ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 9, Issue 11, November 2022

Intelligent Gym Trainer Supporting Pose Correction using PoseNet and YoloV4

[1] Sridhar Raj S, [2]Pratishtha Gaur, [3]Aryan Patel

[1][2][3] Vellore Institute of Technology, Vellore, India

Email: [1]sridharselva394@gmail.com,[2] pratishthagaur03@gmail.com, [3]aryanpatel1852@gmail.com

Abstract*— In the busy schedule people have today, they cannot manage to go to the gym because of their rigid timings also not everybody can afford a personal trainer. For neophytes who are just beginning to start exercising, it is very important to have a correct posture which if not cared about can cause serious health issues. In this paper, we propose an intelligent gym guide, a web app available for every individual to train themselves using AI. This app will list exercises based on the equipment we have at our homes and the same app will also guide us through it. We propose to use the YoloV4 model trained over gym equipment available at an individual's house. The application will take a webcam feed as input to perform the given task and give live results and feedback. It will incorporate machine learning algorithms for object detection and a PoseNet model for pose correction while exercising; these models are further discussed in detail in the paper.

***Keywords*—Mediapipe, Object Detection, PoseNet, Pose Estimation, Yolo**

I. INTRODUCTION

Correct posture reduces strain on the human body by maintaining a balance of the musculoskeletal structure [1]. This balanced state protects the supporting structures in the body and prevents damage or gradual deformation in all positions. Additionally, correct posture implies not having an inclined or bent body and spine structures. Therefore, the importance of correct posture should be emphasized, and maintaining correct posture encouraged. To get the results you want from a workout, keeping your body in an appropriate position is just as important as the workout itself. People exercising without a trainer conveniently forget its importance. For better health of the community appropriate pose while exercising has to be ensured. According to the concerned problem of the application, an apt selection of algorithms and models is necessary for easy implementation and the most effective result. The algorithms used here should be able to execute the task of object detection and classification of the given equipment with the highest accuracy possible such that the user can get the required info even under low lighting and variable distances. For this purpose, many algorithms exist that classify and detect the images. A few of the most commonly used models are Single-Shot Detector, Yolo, RCNN, Detectron, and Tensorflow. Other than object detection, the model should also be able to detect the position and posture of the user's body through the camera in real-time such that the app can guide them through the process and correct the mistakes that they might be doing. For this purpose, pose detection is performed by an algorithm that uses and implements the concept of deep learning and Neural Networks. A few of the common human pose algorithms include PoseNet, BlazePose, OpenPose, AlphaPose and HRNet. These models take input in the form of an image and or a video, either pre-saved or in live time, and detect the joints of the human body such as the wrists, shoulders, knees, ankles etc. and other major joints. Most of the existing models are based on a 17-joints implement that can detect thirty-three joints' movement in the human body and can even follow them in live time. Those detected points are then joined to give a skeletal figure of the pose in consideration.

The paper proposes an intelligent gym trainer with both these features of object detection and posture correction to promote a correct method of exercising after proper study of algorithms and methodology.

In section 2 the research domain knowledge is explained. In section 3 proposed methodology of the application after review has been explained. Sections 4 and 5 summarize the results and conclusion.

II. LITERATURE REVIEW

A. Existing Datasets and Model for Pose Correction

There are a few models that were referred to create the proposed pose trainer. Researchers after Implementing technologies like CNNs, Part affinity fields, DTW (Dynamic Time Warping), and Stacked hourglass architecture have tested the models on reputed and large datasets giving improvements year after year. Some of these are MPII (25k annotated images), LSP (2k annotated sports images), FLIC (5k images from movies), COCO (164k images split into test and train), Human3.6M (3.6 million motion images), MoCap CMU.

B. Traditional CNN methods

The first paper talks about an application that uses a single, complete image to automatically estimate a human body's 3D position and shape [2]. They first predicted the locations of the 2D body joints using a CNN-based method called DeepCut, after which they fitted an SMPL statistical model

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 9, Issue 11, November 2022

for body shape to the 2D joints. Since SMPL records correlations in human shape across the dataset and can be successfully fitted to relatively little data, they were able to achieve their goal by minimizing an objective function that penalizes the error between the visible 3D model joints and identified 2D joints. Another method for human pose estimation is based on Deep Neural Networks (DNNs) [4]. In order to develop the pose estimate with respect to body joints, a DNN-based regression problem was used. In [3] a technique for multi-person identification and 2-D posture estimation challenge was covered. Divided into two phases, it makes up a straightforward yet effective methodology. The first stage employs the Faster RCNN detector to predict the position and size of boxes that are likely to contain humans. The main points of the individual who might be contained in each proposed bounding box are estimated later in the second stage. It uses a fully convolutional ResNet to predict dense heatmaps and offsets for each key-point type.

C. Newer methods

2D pose estimation [5], this approach proposes to detect 2D pose of more than one person in an image effectively. To learn to link body parts with persons in the image, they employed a non-parametric representation known as Part Affinity Fields (PAFs). No matter how many people are in the image, the architecture encodes global context, enabling a greedy bottom-up parsing phase that achieves real-time performance while maintaining excellent accuracy.

Paper [6] describes a convolutional network architecture for pose estimation. It forms a chain of a bottom-up followed by a top-down processing unit with intermediate supervision to improve the performance of the network. This architecture is referred to as a stacked hourglass.[7] discusses an approach to predict the 3D position of joints from a one-bit image, without any temporal feature. It creates subdivisions of parts of the image to represent pose estimation into a per-pixel classification task. In structured prediction applications like articulated pose estimation, the convolutional pose machine implicitly models long-range dependencies between variables [8]. To do this, a sequential architecture made of convolutional networks that directly operate on belief maps from earlier stages is created. This eliminates the need for explicit graphical model-style inference and results in increasingly accurate estimates for part positions. The algorithm proposed in the paper Joint 3D motion capture introduces a statistical framework for quick and reliable 3D motion analysis from 2D video data by integrating monocular 3D pose estimation with physics-based modeling [9]. To learn 3D motion coefficients and combine them with physical parameters that explain the dynamics of a mass-spring model, it employs a factorization approach. The method only uses one camera, requires no further force measurement, nor torque optimization, and allows for the estimation of invisible torques in the human body. Paper [10] suggests an application that recognizes a user's workout posture and offers individualized, thorough advice on how the user might improve their form. Pose Trainer recognizes a user's pose using the most recent advances in pose estimation, then performs an exercise to assess the pose's vector geometry and offer helpful feedback. Paper [11] relies on the most recent developments in human body position estimation utilizing deep learning. It synchronizes user/reference videos using time series data alignment methods like Dynamic Time Warping (DTW) and optical flow tracking. Based on a threshold divergence between the limb angles, it is particularly good in identifying and locating mistakes in a user's activity (position). Future upgrades to the technology will allow doctors to track a patient's healing from an injury. Table 1 summarizes on the datasets and how well the result was based on various metrics, F1 score, mAP, and more. Can decide upon which dataset to use based on this analysis.

Table 1:

Paper no.	Metrics	Metric value	*Dataset*
[1]	Mean joint errors in mm	82.3	Human3.6M
[2]	mAP% (Mean average precision)	79.7%	MPII
[3]	Total PCKh	90.9%	MPII
[4]	AP (average precision)	0.685	COCO test-dev
[5]	mAP	0.731(inf erred body parts)	COCO test-dev
[6]	PCP (Percent correct parts)	0.61 0.69	LSP Image Parse
[7]	PCKh0.5 PCK PCK@0. 2	88.52% 90.5% 95.03%	LSP MPII FLIC
[9]	F1 score	0.85 1 0.85 0.73	Bicep curl Front raise Shoulder shrug Shoulder Press

Summary of existing models for posture correction obtained from the analysis performed from recent research papers.

D. Object Detection, Classification, Recognition Algorithm Selection

2D For our application, we aim to classify gym equipment and map them with appropriate exercises available for it. The analysis states that the process of rendering images in the appropriate format is a struggle. Three branches of image

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 9, Issue 11, November 2022

classification, detection and recognition seem similar but produce varying result formats. The main aim of image classification is to identify features in an image whereas detection is used to determine which category each object belongs to. Image recognition is a combination of both these techniques, process involves gathering and organizing data, building a computing model then using it to recognize images like face recognition to unlock a mobile app that will first detect face organize features of inbuilt data and recognize the face [12],[13].

The most suitable algorithm for classifying gym equipment in this scenario seems to be the object detection technique, considering that several types of equipment can be present in a picture and in any combination which separately will be needed to be identified.

E. Object Detection Model Selection

Adding to the widely used object detection methods, we categorize them broadly into three groups. Single Shot MultiBox Detector (SSD), You Only Look Once (YOLO), and Region Proposal. The first stage of the R-CNN model, which consists of two stages, is solely responsible for finding an object in the scene and providing an initial estimate of the bounding box for the object. Then, in the second stage, it decides the class and fine-tunes the bounding box by using local features in the suggested region [14]. As opposed to single-stage detectors like YOLO or SSD, which use dense sampling and a fully convolutional approach in a single shot to determine whether an object is present or not, what the class probability is in the event that it is, and to regress out bounding box coordinates, multistage detectors perform these tasks in stages. On the output side of single-stage approaches, you have grid cells that loosely correspond to various regions of the input image. Each grid cell is connected to a number of anchor boxes, each of which is attempting to predict the object probability, the conditional class probability, and bounding box coordinate regressions for any objects whose centers fall within the grid cell. From the standpoint of training, one could envision that most grid cells and anchors would be exposed to background or no-object scenarios, while a select few would be notified of the presence of a ground-truth target within the grid cell. In the end, a single-stage detector must choose between the abundance of background or no-object samples against target examples. In the past, this was one of the primary causes of lower accuracy/mAP for single-stage detectors in comparison to systems like R-CNN and its derivatives, which utilize a two-stage approach and can handle this better in the first stage [15]. The second major distinction between 2-stage detectors, in general, is that the second stage, which is in charge of more insight, can concentrate on features isolated to the region of interest where the region proposal is. This can improve the ability to classify, detect objects more reliably, and be less confused by background elements in other regions of the image. Any suitable complexity single-stage fully convolutional network will have a very vast receptive field, and the detections are ultimately influenced by this less concentrated/specialized view of features across a wider swath of the image. Custom 2-stage detectors can be effective, especially when there is a shortage of training data, but they are slower to infer conclusions than a single-shot, fully convolutional object identification method [15][16]. Since the program uses a live stream and the problem is simple equipment detection, timing is more important than accuracy in this situation. Gym equipment is quite recognizable, so there won't be much uncertainty when it comes to precision. Additionally, using a sophisticated method of region suggestions like RCNN is not necessary for a straightforward application like this. For YOLO, detection is a simple regression problem that learns the class possibilities using bounding box coordinates from an input image.

Every image is divided into a S x S grid by YOLO, and each grid predicts N bounding boxes and confidence. The bounding box's accuracy and whether or not it contains an object notwithstanding the defined class are reflected in the confidence.

Even the classification score for each box in each class is predicted by YOLO. To calculate the likelihood of each class showing up in a predicted box, combine the two classes. Swiftness and precision are given equal weight by the Single Shot Detector (SSD). SSD creates a feature map by running a neural network on the input images just once. On this feature map, we are currently running a modest 3 X 3 sized convolutional kernel to predict the bounding boxes and

Table 2:

Models	Backdrop	AP50 Value
Region Proposal (Faster RCNN)	Too complex, concerned with the application detecting and classifying Gym equipment	59.2
Yolo (yolov4)	Unable to identify very small objects, but is very fast in detection concerning live streaming	62.8
SSD (VGG 16)	Gives equal weightage to speed and accuracy but the more important parameter in this application is speed rather than accuracy since gym equipment are fairly distinct.	52.9

Summary of object detection models obtained from the analysis performed for the three most common systems.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 9, Issue 11, November 2022

categorization probability. However, since we need to move very quickly yet accuracy doesn't cause too much concern, using YOLO would be the best course of action [17][18]. Table 2 above summarizes all these models and their issues along with the accuracy they generated.

F. Yolo Version Selection

Taking the example of asphalt pavement cracks detection, The YOLOv4 and YOLOv5 models still achieve good training outcomes even with a small dataset when we compare them to the YOLOv3 model [19]. The detection speed of YOLOv4-tiny is 10.16 frames per second, which is more than twice as fast as YOLOv3-tiny based on a middling CPU. The maximum mAP value for the YOLOv5 models is 94.39 percent, while the values for the YOLOv3 models are both less than 80 percent. The mAP values of the YOLOv5 models all surpass 85 percent. Although the evaluation indices of the YOLOv4 models are inferior to the evaluation indexes of the YOLOv5 models, the YOLOv4 models have better detection performance than the YOLOv5 models. While YOLOv5 models occasionally identify and discover pseudocracks that may be related to the backbone's design elements, YOLOv4 models are more robust for hidden cracks.

This crack analysis explains why Yolov4 is significantly more effective than other YOLO models when focusing on small images, which is one of their weaknesses. It is usually appropriate to take into account smaller objects because it is not possible to specify how much screen space equipment should occupy during the equipment detection process. Thus, YOLOV4 was concluded to be better and was the most suitable for our requirements.

G. Comparison of pose estimation models

We will be tracking and estimating the pose of the human body while performing exercises such that it can be made sure that all the postures are being performed in the correct form and that only the intended muscles and body parts are getting acted upon. For this, we explored pose estimation models such as PoseNet, OpenPose and BlazePose. Basically, these models detect major joints of the body and connect them to give a basic stick skeletal figure that can be tracked in live time.

The algorithm Realtime Multi-Person Pose Estimation proposed by [20] offers a real-time method for identifying the 2D poses of multiple peeple in an image. The proposed method learns to link body parts with persons in the image using a non-parametric representation that we call Part Affinity Fields (PAFs). No matter how many people are in the image, our bottom-up approach delivers great accuracy and real-time performance. PAF-only refinement, as opposed to PAF and body part location refinement, is used by OpenPose, which significantly improves runtime performance and accuracy. Keypoint heatmaps and their pairwise relations (part affinity fields, PAFs) are provided by OpenPose using the inference of neural networks [21]. The output is eight times downsampled. Then it extracts important points from the heatmap peaks and groups by their instances after upsampling the tensors to the original image size. The network extracts features first, then computes initial estimates of heatmaps and PAFs, followed by those five phases of refining. It can locate 18 different kinds of critical spots. From the preset list of keypoint pairs, the grouping algorithm then searches for the best pair (by affinity) for each keypoint. Pose Trainer is a programme that recognises the user's exercise pose and offers specific, in-depth suggestions on how to improve form. Its dataset includes more than 100 distinct workouts [27]. Here, part affinity fields—vectors that encode the position and orientation of limbs—are used by OpenPose, which presents a novel method of pose estimation. The model is made up of a multi-stage CNN with two branches, one to learn the part affinity fields and the other to learn the confidence mapping of a key point on an image. Since OpenPose is available in executable Windows and Linux formats, no installation or programming experience is necessary to utilise it. Additionally, GPU libraries don't need to be installed outside. BlazePose uses a detector-tracker configuration that consists of a pose tracker network and a lightweight body pose detector, and it adopts the strategy of creating heatmaps for each joint and fine-tuning offsets for each position [22]. The tracker predicts keypoint positions, a person's presence in the current frame, and a fine-grained region of interest. Unlike the majority of contemporary object detection solutions that rely on the Non-Maximum Suppression (NMS) algorithm and are unable to detect ambiguous boxes, BlazePose tackles this issue by detecting the bounding box of a relatively rigid body part, such as the human face or torso. Pose estimation anticipates a person's primary body parts in each frame of a video feed [26]. Pose correction, on the other hand, involves defining the angles based on the key points and then carrying out pose correction for wrong angles. In order to regress the 6-DOF camera Dynamic Time Warping (DTW), a single RGB image is used as the starting point for the network, which was then used to calculate the keypoint angle without the need for any correction. 33 key points are produced by the BlazePose model's additional engineering or graph in place of an optimization, employing 17 key points produced by the most effective 23-layer ConvNet models. Two steps are used by Blazepose. This model adheres to the pose estimation technique's architecture: The convolutional network region with six interests for pose estimation is initially detected by the GoogleNet, a 22-layer detector, followed by "inception modules" and two trackers that follow the 33 landmarks. Additional intermediate classifiers are used depending on the kind of exercise that is abandoned during testing. Based on variables like the frequency of various stance correction motions in a frame and urgency, multiple approaches were utilised for various taxonomies/categories of activities that

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 9, Issue 11, November 2022

were being monitored.

It is frequently possible for ambiguous and overlapping points on human bodies to make false posture predictions [23]. In order to avoid this, the given convolutional network implicitly considers such cases and creates discriminators to separate genuine postures from imitations. The posture generator (G) is made to anticipate poses and occlusion heatmaps in a multi-task, stacked way. The discriminators are then given the pose and occlusion heatmaps to forecast the likelihood that the pose is real. The network is trained using conditional generative adversarial networks (GANs).

Paper [25] discusses about various new models, firstly, DeepPose model which uses an AlexNet backend to estimate poses holistically, which means that it considers pose as a whole, it can estimate even if some joints are concealed. Next is OpenPose, its design encodes a global context, facilitating a greedy bottom-up parsing phase that achieves real-time performance regardless of the number of individuals in the image while maintaining high accuracy. The runtime complexity is uncoupled from the number of persons in the image using a bottom-up method. HRNet model instead of moving from low resolution to high resolution, this keeps a high-quality representation throughout the entire process. The architecture adds high-to-low resolution sub-networks one after another, starting with a high-resolution sub-network as the initial step. PoseNet uses neural networks in a bottom-up manner to implement the idea of heatmaps. The Resnet architecture and the MobileNet architecture are the two PoseNet architectures. The MobileNet architecture has been incorporated in this model. The PoseNet model receives the user's live exercise video as input, it extracts the essential points, and records them in an array. Using the Dynamic Time Warping, this array is normalised and compared to the dataset pickle file. PoseNet produces heatmaps together with a confidence score. The offset vector, which indicates where the heatmaps are located, is the second output.

The aim in [29] is to learn various commands through different VR gestures and signs for the ease of communications with devices through Extended Virtual reality (XR).

Table 3:

Model	Method	*FPS*	*Avg.precision*
PoseNet	ResNet	30	81
OpenPose	CNN	22	79
BlazePose	MediaPipe	98	74.2

Summary of existing models for joint extraction obtained from the analysis performed from recent research papers.

Its dataset consists of 1750 images 10 for each gesture. Three non-gesture classes were also added to reduce incorrect predictions. The training dataset was expanded by flipping every image, doubling the number of them, to enhance the ML model performance. This ensures that the model is able to recognise the pose from any angle or viewpoint.

III. PROPOSED WORK

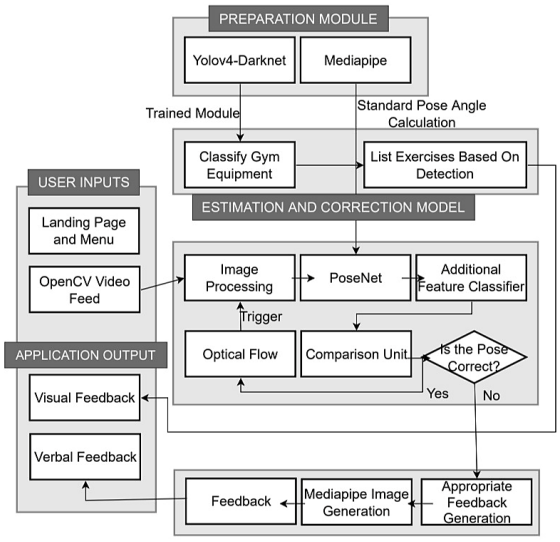
To solve the issues concerning body posture while exercising, this paper suggests an approach based on app development in order to make the process user friendly. The application follows this flow:

1. Input from Live feed
2. Select Pose correction or Object Detection
3. Object detection uses a pretrained yolov4 darknet model, to display possible exercises
4. Pose correction model builds a skeletal sketch using Mediapipe and PoseNET to extract features from it and compare
5. Gives appropriate prefeed feedback in all cases

As visible from Figure 1, the model takes in the photo/video input using OpenCV. It opens a series of picture files, a camera, or an IP video stream for video recording, and this live feed is utilised for two things: detecting any well-known gym equipment and providing exercises for it, and primarily to correct postures. The module of identifying gym equipments uses darknet for feature extraction and PANet for the feature aggregation of the network using a dense input network size thus improving accuracy. This functionality is more important than people find it. Since the application is for neophytes, they might not know what all exercise can be performed using a gym ball, there is much such equipment found in our houses which one can use to exercise.

The other section is posture correction. The input live feed in the RGB format is passed through the encoder which is then further passed through the Localizer. PoseNET [24] utilizes a Convolutional Network for Real-Time 6-DOF Camera relocalization.

Figure 1:



Architecture diagram of object detection and pose correction.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 9, Issue 11, November 2022

This monocular six degree of freedom (DOF) relocalization system is reliable in real-time. It uses an effective 23-layer deep ConvNet to train a convolutional neural network to regress the 6-DOF camera posture from a single RGB image in an end-to-end manner without the need for further engineering or graph tuning. This model is based on the GoogleNet architecture, which consists of a 22-layer convolutional network with six "inception modules" and two additional intermediate classifiers that are eliminated during testing. Heatmaps and a skeletal representation of the stance are the results. PoseNet basically plots the 17 joints' body coordinates. It continuously records human body movements and stores the coordinates the joints in order and place it in array. This is later used to calculate angles. Say plank, the angle between shoulder elbow and hand should be 90 degrees and for back shoulder hip and leg which should be 180 degrees These will be verified and feedback displayed. The output will be in both visual and verbal feedback form, in verbal, an audio feedback will be provided and in visual the angles made between joints and correct ones will be displayed at any given time.

Dataset: The dataset we utilise in this scenario is one that we created ourselves and consists of six pieces of gym equipment that were retrieved using Google's Open Images Dataset (OID) along with annotations. It was composed of 200 images from each class, which were further augmented with noise, distortion, rotation, and hue attributes to cover all potential camera settings.

IV. RESULT AND DISCUSSION

According to the requirement of the application, the live feed from the webcam will provide the necessary analysis for respective exercises. A repetition counter for exercises like pushups and squats and a pose correction feature for exercises like planks are included based on the requirement of a particular exercise. From the above-mentioned applications, we chose PoseNet since it can detect multiple people at the same time. It can also rule out poses that may get detected due to possible overlapping of various joints and aren't humanly possible.

As mentioned in Table 3, we can see that PoseNet isn't a very complex algorithm but it is comparatively smaller in size and offers a higher rate of frames per second when compared to OpenPose. Even though BlazePose offers a higher FPS count, it loses out to all the mentioned algorithms in terms of accuracy of detection. Another reason for choosing PoseNet is that it offers 33 keypoint locations as compared to 17 and 13 key points offered by other algorithms such as BlazePose and OpenPose.

As for the object detection algorithm, we studied all basic object detection model systems like Region Proposal, Yolo and Single Shot Detector as explained in section 2. As per the study, the region proposal is a very complex model considering the issue which is a simple equipment detection the priority here is to time and not accuracy since the application works on a live feed. Gym equipment are fairly distinct which won't create a lot of confusion when accuracy is concerned. Since among SSD and YOLO, SSD gives equal weightage to speed and accuracy but since exactness is not too much of a disquiet for us but, we require to execute at a faster speed since it is running on live feed we decide to use the Yolo model for object detection.

On deciding the Yolo version, it was analyzed that the only drawback with Yolo models is that it is unable to detect small images. When comparing different versions of Yolo, Yolov4 is comparatively better in detecting small images.

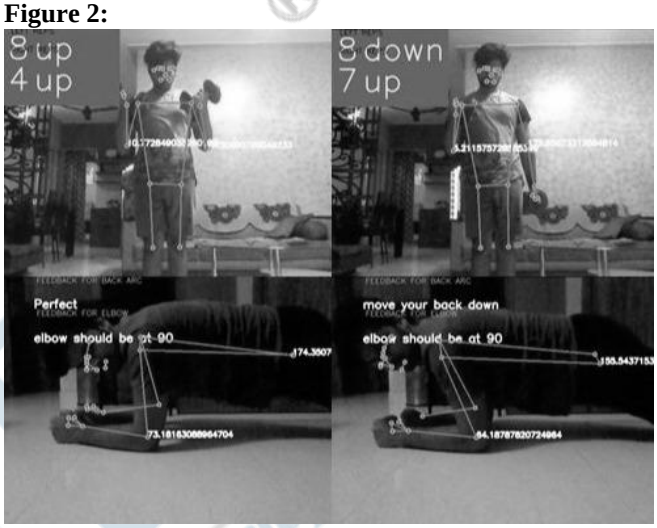
In Fig 3, we can see that the mean average precision for almost all the given classes is above 90%. The mean average precision for the entire model came out to be 93%, which can be considered pretty accurate. The model was trained for 4000 iterations to avoid overfitting, training was stopped at 4000 iterations since based on loss versus epoch graph a stagnation was observed after 4000 iterations.

Table 3:

ID	Equipment	AP (in %)
1	Punching bag	91.04
2	Dumbell	94.97
3	Indoor Rower	86.79
4	Gym Ball	95.71
5	Treadmill	96.97
6	Stationary Bicycle	97.54

Precision result of gym equipment from the proposed system

Figure 2:



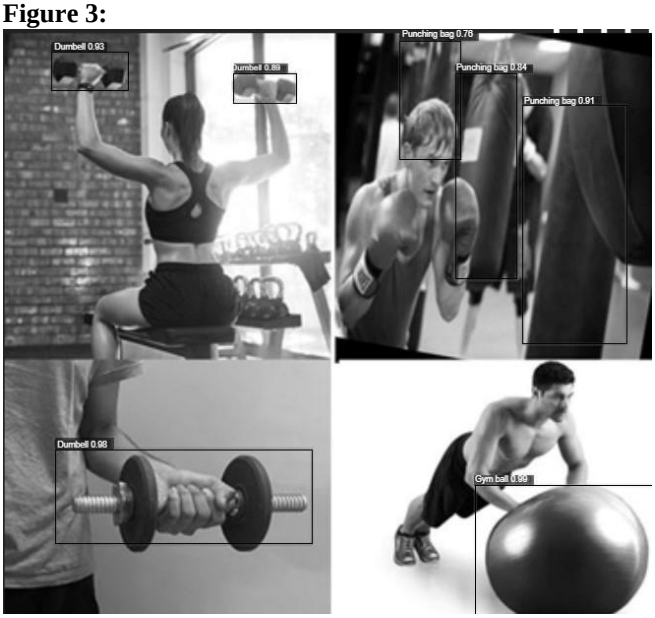
Sample result of exercise feedback for repetition counter in dumbbell and pose correction for plank.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 9, Issue 11, November 2022

Figure 3:



Result of object detection on test images

V. CONCLUSION

The final application consists of two modules, the object detection module with the YoloV4 model which first classifies the gym equipment and redirects to the exercise list based on the classification and the second module which works on the PoseNet model. This on choosing the exercise from the list will display the appropriate feedback as shown in Figure 2 below. Like while performing a plank the hand elbow and shoulder must be 90 degrees if not the model will suggest the change. If the back should be kept straight then the appropriate feedback will be provided for that.

This application can support the neophytes who want to exercise but don't have the proper time and resources to go to a gym. The above two applications will be able to solve all the issues of a beginner starting from the exercise list based on equipment and the posture as well. In the future addition, posture detection can be formed for the posture correction module.

REFERENCES

[1] S. Chen, B. Mulgrew, and P. M. Grant, "A clustering technique for digital communications channel equalization using radial basis function networks," IEEE Trans. on Neural Networks, vol. 4, pp. 570-578, July 1993.
[2] J. U. Duncombe, "Infrared navigation—Part I: An assessment of feasibility," IEEE Trans. Electron Devices, vol. ED-11, pp. 34-39, Jan. 1959.
[3] C. Y. Lin, M. Wu, J. A. Bloom, I. J. Cox, and M. Miller, "Rotation, scale, and translation resilient public watermarking for images," IEEE Trans. Image Process., vol. 10, no. 5, pp. 767-782, May 2001.
[4] Alexander Toshev, "DeepPose: Human Pose Estimation via Deep Neural Network," CVPR, 2014.
[5] Zhe Cao, "Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields," CVPR, 2017.
[6] Alejandro Newell, "Stacked Hourglass Networks for Human Pose Estimation," ECCV, 2016.
[7] Jamie Shotton, "Real-Time Human Pose Recognition in Parts from Single Depth Images," CVPR, 2014.
[8] Shih-En Wei, "Convolutional Pose Machines," CVPR, 2016.
[9] Petrissa Zell, "Joint 3D Human Motion Capture and Physical Analysis from Monocular Videos," CVPR, 2017.
[10] Steven Chen, "Pose Trainer: Correcting Exercise Posture using Pose Estimation," 2020.
[11] Amit Nagarkoti, "Realtime Indoor Workout Analysis Using Machine Learning ," IEEE, 2019.
[12] Kumari, Riya and Nikki, Shikha and Beg, Robin and Ranjan, Shashi and Gope, Sawan Kumar and Mallick, Ritesh Ranjan and Dutta, Arijit, A Review of Image Detection, Recognition and Classification with the Help of Machine Learning and Artificial Intelligence (May 26, 2020). International conference on Recent Trends in Artificial Intelligence, IOT, Smart Cities Applications (ICAISC-2020)
[13] Khan, Asharul Islam; Al-Habsi, Salim (2020). Machine Learning in Computer Vision. Procedia Computer Science, 167(), 1444–1451. doi:10.1016/j.procs.2020.03.355
[14] Yadav, Nikhil and Utkarsh Binay. "Comparative Study of Object Detection Algorithms." (2017).
[15] Mariano, Vladimir Min, Junghye Park, Jin Kasturi, Rangachar Mihalcik, David Li, Huiping Doermann, David Drayer, Thomas. (2002). Performance Evaluation of Object Detection Algorithms. Proceedings – International Conference on Pattern Recognition. 3. 965-969. 10.1109/ICPR.2002.1048198
[16] J. -a. Kim, J. -Y. Sung and S. -h. Park, "Comparison of FasterRCNN, YOLO, and SSD for Real-Time Vehicle Type Recognition," 2020 IEEE International Conference on Consumer Electronics - Asia (ICCE-Asia), 2020, pp. 1-4, doi: 10.1109/ICCE-Asia49877.2020.9277040.
[17] Li, Min et al. "Agricultural Greenhouses Detection in HighResolution Satellite Images Based on Convolutional Neural Networks: Comparison of Faster R-CNN, YOLO v3 and SSD." Sensors (Basel, Switzerland) 20 (2020)
[18] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi," You only look once: Unified, real-time object detection",2016
[19] Ross Girshick et al. "Rich feature hierarchies for accurate object detection and semantic segmentation,2014
[20] Z. Cao, G. Hidalgo, T. Simon, S. -E. Wei and Y. Sheikh, "OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, no. 1, pp. 172186, 1 Jan. 2021, doi: 10.1109/TPAMI.2019.2929257.
[21] Daniil Osokin, "Real-time 2D Multi-Person Pose Estimation on CPU: Lightweight OpenPose", Submitted on 29 Nov 2018
[22] Valentin Bazarevsky, Ivan Grishchenko, Karthik Raveendran, Tyler Zhu, Fan Zhang, Matthias Grundmann, "BlazePose: Ondevice Real-time Body Pose tracking", Submitted on 17 Jun 2020
[23] Yu Chen, Chunhua Shen, Xiu-Shen Wei, Lingqiao Liu, Jian Yang; Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017, pp. 1212-1221
[24] Alex Kendall, Matthew Grimes, Roberto Cipolla; Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2015, pp. 2938-2946
[25] Chiddarwar, Girija Gireesh, et al. "AI-based yoga pose estimation for android application." Int J Inn Scien Res Tech 5 (2020): 1070-1073.
[26] Ohri, Ashish, Shashank Agrawal, and Garima S. Chaudhary. "On-device Realtime Pose Estimation Correction."

[27] Chen, Steven, and Richard R. Yang. ”Pose Trainer: correcting exercise posture using pose estimation.” arXiv preprint arXiv:2006.11718 (2020).

[28] Rakbeh, Hussein, et al. ”Automatic Feedback For Physiotherapy Exercises Based On PoseNet.” Information Bulletin in Computers and Information 2.2 (2020): 10-14.

[29] Huesser, Cloe, Simon Schubiger, and Arzu C , ¨oltekin. ”Gesture Interaction in Virtual Reality.” IFIP Conference on HumanComputer Interaction. Springer, Cham, 2021.