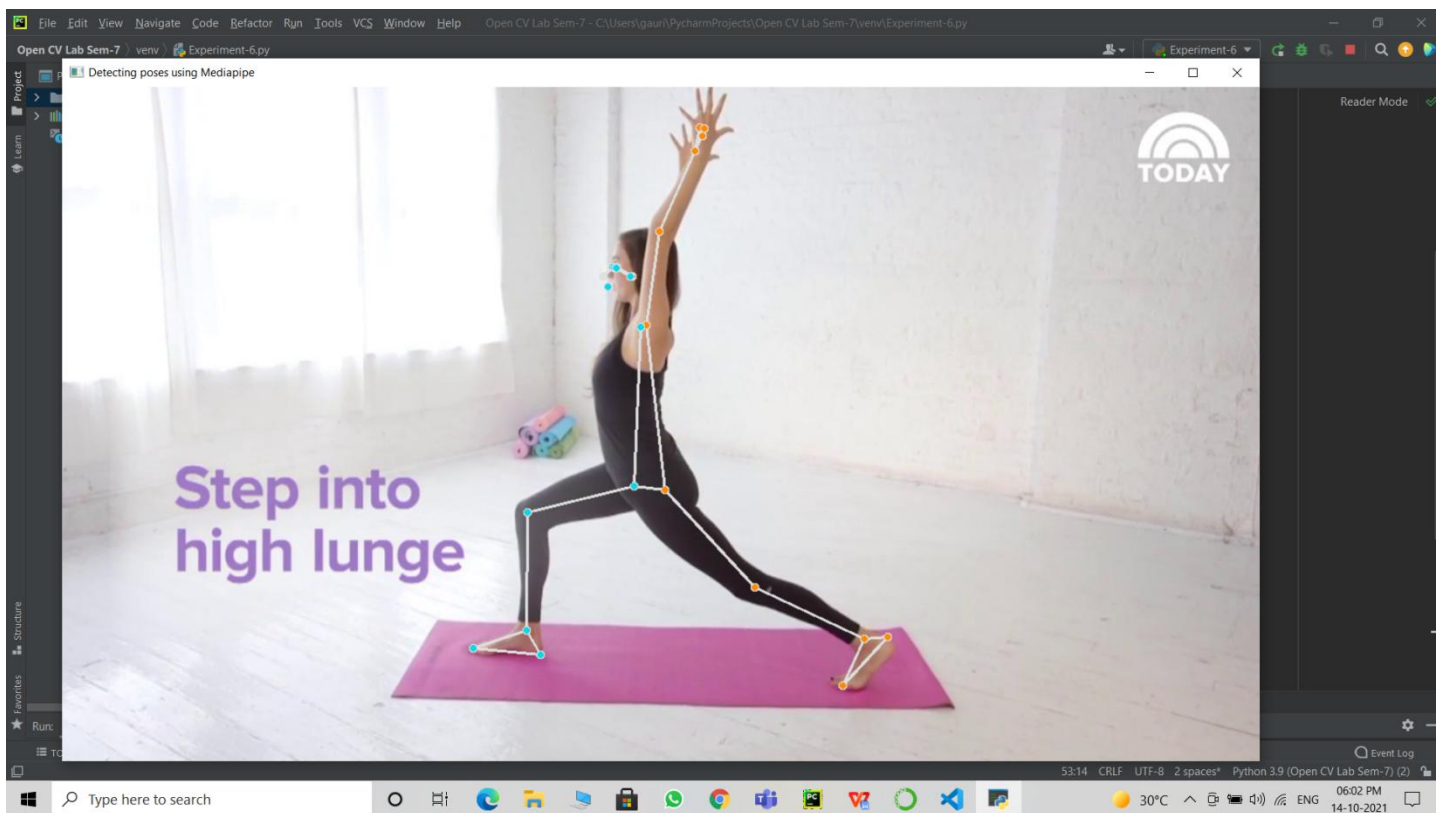
```
1  # 18BCS6201-CV Practical-6 (Gauri Prabhakar) (AI-ML-2)(B)
2  # Aim: To implement pose detection using mediapipe in python and OpenCV.
3
4  # Importing necessary modules.
5  import cv2
6  import mediapipe as mp
7
8  # We will use 'drawing_utils' to draw the key points.
9  mp_drawing = mp.solutions.drawing_utils
10 mp_drawing_styles = mp.solutions.drawing_styles
11 # Importing 'mp.solutions.pose' to a variable 'mp_pose'.
12 mp_pose = mp.solutions.pose
13
14 # Creating a variable to store the video using the '.VideoCapture()' function.
15 cap = cv2.VideoCapture(r"C:\Users\gaurn\Desktop\OpenCV Media\pose.mp4")
16
17 # Specifying the detection and tracking confidence using 'mp_pose.pose()'.
18 with mp_pose.Pose(
19     min_detection_confidence=0.5, min_tracking_confidence=0.5) as pose:
20
21     # While the video is running.
22     while cap.isOpened():
23         # Capturing the video frame by frame using the '.read()' method.
24         success, image = cap.read()
25
26         # To improve performance, marking the image as not writable and passing by reference.
27         image.flags.writeable = False
28         # Reading the frames and converting them to RGB.
29         image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
```

```
File Edit View Navigate Code Refactor Run Tools VCS Window Help Open CV Lab Sem-7 - C:\Users\gaurn\PycharmProjects\Open CV Lab Sem-7\venv\Experiment-6.py
Open CV Lab Sem-7 venv Experiment-6.py Experiment-6.py Experiment-7.py
Project
  Open CV Lab Sem-7 [pythonProject1] C:\Users\gaurn\
  External Libraries
  Scratches and Consoles
Experiment-6.py
28 # Reading the frames and converting them to RGB.
29 image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
30 # Detecting poses in the frame using the function 'pose.process()'.
31 results = pose.process(image)
32
33 # Drawing the pose annotations on the image.
34 image.flags.writeable = True
35
36 # Reading the frames and converting them to RGB.
37 image = cv2.cvtColor(image, cv2.COLOR_RGB2BGR)
38
39 # Connecting the key points using the function 'mp_drawing.draw_landmarks()'.
40 mp_drawing.draw_landmarks(image, results.pose_landmarks, mp_pose.POSE_CONNECTIONS,
41                           landmark_drawing_spec=mp_drawing_styles.get_default_pose_landmarks_style())
42
43 # Rendering the video with effective Posetracking to the console by using the function 'imshow()'.
44 cv2.imshow('Detecting poses using Mediapipe', image)
45
46 # Setting up '.waitkey()' to wait for a specific time until any key is pressed and break the loop.
47 # '.waitkey(1)' displays a frame for 1ms after which it moves to the next frame in the video.
48 # Setting 'x' as the quitting button.
49 if cv2.waitKey(5) & 0xFF == ord('x'):
50     break
51
52 # Releasing the variable/object 'cap'.
53 cap.release()
```

Run: Experiment-7

53:14 CRLF UTF-8 2 spaces* Python 3.9 (Open CV Lab Sem-7) (2)

Type here to search 30°C 06:02 PM 14-10-2021



5. Learning outcomes (What I have learnt):

- Open CV modules.
- The mediapipe library.
- Detect poses using the mediapipe library.
- Pose tracking a saved video.
- Highlighting key points.

Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.			
2.			
3.			