International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 05 | May - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

Multiview Human Pose Estimation and Corrective Feedback

Dr. Ugranada Channabasava
Associate Professor
Don Bosco Institute of Technology, Bengaluru, India
Pin – 560074
E-mail - channasan11@gmail.com

Nishkal S Bhooshetty
Department of Information Science and Engineering
Don Bosco Institute of Technology, Bengaluru, India
Pin - 560074
E-mail - nishkalbhooshetty14@gmail.com

Nagesh C
Department of Information Science and Engineering
Don Bosco Institute of Technology, Bengaluru, India
Pin – 560074
E-mail – nagesh.chennakeshava@gmail.com

Prajwal K K
Department of Information Science and Engineering
Don Bosco Institute of Technology, Bengaluru, India
Pin - 560074
E-mail – kkprajwal3@gmail.com

Nishanth R Gowda
Department of Information Science and Engineering
Don Bosco Institute of Technology, Bengaluru, India
Pin - 560074
E-mail – nishanthgowda268@gmail.com

ABSTRACT: *Exercise posture is a basic essential activity in people's spare time. Proper and correct exercise pose can not only help people to get a healthy body, but also help to reduce pressure and relax their mood. So, for practicing exercise a trainer is important who can guide and monitor the perfectness of different exercise poses. However, the wrong way of exercising not only makes the exercise fall short of expectations, but also causes muscle damage. Therefore, we have developed a set of human exercise posture analysis and guidance software to analyze the user's exercise posture through the video captured by the camera. This software chooses OpenPose as the basic pose estimation network, and two methods are proposed to improve the model effect. The software firstly conducts the human body posture detection, obtains the coordinate information of key points, then carries on the exercise posture analysis, and gives relevant suggestions according to the definition of the correct exercise form. Integrating computer vision techniques, the proposed system analyses the user's human pose then based on the main knowledge of exercise, the user is guided to correct the pose. Precise recognition of exercise pose is a difficult task because of high computation and lack of availability of dataset. For the accurate detection of the exercise pose, different feature extraction and preprocessing methods are applied to the dataset which results in 97.4% accuracy just by using machine learning algorithms.*

INTRODUCTION: In daily life, physical exercise has become an essential fitness program, which is very beneficial to the physical and mental health, but the wrong way of exercise can cause to get injured while doing exercises, and professional fitness instructor courses are expensive, not suitable for the common public [1]. At the same time, the traditional exercise posture analysis equipment of wearable devices is very accurate in information collection, but it is inconvenient to wear and is only applicable to the coaching of professional athletes. In previous years, with the evolution of computer vision and deep learning, this has become achievable for users to analyse the posture by camera. Therefore, we designed a software to detect the user's movement in the video and conduct exercise analysis and guidance. In our software, users can perform exercise facing the camera. The system of this application can recognize and correct the accuracy of poses from fitness practitioners by using input in the guise of videos that record the movements of fitness practitioners continuously. Then we adopt a bottom-up pose detection method-- OpenPose, which uses two branches of CNN[2]. Compared with other top-down methods, it is not only simple besides it also has high robustness. At the same time, the recognition value of the indigenous method in the special scene is not high, which is enhanced and improved, so that the method is more widely applicable to the scene. Then, we analyzed the detected key points, calculated the exercise information of the limbs in each frame, carried out statistical analysis in the time domain, and compared with the defined standard form of exercise, so as to determine whether the exercise mode is correct, and carried out targeted problem correction. This software also uses the support vector machine (SVM) method to classify or recognize poses from the users. Once the pose is recognized, the system evaluates the accuracy of the movement on the basis of angular geometry of each joint. The output of this system is to display the type of movement and advice on the monitor regarding the precision of the movement or pose of the exercise performed. We achieve this by including top-down feedback – instead of trying to directly forecast the target outputs, as in feedforward processing, we forecast what is wrong with their current estimate and correct it repeatedly. We call our framework Iterative Error Feedback, or IEF.

ARCHITECTURE OF SOFTWARE: The software consists of four modules: input, pose estimation, regularization, and posture analysis. The basic frame diagram of the software is shown in Figure 1. The following is a brief description of these four modules.

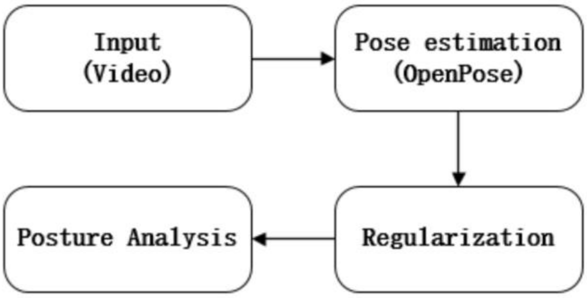


Figure 1. The Architecture of the software

A. Input:

Users need to record exercise videos from a particular perspective. The software has no requirements on the video acquisition equipment, but it needs to ensure that the exercise is clearly visible.

B. Pose Estimation:

The basic model we chose is OpenPose, a model composed of multi-stage CNN with two branches, which is a bottom-up model with PAFs algorithm as its core idea. OpenPose is the first algorithm to apply deep neural network to pose estimation. It is not only adequate, additionally it is also very accurate. At the exact time, its convenient installation and operation is also amongst the reasons why we choose it as the basic model of this design.

C. Regularization:

Since various users have different height and body type, such as adults and children have great differences, in the intrest to solve the similarity error caused by body type, we will regularize the data.

D. Posture:

Analysis Through the key point detection conducted by OpenPose, we will obtain the human body architecture diagram consisting of 18 key points and the coordinate information of each key node in each frame. The software will analyze the coordinate sequence of each key point and calculate the key information such as the angle of key body parts.

LITERATURE SURVEY:

A. Human Pose Estimation:

A computer vision-based tool called "Human Pose Estimation" finds and evaluates human posture. The modeling of the human body is the primary aspect of human pose estimation. The procedure typically entails the removal of human body joints, followed by the study of human positions using deep learning algorithms. When video footage is used as the data source for the human pose estimate system, critical points (joint positions) are identified from a series of frames rather than a single image. Due to the system's analysis of a person's actual movement rather than a fixed position, we may now attain more accuracy [3].

B. OpenPose:

The first real-time system, OpenPose, can identify 135 important points in a single image, including body, hand, face, and foot key points.

Human re-identification, retargeting, and Human-Computer Interaction are just a few examples of the many research subjects involving human analysis that now apply the use of the OpenPose library. Additionally, OpenPose has been applied to the OpenCV library [4].

C. Skeleton Detection:

Skeleton detection is the processing of data from a two-dimensional camera to define the location of the joints in the human body and the skeleton-like shape of the human body. Skeleton detection on two-dimensional cameras can be done using the OpenPose library on OpenCV. OpenPose can recognize the shape of the human body as a whole and can distinguish between human objects and non-human objects. In addition, OpenPose can also identify the pose of the human body, face, and hands. For identification of human body poses, OpenPose can identify based on 18 points body parts of the head, neck, shoulders, elbows, wrists, hips, and feet. Figure. 2 shows the COCO body format index.

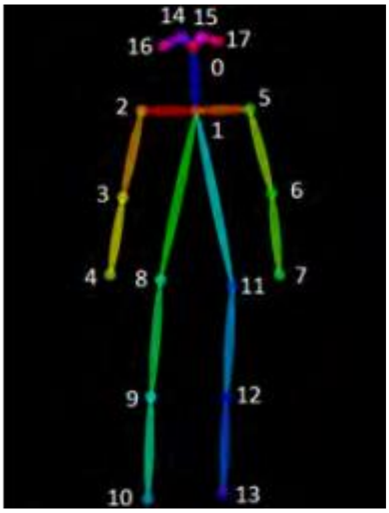


Figure 2. COCO Body Format

D. Skeletal Angle:

Skeletal angle or skeletal angle has a direct relationship to human activities. Angles at the shoulder, elbow, hip and inner knee provide useful and informative features. To get the angle value information, a calculation is carried out using the x and y coordinate values that have been obtained in the skeleton detection. Figure 3. and Eq. (1) show how to get the skeletal angle values using the inverse cosine equation.

$$angle\_p1 = cos^{-1}\left(\frac{a^2+b^2-c^2}{2ab}\right) \times \frac{180}{\pi} \qquad (1)$$

On Eq. (1), there are p1 which is the point of the angle of the frame being searched for, p0 and p2 are the angles flanking the skeletal angle. Then there are a, b, and c which are the distances between each angle that can be calculated using the Euclidian distance equation in the Eq. (2).

$$a = \sqrt{(p1(x) - p0(x))^2 + (p1(y) - p0(y))^2}$$
$$b = \sqrt{(p1(x) - p2(x))^2 + (p1(y) - p2(y))^2} \qquad (2)$$
$$c = \sqrt{(p2(x) - p0(x))^2 + (p2(y) - p0(y))^2}$$

E. Real Coordinate Conversion:

Real coordinate conversion is done so that the data obtained becomes more centered. Coordinate data that was previously in pixels will be converted into meters. The center point or zero of the data will also be changed from being at the top left of the frame to the middle of the frame. Eq. (3) shows how to convert pixel coordinates to real coordinates in meters for x and y axis.

$$pos\_x = \left(x - \frac{width}{2}\right) \times z \times SCL$$
$$pos\_y = \left(y - \frac{height}{2}\right) \times z \times SCL \qquad (3)$$

The actual coordinates that will be calculated are pos_x and pos_y. The coordinates in pixels are x and y. Z is the human object depth where in this research it is determined to be 2.4 meters. SCL is a comparison scale. Also, there are width to calculate x axis and height for y axis.

F. Support Vector Machine :

When classifying or regressing a collection of data, the Support Vector Machine (SVM) method can be utilized. SVM can perform linear classification and has developed to be able to solve non-linear problems by finding the maximum distance between data classes. SVM automatically identifies informative point subnets known as support vectors and uses them to represent a separating hyperplane which is a linear combination of points.

IMPLEMENTATION: By using OpenPose lightweight for CPU human pose estimation was done and the location of 18 keypoints was extracted. For the given set of poses, the steps [6] to do a certain pose were extracted and the subsequent rules for those particular steps were mentioned. For each step, the user is validated against the rule of that particular step and once the rules are met true then only the practitioner is allowed to further proceed to the next step. In this way, a systematic procedure to do the correct pose is achieved.

For the convenience of the user, for each pose, the reference image for the step is shown on the screen along with the real-time video feed of the user, as well as the correction text is also displayed on the screen over the video, as shown in Figure 3. The reference image is the snapshot of the correct pose that the user should do, however there is an threshold set with allows people of different body type and varied experience to practice the yoga correctly.

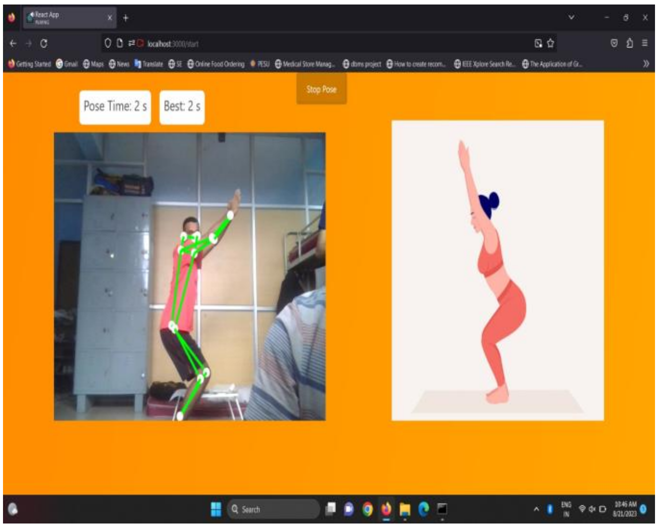


Figure 3. Snapshot of System for Chair Pose

To classify the given yoga pose, classification machine learning models were used, namely Random Forests and SVM. The extracted data of values of 18 keypoints is converted into angles of each joint and distance between the certain joint to make it easy for the algorithm to generate a factual pattern. The data after processing and extracting the features is stored in a .CSV file and then the data is split into training and testing set in a ratio of 75:25.

A. Camera Access

At this stage, camera access is carried out using the OpenCV library. This process aims to get the image frame from the video captured by the camera in real-time. To access the camera on OpenCV, the first is to load or import the OpenCV library. Then define a video capture object which means the index of the camera used. Then the camera capture results are displayed in the frame repeatedly or looping so that the obtained image can change in real-time.

B. Skeleton Detection

After the image data is obtained by the camera, the skeleton detection process is carried out using the OpenPose library on OpenCV. To perform skeleton detection, the first step is to load the OpenPose library. In the looping process, the input function calls to OpenPose with OpenCV object capture parameters. Then the output of OpenPose, which is an image of a human skeleton, is displayed on the frame.

C. Skeleton Keypoints

During this stage, joint coordinate data is collected from the skeleton detection results. The data obtained in the form of 18 data coordinates x and y in units of pixels.

D. Skeletal Angle

From the joint coordinate data obtained, calculations were also executed to obtain the skeletal angle value. Skeletal angle values obtained from calculations using the Euclidian equation and inverse cosine. The Euclidian equation is used to obtain the distance between the angles that are flanking each other, then the inverse cosine is used to obtain the angle value based upon the input from the Euclidian calculation.

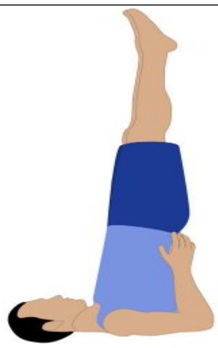
E. Real Coordinate Conversion

After getting all the coordinates of the joint skeleton (key points), the next step is to do a conversion to change the x and y coordinates of the pixel units into real coordinates (meters) so that the data collected is more centered. Coordinate data that was previously in pixels will be converted into meters. The center point or zero of the data will also be changed from being at the top left of the frame to the middle of the frame.

F. Creating Datasets

The dataset was made by collecting pictures of animated object poses doing fitness movements. The dataset is made with animated objects that are around 1-2 meters away from the camera. For the provisions of the human position, namely facing the camera, then tilted left and right, finally facing left and right. The body posture of human objects is limited to a minimum height of 150 cm and a maximum of 180 cm.

Table 1. Example of Image Dataset

 Tree Pose	 Chair Pose
 Cobra Pose	 Warrior Pose
 Dog Pose	 Shoulderstand Pose
 Triangle Pose	

G. Data Training

The data training was executed out using Google Colab with the prepared dataset. There are 4 types of data used, namely Tree Pose, Chair Pose, Cobra Pose, Warrior Pose, Dog Pose, Shoulderstand Pose and Triangle Pose. The training is executed out using the SVM method using the Scikit-learn library.

G. Movement Correction

The fitness movement correction process is carried out using geometric evaluation. This process is carried out by determining the value of the main joint angle for each movement. Then this value is compared with the main joint value obtained directly. From the comparison results, an error value will be obtained which can again be used as a correction system for whether the movement is correct or not.

EXPERIMENTAL RESULTS: The proposed Exercise Pose Validation system made implemented in Python Programming Language with OpenCV 4.5.1 and Lightweight OpenPose [7] (PyTorch 1.7.1 backend) and runs on Lenovo Ideapad 330 (Intel Core i3-6006U CPU @2.00GHz, 4GB RAM, Windows 10 64-bit OS). The system was able to correct the user for the given set of poses. It was tested on the different users for each defined pose and the system was successfully able to guide the practitioner for the mistakes.

The data collection, image dataset generation, and training of the model were done on Google Colab. Data preprocessing and feature extraction were done on PyCharm (Python 3.7.9) on Windows 10 64-bit OS. The designed system is cost effective because it uses computer vison technique which is not expensive alike dedicated sensors, moreover it can also work on a CPU system with just a camera to perform, it doesn't need any GPU for processing which make it computationally efficient too[10]. So it doesn't use any dedicated sensors or a high-end device for its functioning, which cuts a lot of operating cost and can be accessible to a large group of people. The accuracy of the SVM classification model was 97.64% while the accuracy of the random forests classification model was 96.47% when the model was tested against the training set.[11]

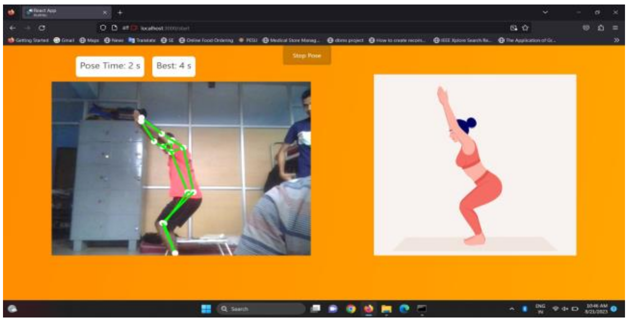


Figure 4(a). True Prediction of the Chair Pose

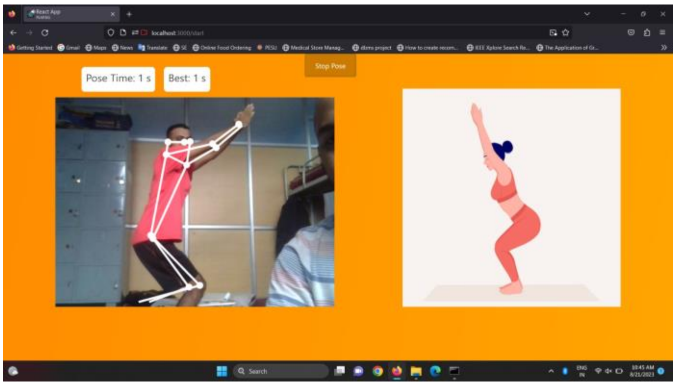


Figure 4(b). False Prediction of the Chair Pose

The result of the system made is that it can detect and recognize fitness movements based on output images. Figure 4(a) and 4(b) shows the example of classification system results. Figure 4(a) shows the True Prediction of the Chair Pose and Figure 4(b) shows the False Prediction of the Chair Pose.

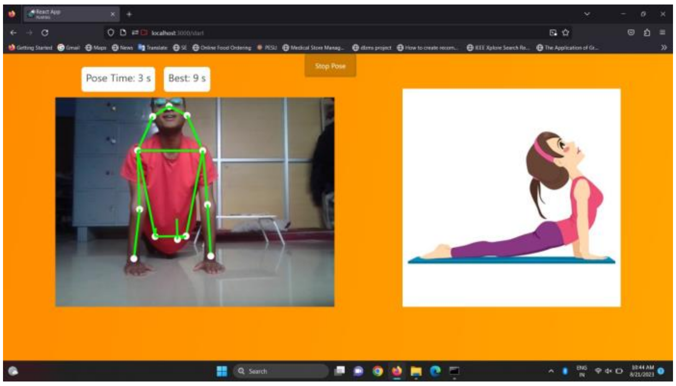


Figure 5(a). Front View of the Cobra Pose

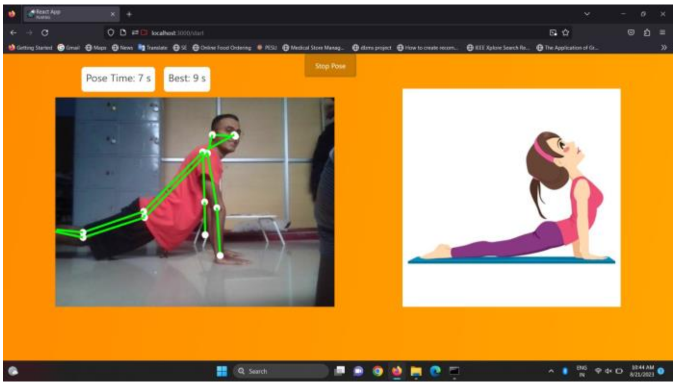


Figure 5(b). Side View of the Cobra Pose

The Figure 5. shows the result of the system which recognizes the fitness movement of Cobra Pose performed by the user. The system can recognizes both front view and the side view of the Pose.Figure 5(a) shows the front view and the Figure 5(b) shows the side view of the Cobra Pose.

The training method for classification of movement types is SVM by utilizing the Sci-kit library. This method was chosen because there is research that makes a system that can determine the type of stroke in table tennis and SVM is the best method because it produces accuracy with low variance. There are also works that perform posture analysis to classify activities carried out using SVM [8][9].

CONCLUSION:

The Proposed System holds significant promise in revolutionizing various domains, from fitness and sports training to healthcare and rehabilitation. Through the development of advanced algorithms and technologies, this project has the potential to profoundly impact human well-being by enhancing movement quality, reducing injury risk, and optimizing performance.

By leveraging multi-modal data fusion techniques, real-time feedback systems, and adaptive learning models, we can create personalized coaching experiences that cater to individual differences in body morphology, movement patterns, and skill levels. These systems provide immediate corrective guidance during physical activities and also enable longitudinal tracking and progress monitoring, fostering continuous improvement and engagement over time.

Furthermore, by integrating context-aware pose detection with immersive technologies such as virtual reality and augmented reality, we can create interactive training environments that enhance user motivation and enjoyment. These platforms facilitate collaborative learning and knowledge sharing among users, fostering a supportive community environment that empowers individuals to achieve their movement goals effectively and safely.

As we continue to innovate and explore future directions such as semantic understanding of poses, privacy-preserving techniques, and collaborative learning approaches, the potential of this project to positively impact human health, and well-being will only grow. By embracing these opportunities and addressing challenges with creativity and diligence, we can unlock the full potential of machine learning in human pose detection and corrective feedback, shaping a future where optimal movement is accessible to all.

REFERENCES:

[1] Takeichi K , Ichikawa M , Shinayama R , et al. A Mobile Application for Running Form Analysis Based On Pose Estimation Technique[C] 2018 IEEE International Conference on Multimedia & Expo Workshops (ICMEW). IEEE, 2018.

[2] Hou Y , Yao H , Li H , et al. Dancing like a superstar: Action guidance based on pose estimation and conditional pose alignment[C] 2017 IEEE International Conference on Image Processing (ICIP). IEEE, 2017.

[3] Sigal, Leonid. “Human pose estimation”. Computer Vision: A Reference Guide. Cham: Springer International Publishing. pp. 573- 592. 2021.

[4] Cao, Zhe, et al. “Realtime multi-person 2d pose estimation using part affinity fields”. Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

[5] İnce, Ömer Faruk, et al. “Human activity recognition with analysis of angles between skeletal joints using a sensor”. ETRI journal. Vol. 42, No. 1: pp. 78-89. 2020.

[6] "yogajournal," Pocket Outdoor Media Inc., [Online]. Available: https://www.yogajournal.com/.

[7] D. Osokin, "Real-time 2D Multi-Person Pose Estimation on CPU: Lightweight OpenPose," CPRAM 2019 - Proceedings of the 8th International Conference on Pattern Recognition Applications and Methods, pp. 744-748, 2019.

[8] Kulkarni, Kaustubh Milind, and Sucheth Shenoy. “Table tennis stroke recognition using two-dimensional human pose estimation”. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021.

[9] Gaglio, Salvatore, Giuseppe Lo Re, and Marco Morana. “Human activity recognition process using 3-D posture data”. IEEE Transactions on Human-Machine Systems. Vol. 45, No. 5: pp. 586- 597. 2014.

[10] V. Ferrari, M. Marin-Jimenez, and A. Zisserman, “Progressive search space reduction for human pose estimation,” pp. 1–8, June 2008.

[11] T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. L. Zitnick, “Microsoft COCO: ´ common objects in context,” CoRR, vol. abs/1405.0312, 2014.