Gym Posture Recognition and Feedback Generation Using Mediapipe and OpenCV

Tharani G[1], Gopika Sri R[2], Hemapriya R[3], Karthiga M[4]
Department of Computer Science and Engineering, Bannari Amman Institute of Technology, Erode, India

Abstract – Everyone benefits from exercise and physical activity. Staying active can benefit you in a variety of ways, regardless of your health or physical abilities. In reality, research shows that "taking it easy" is dangerous. Exercises are often practiced in training centers, through personal tutors, and may also be learned on one's own with the help of the recorded clips, etc. In fast-paced lifestyles, many people prefer self-learning because the above-mentioned resources might not be available all the time. But in self-learning, one may not find an incorrect pose. One's health might suffer from improper posture, which can cause both short-term acute discomfort and long-term chronic problems. We investigated many applications that can be implemented using data provided by a pre-trained posture estimation model called MediaPipe. Applications include motion capture, gait analysis, sign language detection etc. Building a gym posture monitoring system, which analyses and tracks user motions and postures for faults, is the major goal of this project. The user is then notified of his/her error in the posture through a display screen or a wireless speaker. The inaccurate body pose of the user can be pointed out in real-time so that the user can rectify his/her mistakes.

Keywords – *Pose Estimation Model, Deep Learning, Mediapipe, OpenCV, Python.*

---

1 INTRODUCTION

Out of our world's population, around 39 % of adults are overweight. This shows how exercising is important in daily life. Exercise not only helps to reduce the body weight, but also if done regularly helps us stay active with good blood circulation and also to maintain a healthy weight, fit body and a peaceful mind. To maintain a fit and healthy body just like the way you used to go to gyms. Lifting weights is a great way to develop muscles, protect bones, burn calories, and stay fit. Maybe some people don't know where to start or how to perform the exercises. You might be tempted to simply repeat the exercises as others do them, but this could lead to incorrect results. Visiting the gyms or attending yoga classes and getting trained under a trainer is not cheap and not affordable for all. Another option is self-training which will have the steps to do the exercise routine which is pre-recorded but lacks feedback. Without proper feedback about our postures, injuries can happen and it will do more harm than good. By using human pose estimation technique's, we could determine the position of a human being at key points. By doing so, we would be able to gauge or assess the pose of the human body and provide commentary on it. And a variety of methods have been used to accurately and effectively recognize e the human position in real-time. With increasing computing power deep learning models have vastly improved and are the most used approach for body pose estimation. Applications include Biometric authentication (gait analysis), Video surveillance (anomaly detection), Exercise monitoring system, Animation and motion capture, etc. Because of our ever-busy lives we do not give enough importance to one's mental and physical health. Our project tries to fill that gap with feedback on the exercise routine in real-time.

2. LITERATURE REVIEW

The following are journal and research papers of pose estimation which have been released earlier. [1] The estimation of human posture using deep high resolution representation learning. The Pose estimation has advanced significantly and the typical method is creating heatmaps for each joint and fine-tuning offsets for each point. The models for a single individual are much larger than what is necessary for real-time interpretation on mobile phones, despite the fact that this selection of heatmaps scalable to numerous persons with little overhead. [2] This review began with Pose estimate was divided into single-person and multi-person pipelines, with sub-categories being formed for each. A person is discovered in one at that point. The system is able to determine which cameras across all locations this particular person is visible in [3]. The article outlines a technique for automatically recovering these lines by Observing motion in the environment and analysing those images, in addition to the ability to identify homographs between views. Regression-based approaches, in comparison to heatmap - based techniques, aim to forecast the mean coordinate values while being less computationally intensive and more scalable, frequently failing to resolve the underlying uncertainty [4]. Have demonstrated that, even with fewer factors, the layered hourglass architecture significantly improves the quality of the forecast. The paper on education precise human pose prediction from effective Results were explained by annotation dividing up the posing space [5]. The versatile strong can capture appearance handled bigger and nonlinear classifiers amounts of information utilised for training and employed AMT (amazon mechanical turk) which offers the best value at the highest quality. This paper's primary objective is to present a complete study of the most extensively applied effective models, which will help to develop better human pose estimate models utilizing better evaluation metrics or effective backbone architecture. Additionally, it gives readers numerous options for fusing model architectures [14].

3. PROPOSED METHODOLOGY

1. IMAGE ACQUISITION

The input image is taken as input using a camera which can be either a smartphone camera which is now ubiquitously available or we can use a webcam which is the good way for capturing images since most of us have any one of this type of camera input solution. Camera acts as the input component of the system. The camera source can be from a webcam, mobile camera. Camera is used for image acquisition and as data input to the model. Our project is compatible with an RGB (Red, Green, Blue) camera. A reference square box is shown on screen and the user is asked to stand at a particular distance so that his/her whole body is within this square boundary. The built-in camera or we can use a separate camera module i to capture the image of the user continuously during the routine. Which is then sent to the system (smartphone or computer) for processing.
The Figure 1 represents the overall block diagram.

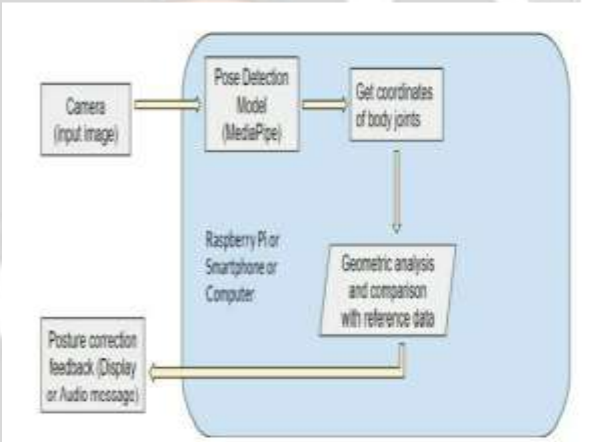

Figure 1. Overall Block Diagram

2. DATA SET

There are ‘n’ number of datasets available for human pose estimation like MPII which is benchmark for evaluating of articulate human pose estimation it contains every day human activities. The Dataset CMU Panoptic (OpenPose) and coco dataset. Out of these datasets we have used coco dataset which outperforms yoga or fitness use cases. we have trained using the COCO dataset [13]. This dataset contains a 200,000 images and it has 250,000 people with keypoints were each individual instace will be labeled with 17 joints.

The outputs plotted on a person is shown in the image below Fig.2

Figure 2. BlazePose landmark Results

Output Format of COCO dataset: Nose – 0, Neck – 1, Right Shoulder – 2, Right Elbow – 3, Right Wrist – 4, Left Shoulder – 5, Left Elbow – 6, Left Wrist – 7, Right Hip – 8, Right Knee – 9, Right Ankle – 10, Left Hip – 11, Left Knee – 12, Left Ankle – 13, Right Eye – 14, Left Eye – 15, Right Ear – 16, Left Ear – 17.

3.3 POSE ESTIMATION

The input from the user is given to the mediapipe library for keypoint detection of the user's body. The result is a list of coordinates in the X,Y and Z axis for 33 major key-points of a person. These keypoints define the location of each major body part in the input image. Using these keypoints we can build an accurate skeletal orientation of the user. Below Fig. 3, the landmarks indicate the major joints and locations on a person. Landmarks are indexed from 0 to 32 total of 33 landmarks. The facial landmarking procedure uses landmarks from 0 to 10. The next landmarks from 11 to 22 are used for the detection of the upper body. The upper body indicates shoulders, wrists, elbows, hands [11]. The final landmarks from 23 to 32 are used to define the lower body consisting of the hips, knees, legs, and feet. These landmarks give the specified positions of the body in 3D space. Pose detection is achieved using the pre-trained model MediaPipe. Mediapipe the open-source library, cross-platform and a customisable machine learning solution for real-time streaming media such as audio, video and series data [10]. The library is supported in iOS, python, JavaScript and Android. The mediapipe library also provides solutions which include pose estimation, hair segmentation, face mesh, motion tracking.The input data to the MediaPipe library is the image captured by the camera in real-time. The output from the MediaPipe library is a list of corresponding key-points in X, Y and Z cartesian coordinates. These coordinates can be used to get the rough estimate of the human body structure and orientation in the given image or video stream in real time. The frame-rate mentioned in the documentation of MediaPipe library is 30 frames per second.

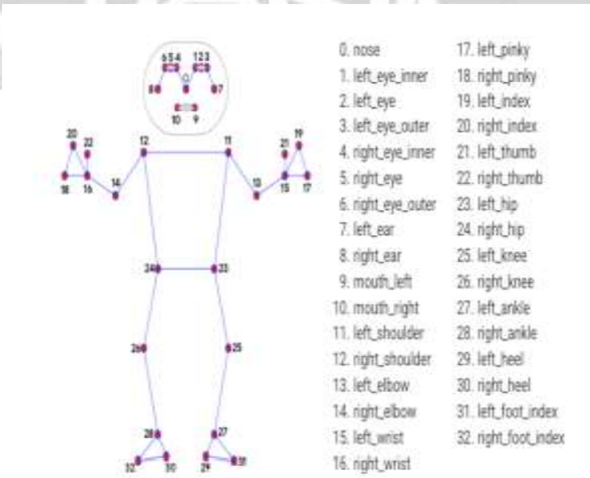

Figure 3. The 33 keypoint topology

4. EXPERIMENTAL RESULTS

4.1 TRACKING MODEL

All the 33 keypoints having 3 degrees of freedom each x, y location and visibility plus the two virtual alignments is predicted using human pose estimation model. In our model we used a regression approach that is *supervised* by a combined heat map/offset prediction of all keypoints. The below fig 4. Represents the network architecture of heat maps and offset maps.

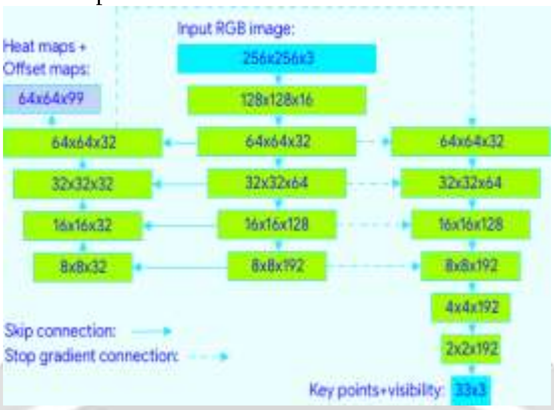

Figure 4. Network Architecture

4.2 CALCULATING ANGLES

First we get the coordinates of the threejoints which we require to get the angle calculated. Then we can calculate the slopes of the joints using NumPy. Angles are calculated in radians which then can be converted into degrees [8].

Angle = math.degrees(math.atan2(y3 - y2, x3 - x2) - math.atan2(y1 - y2, x1 - x2)) this formula is used to calculate angle between two lines without calculating slope. Finds the angle between joints (depending on which joints you select). For my left bicep curl program, I chose to find the angle at the elbow as this is vital for proper left bicurl form. The Fig.5 which is calculated for left bicepcurl.

Figure 5. final output of left bicep curl

4.3 REPETITION COUNTING

To count the repetitions, the algorithm monitors the probability of a target pose class. When the probability of the "down" pose class passes a certain threshold for the first time, counter increases by one. Then successful transition of hand from down or up increases counter.

4.4 PROPER FEED BACK TO THE USER

In our project to make the user aware of inaccurate range of motion in real-time we give concise feedback to the practitioner. So here we utilized python text to speech library which enable us to give feedback through audio and help them for improvement that he/she can correct it immediately.

5. RESULTS AND DISCUSSION

The program has successfully run and has calculated the angle of the model. It's able to detect the joints accurately and calculated the angle and make the user aware inaccurate range of motion and concise feedback in real-time that helps them for improvement in their workouts.

6. APPLICATIONS

Based on human pose estimation, we can build a variety of applications, like fitness or yoga trackers. As an example, we present left bicep curl, plank and push up counters, which can automatically count user statistics and verify the quality of exercises performed. Such use cases can be implemented either using an additional classifier network or even with a simple joint pairwise distance lookup algorithm, which matches the closest pose in normalized pose space.

7. CONCLUSIONS

We focused specifically on applications in areas of human development and performance optimization. Human pose estimation is an important problem computer vision which is being able to track a person's every small movement and do a bio-mechanical analysis in real-time. This helps the gym practitioner to learn self without assistance of a coach.

REFERENCES

[1]. Sun, K., Xiao, B., Liu, D., & Wang, J. (2019). Deep high-resolution representation learning for human pose estimation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 5693-5703).

[2]. Dang, Q., Yin, J., Wang, B., & Zheng, W. (2019). Deep learning based 2d human pose estimation: A survey. Tsinghua Science and Technology, 24(6), 663-676.

[3]. Khan, S., Javed, O., Rasheed, Z., & Shah, M. (2001, July). Human tracking in multiple cameras. In Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001 (Vol. 1, pp. 331-336). IEEE.

[4]. Newell, A., Yang, K., & Deng, J. (2016, October). Stacked hourglass networks for human pose estimation. In European conference on computer vision (pp. 483-499). Springer, Cham.

[5]. Johnson, S., & Everingham, M. (2011, June). Learning effective human pose estimation from inaccurate annotation. In CVPR 2011 (pp. 1465-1472). IEEE.

[6]. Sapp, B., & Taskar, B. (2013). Modec: Multimodal decomposable models for human pose estimation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3674-3681).

[7]. Zhang, F., Bazarevsky, V., Vakunov, A., Tkachenka, A., Sung, G., Chang, C. L., & Grundmann, M. (2020). Mediapipe hands: On-device real-time hand tracking. arXiv preprint arXiv:2006.10214.

[8]. Sountharrajan, S., Nivashini, M., Shandilya, S. K., Suganya, E., Bazila Banu, A., & Karthiga, M. (2020). Dynamic recognition of phishing URLs using deep learning techniques. In Advances in cyber security analytics and decision systems (pp. 27-56). Springer, Cham.

[9]. Ramakrishna, V., Munoz, D., Hebert, M., Andrew Bagnell, J., & Sheikh, Y. (2014, September). Pose machines: Articulated pose estimation via inference machines. In European Conference on Computer Vision (pp. 33-47). Springer, Cham.

[10]. Karthiga, M., Sountharrajan, S., Nandhini, S. S., & Sathis Kumar, B. (2020, May). Machine Learning Based Diagnosis of Alzheimer's Disease. In International Conference on Image Processing and Capsule Networks (pp. 607-619). Springer, Cham.

[11]. Sountharrajan, S., Karthiga, M., Suganya, E., & Rajan, C. (2017). Automatic classification on bio medical prognosisof invasive breast cancer. Asian Pacific Journal of Cancer Prevention: APJCP, 18(9), 2541.

[12]. Sun, M., & Savarese, S. (2011, November). Articulated part-based model for joint object detection and pose estimation. In 2011 International Conference on Computer Vision (pp. 723-730). IEEE.

[13]. Karthiga, M., Sankarananth, S., Sountharrajan, S., Kumar, B. S., & Nandhini, S. S. (2021). Challenges and opportunities of big data integration in patient-centric healthcare analytics using mobile networks. Demystifying Big Data, Machine Learning, and Deep Learning for Healthcare Analytics, 85-108.

[14]. Tharsanee, R. M., Soundariya, R. S., Kumar, A. S., Karthiga, M., & Sountharrajan, S. (2021). Deep convolutional neural network–based image classification for COVID-19 diagnosis. In Data Science for COVID-19 (pp. 117-145). Academic Press.

[15]. Newell, A., Yang, K., & Deng, J. (2016, October). Stacked hourglass networks for human pose estimation. In European conference on computer vision (pp. 483-499). Springer, Cham.