# Human Pose Estimation for Fitness Exercise Movement Correction

1st Atharian Rahmadani
*Dept. of Informatics and Computer Engineering*
*Politeknik Elektronika Negeri Surabaya*
Surabaya, Indonesia
atharianrahmadani@gmail.com

2nd Bima Sena Bayu Dewantara
*Dept. of Informatics and Computer Engineering*
*Politeknik Elektronika Negeri Surabaya*
Surabaya, Indonesia
bima@pens.ac.id

3rd Dewi Mutiara Sari
*Dept. of Informatics and Computer Engineering*
*Politeknik Elektronika Negeri Surabaya*
Surabaya, Indonesia
dewi.mutiara@pens.ac.id

2022 International Electronics Symposium (IES) | 978-1-6654-8971-3/22/$31.00 ©2022 IEEE | DOI: 10.1109/IES55876.2022.9888451

***Abstract*—Based on computer vision technology, this research suggests an application to identify and assess a fitness practitioner's movements. Several fitness movements such as lifting weights, squat jumps, and pull-ups that are very beneficial for health and body fitness become the main movement for body building. However, those kinds of activities may be very dangerous if done incorrectly. Based on the problem, we developed an application based on computer vision to recognize and correct the pose accuracy of fitness practitioners by using input in the form of videos that record the movements of fitness practitioners continuously. To categorize the many forms of fitness sport movements, this system uses the support vector machine (SVM) method. On the monitor screen, the classification results will be visible. The result shows that the accuracy of the system is 96.87% by using SVM with the Radial Basis Function (RBF) kernel type and can make corrections to four types of fitness movements with a testing accuracy of 90.62%.**

***Keywords—Human Pose Estimation, OpenPose, Sports, Skeleton, Fitness***

## I. INTRODUCTION

Fitness sports activities such as lifting weights, squat jumps, and pull-ups are very beneficial for health and fitness, but can also be very dangerous if done incorrectly. The heavy weights involved in these training activities can cause severe injury to the muscles or ligaments. Many people exercise and do these exercises regularly but do not maintain proper form of movement (pose). This could be due to a lack of formal training through classes or personal trainers [1].

The presence of an instructor when doing sports can be the right choice. With an instructor when exercising, this can help to make sports movements more directed and on target. Instructors also provide motivation and encouragement so that sports players remain enthusiastic in carrying out sports movements. The presence of a professional instructor can see how capable our body is in making the right and correct movements. Thus, exercise can be carried out optimally and optimally.

However, during a pandemic, it is definitely not easy to bring an instructor to do sports. The application of physical distancing which requires us to keep our distance from other people is one of the factors that makes it difficult to bring instructors. The cost factor is expensive if we recruit instructors online. During this pandemic, many people were affected economically, so they had to save on expenses and use money efficiently. Meanwhile, to recruit instructors online, it costs an average of 200,000 rupiah per session.

In this study, a software application for the implementation of integrated human pose estimation is proposed to correct computer vision-based fitness sports movements. The system of this application can recognize and correct the accuracy of poses from fitness practitioners by using input in the form of videos that record the movements of fitness practitioners continuously. This application uses the support vector machine (SVM) method to classify or recognize poses from fitness practitioners. Once the pose is recognized, the system evaluates the accuracy of the movement based on the angular geometry of each joint. The output of this system is to display the type of movement and advice on the monitor regarding the accuracy of the movement or pose of the exercise performed.

The remainder of this paper is organized as follows. Section II discuses some literature studies regarding our system's supporting functions. Section III discuses how we implemented the system proposed. Section IV discuses the test and result of the system implementation. Section V is conclusion about this work.

## II. LITERATURE STUDY

### *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 [2].

### *B. Action Recognition*

The conclusion that a person draws from an action is known as action recognition. Inferences are made by using a sample of data in the form of graphs, numbers, and linguistics. Activity recognition utilizes numerous prior data samples, whereas action recognition just uses one. Action recognition can be done using the Bag of Features (BoF), Dense Trajectory Features (DTF), Improved Dense Trajectory (IDT) methods, and the most widely used is Deep Learning, with image data and non-image data input. In this study, action recognition using the SVM method and input in the form of images from the camera is used. Problems that may arise for the system are susceptibility or unrelated backgrounds, this is usually solved by providing limited image input to the people who will be introduced to the action rather than fully in the training or introduction phase.

### *C. OpenPose*

The first real-time system, OpenPose, can identify 135 important points in a single image, including body, hand, face, and foot key points. Together with Zhe Cao, Tomas Simon, Shih-En Wei, Hanbyul Joo, and Yaser Sheikh, Ginés



Authorized licensed use limited to: Dongguk University. Downloaded on December 30,2023 at 08:56:11 UTC from IEEE Xplore. Restrictions apply.

Hidalgo developed OpenPose. Ginés Hidalgo and Yaadhav Raaj are in charge of running OpenPose.

Human re-identification, retargeting, and Human-Computer Interaction are just a few examples of the many research subjects involving human analysis that now make use of the OpenPose library. Additionally, OpenPose has been added to the OpenCV library [3]. Fig. 1 shows the OpenPose pipelines.

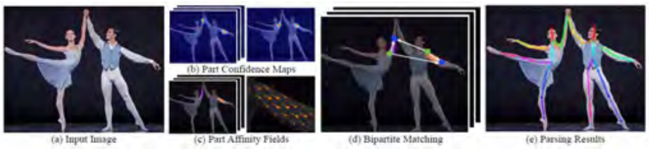



Fig. 1. OpenPose Pipelines

### D. 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. Fig. 2 shows the COCO body format index.

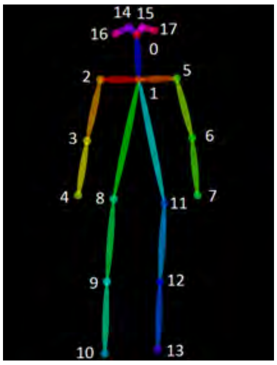



Fig. 2. COCO Body Format

### E. 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 [4]. Fig. 3 shows skeletal angle pair.

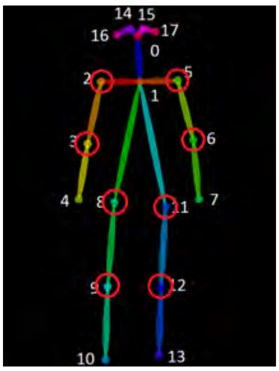



Fig. 3. Skeletal Angle Pair

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. Fig. 4 and Eq. (1) show how to get the skeletal angle values using the inverse cosine equation.

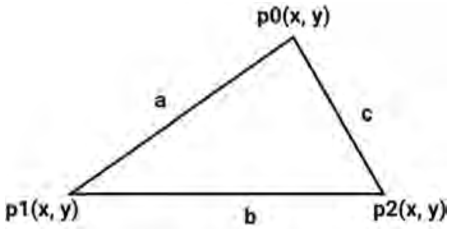



Fig. 4. Sided Angles of Each Joint

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

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).

$$\begin{aligned} a &= \sqrt{\left(p1(x) - p0(x)\right)^2 + \left(p1(y) - p0(y)\right)^2} \\ b &= \sqrt{\left(p1(x) - p2(x)\right)^2 + \left(p1(y) - p2(y)\right)^2} \\ c &= \sqrt{\left(p2(x) - p0(x)\right)^2 + \left(p2(y) - p0(y)\right)^2} \end{aligned} \tag{2}$$

### F. 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 center of the frame. Eq. (3) shows how to convert pixel coordinates to real coordinates in meters for x and y axis.

$$\begin{aligned} pos\_x &= \left(x - \frac{width}{2}\right) \times z \times SCL \\ pos\_y &= \left(y - \frac{height}{2}\right) \times z \times SCL \end{aligned} \tag{3}$$

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.

### G. Support Vector Machine

When classifying or regressing a set 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 [5].

Data points of the following type are taken into consideration:

$$\{(x_1,y_1),(x_2,y_2),(x_3,y_3),(x_4,y_4)\ldots\ldots\ldots,(x_n,\ y_n)\},$$



Authorized licensed use limited to: Dongguk University. Downloaded on December 30,2023 at 08:56:11 UTC from IEEE Xplore. Restrictions apply.

The class to which that point $x_n$ belongs is indicated by the constant $y_n$=1/-1. Sample size is indicated by the notation $n$. Each $x$ is a real, p-dimensional vector. Scaling is crucial to protect against variables with higher volatility. The dividing hyperplane, which includes Eq. (4), allows us to see this training data.

$$w.\, x + b = 0 \qquad (4)$$

A scalar value for $b$, and a p-dimensional vector for $w$. The separating hyperplane is parallel to the vector $w$, which points there. We can enhance the margin by including the offset parameter $b$. Without $b$, the hyperplane must pass through the origin, which limits the solution. SVM and parallel hyperplanes are particularly relevant to us because we are interested in the largest margin. Eq. (5) can be used to describe parallel hyperplanes.

$$\begin{aligned} w.\, x + b &= 1 \\ w.\, x + b &= \text{-}1 \end{aligned} \qquad (5)$$

We can choose these hyperplanes so that there are no points between them and then aim to maximize their distance if the training data are linearly separable. We determine the distance between the hyperplane to be $2/|w|$ through geometry. Therefore, we desire to reduce $|w|$. We apply the Eq. (6) to stimulate data points.

$$w.\, x_i - b \geq 1 \text{ or } w.\, x_i - b \leq -1 \qquad (6)$$

It can alternatively be written as Eq. (7).

$$y_i\,(w.\, xi - b) \geq 1,\ 1 \leq i \leq n \qquad (7)$$

From these formulas, we can find the hyperplane and its margin to separate and classify two types of classes. Fig. 5 shows maximum margin of SVM hyperplane with sample from two classes [6].

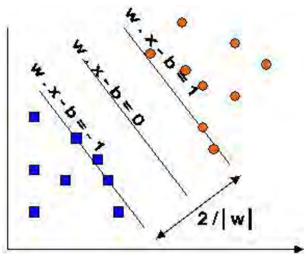



Fig. 5. Maximum margin hyperplanes for a SVM trained with samples from two classes

An input feature vector is projected in a high-dimensional space using an internal kernel function in this algorithm, which then linearizes the features and classifies them. SVM can be used to discriminate input observations and select the optimum hyperplane in the converted space for classification. SVM has advantages in small sample pattern recognition and was first presented for binary classification. Many frequently used kernel functions, including linear, polynomial (poly), and radial basis function (RBF), are available in SVM [7].

### H. Geometry Evaluation

Geometry Evaluation is used to correct the accuracy of the fitness movements performed. This process is done by set the fixed main angles for each movement. Then this value is compared with the main joint value obtained directly from the program. From the comparison results, an error value will be obtained which can then be used as a motion correction system.

## III. Implementation

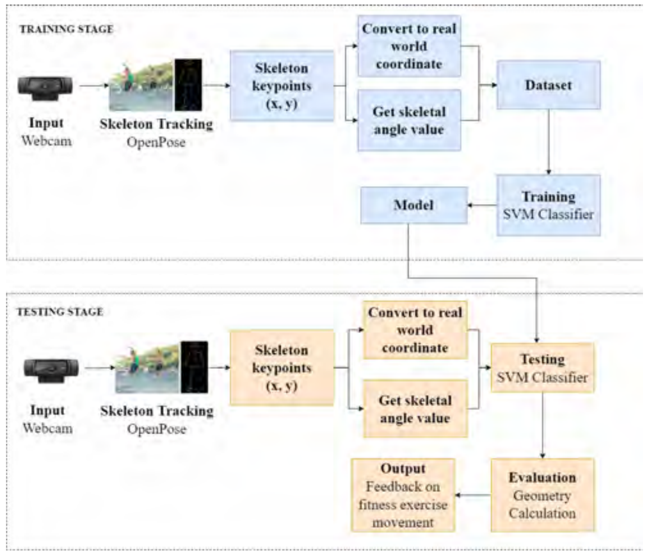



Fig. 6. System Block Diagram

### 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. Fig. 7 shows the results of camera access.

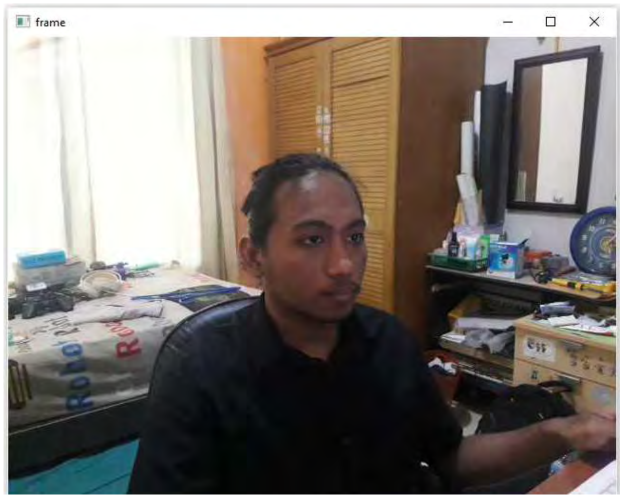



Fig. 7. OpenCV Camera Access Results

### 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. Fig. 8 shows the skeleton detection results.



Authorized licensed use limited to: Dongguk University. Downloaded on December 30,2023 at 08:56:11 UTC from IEEE Xplore. Restrictions apply.

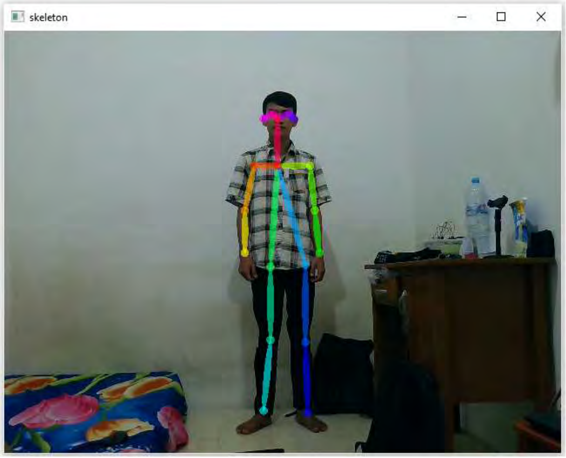



Fig. 8. Skeleton Detection Results

*C. Skeleton Keypoints*

At 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 carried out 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 find the distance between the angles that are flanking each other, then the inverse cosine is used to get the angle value based on 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 center of the frame. Fig. 9 shows the visualization of real coordinate conversion.

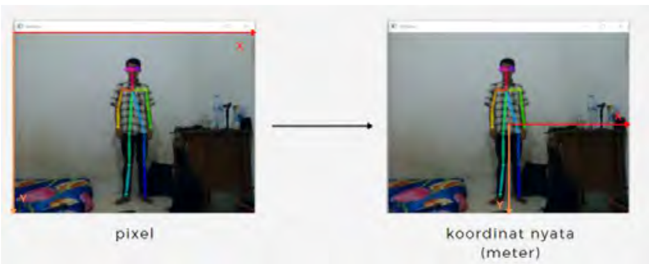
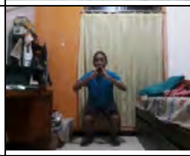
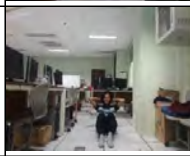
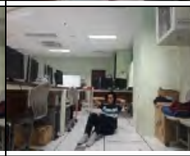
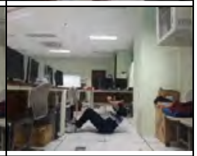



Fig. 9. Coordinate Conversion Visualization

*F. Creating Datasets*

The dataset was made by collecting pictures of human object poses doing fitness movements, bicep curls, shoulder presses, squats, sit ups, pushups, planks, and triceps drips. The dataset is made with human objects that are 2.4 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 I shows the example of obtained image for creating datasets.

TABLE I. EXAMPLE OF IMAGE DATASET

| | | | |
|---|---|---|---|
| Bicep Curl |  |  |  |
| Shoulde r Press |  |  |  |
| Squat |  |  |  |
| Sit Up |  |  |  |
| Plank |  |  |  |

Pictures are taken of three different people in each type of movement. There are eight image data per person, resulting in 24 data for each movement. The images that have been obtained will be processed for the calculation of the conversion of real coordinates and skeletal angles. The calculated data will be saved to a txt file and written in a certain format. Fig. 10 shows the example of dataset writing format.

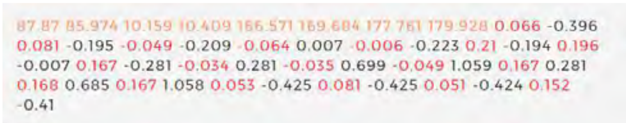

87.87 85.974 10.159 10.409 166.571 169.684 177.761 179.928 0.066 -0.396 0.081 -0.195 -0.049 -0.209 -0.064 0.007 -0.006 -0.223 0.21 -0.194 0.196 -0.007 0.167 -0.281 -0.034 0.281 -0.035 0.699 -0.049 1.059 0.167 0.281 0.168 0.685 0.167 1.058 0.053 -0.425 0.081 -0.425 0.051 -0.424 0.152 -0.41

Fig. 10. Dataset Writing Format

The saving format begins with a set of orange numbers which are the values of the skeletal angle calculations where the value is a decimal number or float. There are eight data in the written order format, namely the skeletal angles of the shoulder, elbow, hip, and knee. Next there is a combination of a set of numbers with decimal values or red and black floats. The red color is the value of the real x-axis coordinates, and the black color is the value of the real y-axis coordinates. There are 18 data in the written order format according to the COCO Human Body Part Model index. The data is written in one line with a space separator. So, the total data obtained is 44 data.

*G. Data Training*

The data training was carried out using Google Colab with the prepared dataset. There are 4 types of data used, namely bicep curl, shoulder press, squat, and sit up. The training is carried out using the SVM method using the Scikit-learn library.



Authorized licensed use limited to: Dongguk University. Downloaded on December 30,2023 at 08:56:11 UTC from IEEE Xplore. Restrictions apply.

## H. 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 then be used as a correction system for whether the movement is correct or not.

In this research, there are four types of fitness movement will be corrected. Bicep curl and shoulder press have elbows as main joint. Squat and sit up have hips and knees as main joints. For bicep curl, main joint must between 45 and 60 degree value. For shoulder press, main joint must have 90-degree value. For squat, the hips must between 45 and 75 degree value and the knee must between 50 and 60 degree value. For sit up, the hips must have below 90 degree value and the knee must have 90 degree value.

# IV. Experimental Results

This section explained the testing of the system that has been made to determine whether the system can work as designed. With this test, the advantages and disadvantages of the system will be known.

## A. System Specification

The system is created and implemented using a PC. There are several specifications used to run the system that has been made. Table II shows the specification of the system used.

TABLE II. System Specification

| Description | Specification |
|---|---|
| Processor | Intel® Core™ i7-4720HQ CPU @ 2.60GHz |
| Memory | 8.00 GB |
| Operating System | Windows 10 Home Single Language |
| *Software build* | Visual Code |
| *Library* | OpenCV 4.5.3<br>OpenPose<br>Sci-kit Learn |
| Camera | ASUS N550JX Laptop Webcam<br>NYK A95 Webcam |
| Resolution | 1280x720 pixels @30 fps |

## B. Testing Data Setup

There are also specifications for taking images of objects for test data.

- The camera position does not change, and the placement height is ±1 meter from the floor.
- The number of human objects is only one person.
- The position of the human object is ± 2.4 meters in front of the camera.
- The height of the human object is a minimum of 150 cm and a maximum of 180 cm.
- The room must have lighting with a minimum of 100 lux.

## C. SVM Method Configuration

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 studies that perform posture analysis to classify activities carried out using SVM [8][9].

The SVM training model uses several parameters, including the RBF kernel, which has C ranges from 0.1 to 100 and gamma ranges from 0.0001 to 1. Fig. 11 shows the SVM configuration.

```
param_grid={'C':[0.1,1,10,100],'gamma':[0.0001,0.001,0.1,1],'kernel':['rbf']}
svc=svm.SVC(probability=True)

print("The training of the model is started, please wait for while as it may tak

model=GridSearchCV(svc,param_grid)
model.fit(x_train,y_train)
```

Fig. 11. SVM Configuration using Sci-kit Library

After the training, we also must know the best parameters used in the training process. We can use the '*best_params_*' function from Sci-kit learn. The function can be called after the model trained. Fig. 12 shows the model best parameters.

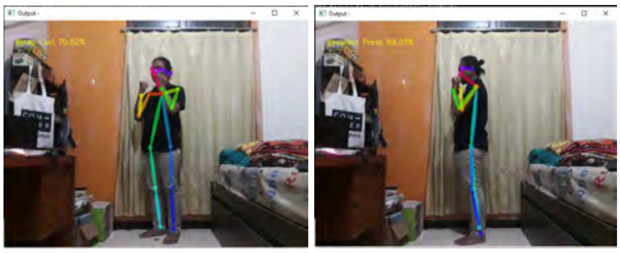
```
print('The Model is trained well')
model.best_params_
The Model is trained well
{'C': 1, 'gamma': 0.0001, 'kernel': 'rbf'}
```

Fig. 12. Model Best Parameters

From Fig. 11, we can know that the best parameters used for training is 1 for the C value and 0.0001 for the gamma value.

## D. Movement Classification

The result of the system made is that it can detect and recognize fitness movements based on RGB images. Fig. 13 shows the example of classification system results.



(a) (b)

Fig. 13. The results of Classification System, (a) True Prediction, (b) False Prediction

To find out the percentage of the success rate of the system, testing was carried out on the classification system. Eight different samples with training datasets were used for each type of fitness movement to test the system that had been created. So, there are 32 samples for this test. Table III shows the classification system test results.

TABLE III. Result of Classification System Testing

| No | Real | Prediction | Confidence Score (%) |
|---|---|---|---|



Authorized licensed use limited to: Dongguk University. Downloaded on December 30,2023 at 08:56:11 UTC from IEEE Xplore. Restrictions apply.

| | | | |
|---|---|---|---|
| **Bicep Curl** | | | |
| 1 | Bicep Curl | Bicep Curl | 57.53 |
| 2 | Bicep Curl | Bicep Curl | 76.55 |
| 3 | Bicep Curl | Bicep Curl | 80.84 |
| 4 | Bicep Curl | Shoulder Press | 69.03 |
| 5 | Bicep Curl | Bicep Curl | 54.65 |
| 6 | Bicep Curl | Bicep Curl | 63.17 |
| 7 | Bicep Curl | Bicep Curl | 72.22 |
| 8 | Bicep Curl | Bicep Curl | 61.73 |
| **Shoulder Press** | | | |
| 1 | Shoulder Press | Shoulder Press | 90.02 |
| 2 | Shoulder Press | Shoulder Press | 89.70 |
| 3 | Shoulder Press | Shoulder Press | 89.96 |
| 4 | Shoulder Press | Shoulder Press | 75.43 |
| 5 | Shoulder Press | Shoulder Press | 37.06 |
| 6 | Shoulder Press | Shoulder Press | 90.31 |
| 7 | Shoulder Press | Shoulder Press | 89.84 |
| 8 | Shoulder Press | Shoulder Press | 90.40 |
| **Squat** | | | |
| 1 | Squat | Squat | 38.15 |
| 2 | Squat | Squat | 89.22 |
| 3 | Squat | Squat | 80.37 |
| 4 | Squat | Squat | 85.02 |
| 5 | Squat | Squat | 87.12 |
| 6 | Squat | Squat | 82.99 |
| 7 | Squat | Squat | 55.82 |
| 8 | Squat | Squat | 47.19 |
| **Sit Up** | | | |
| 1 | Sit Up | Sit Up | 82.40 |
| 2 | Sit Up | Sit Up | 90.16 |
| 3 | Sit Up | Sit Up | 81.59 |
| 4 | Sit Up | Sit Up | 67.74 |
| 5 | Sit Up | Sit Up | 81.91 |
| 6 | Sit Up | Sit Up | 90.96 |
| 7 | Sit Up | Sit Up | 89.22 |
| 8 | Sit Up | Sit Up | 85.18 |

From the results of testing with eight samples for each type of movement, there is only one result that is less precise on the type of bicep curl, so that an accuracy of 96.87% is obtained in all movements. The bicep curl movement which is predicted as a shoulder press is a movement when the user is facing to the side. This can be influenced by the similarity of the coordinates when doing the bicep curl and shoulder press facing sideways.

There are average confidence scores of 66.96% for bicep curls, 81.59% for shoulder presses, 70.73% for squats, and 83.27% for sit ups. The bicep curl movement has the lowest average confidence score, it could be because when the test is facing sideways there are body parts that are covered by other body parts, causing the skeleton data to be inaccurate. Confidence score also can be influenced by the similarity of the types of movements classified.

### E. *Movement Correction*

The result of the system made is that it can correct the predicted movement. Fig. 14 shows the example of correction system results.


(a) (b)

Fig. 14. The results of Correction System, (a) Correction result of Shoulder Press movement with correct elbow condition, (b) Correction result of Shoulder Press movement with incorrect elbow condition

To find out the percentage of the success rate of the system, testing was carried out on the correction system. Eight samples of each type of movements were used to test the system that had been created. So, there are 96 joint samples for this test. Table IV shows correction system test results.

TABLE IV. RESULT OF CORRECTION SYSTEM TESTING

| No | Main Joint Name | Actual Correction | System Correction |
|---|---|---|---|
| **Bicep Curl** | | | |
| 1 | Right Elbow | Correct | Incorrect |
| | Left Elbow | Correct | Incorrect |
| 2 | Right Elbow | Correct | Correct |
| | Left Elbow | Correct | Correct |
| 3 | Right Elbow | Correct | Correct |
| | Left Elbow | Correct | Correct |
| 4 | Right Elbow | Correct | Incorrect |
| | Left Elbow | Correct | Correct |
| 5 | Right Elbow | Correct | Incorrect |
| | Left Elbow | Correct | Correct |
| 6 | Right Elbow | Incorrect | Incorrect |
| | Left Elbow | Incorrect | Incorrect |
| 7 | Right Elbow | Incorrect | Incorrect |
| | Left Elbow | Correct | Correct |
| 8 | Right Elbow | Incorrect | Incorrect |
| | Left Elbow | Incorrect | Incorrect |
| **Shoulder Press** | | | |
| 1 | Right Elbow | Correct | Correct |
| | Left Elbow | Correct | Correct |
| 2 | Right Elbow | Correct | Correct |
| | Left Elbow | Incorrect | Incorrect |
| 3 | Right Elbow | Correct | Correct |
| | Left Elbow | Incorrect | Incorrect |
| 4 | Right Elbow | Correct | Correct |
| | Left Elbow | Correct | Correct |
| 5 | Right Elbow | Correct | Correct |
| | Left Elbow | Correct | Correct |
| 6 | Right Elbow | Incorrect | Incorrect |
| | Left Elbow | Incorrect | Incorrect |
| 7 | Right Elbow | Correct | Correct |
| | Left Elbow | Correct | Correct |
| 8 | Right Elbow | Incorrect | Incorrect |
| | Left Elbow | Incorrect | Incorrect |
| **Squat** | | | |
| 1 | Right Hip | Correct | Correct |
| | Left Hip | Correct | Correct |



Authorized licensed use limited to: Dongguk University. Downloaded on December 30,2023 at 08:56:11 UTC from IEEE Xplore. Restrictions apply.

| | | | |
|---|---|---|---|
| | Right Knee | Incorrect | Incorrect |
| | Left Knee | Incorrect | Incorrect |
| 2 | Right Hip | Correct | Incorrect |
| | Left Hip | Correct | Correct |
| | Right Knee | Incorrect | Incorrect |
| | Left Knee | Incorrect | Incorrect |
| 3 | Right Hip | Correct | Correct |
| | Left Hip | Correct | Correct |
| | Right Knee | Correct | Incorrect |
| | Left Knee | Correct | Correct |
| 4 | Right Hip | Correct | Correct |
| | Left Hip | Correct | Correct |
| | Right Knee | Correct | Correct |
| | Left Knee | Correct | Incorrect |
| 5 | Right Hip | Incorrect | Incorrect |
| | Left Hip | Incorrect | Incorrect |
| | Right Knee | Incorrect | Incorrect |
| | Left Knee | Incorrect | Incorrect |
| 6 | Right Hip | Incorrect | Incorrect |
| | Left Hip | Incorrect | Incorrect |
| | Right Knee | Incorrect | Incorrect |
| | Left Knee | Incorrect | Incorrect |
| 7 | Right Hip | Correct | Correct |
| | Left Hip | Correct | Correct |
| | Right Knee | Correct | Correct |
| | Left Knee | Correct | Correct |
| 8 | Right Hip | Correct | Correct |
| | Left Hip | Correct | Correct |
| | Right Knee | Correct | Incorrect |
| | Left Knee | Correct | Correct |
| **Sit Up** | | | |
| 1 | Right Hip | Correct | Correct |
| | Left Hip | Correct | Correct |
| | Right Knee | Incorrect | Incorrect |
| | Left Knee | Incorrect | Incorrect |
| 2 | Right Hip | Correct | Correct |
| | Left Hip | Correct | Correct |
| | Right Knee | Correct | Incorrect |
| | Left Knee | Correct | Correct |
| 3 | Right Hip | Correct | Correct |
| | Left Hip | Correct | Correct |
| | Right Knee | Incorrect | Incorrect |
| | Left Knee | Incorrect | Incorrect |
| 4 | Right Hip | Correct | Correct |
| | Left Hip | Correct | Correct |
| | Right Knee | Incorrect | Incorrect |
| | Left Knee | Incorrect | Incorrect |
| 5 | Right Hip | Correct | Correct |
| | Left Hip | Correct | Correct |
| | Right Knee | Correct | Correct |
| | Left Knee | Correct | Incorrect |
| 6 | Right Hip | Correct | Correct |
| | Left Hip | Correct | Correct |
| | Right Knee | Incorrect | Incorrect |
| | Left Knee | Incorrect | Incorrect |
| 7 | Right Hip | Correct | Correct |
| | Left Hip | Correct | Correct |
| | Right Knee | Correct | Correct |
| | Left Knee | Correct | Correct |
| 8 | Right Hip | Correct | Correct |
| | Left Hip | Correct | Correct |
| | Right Knee | Correct | Correct |
| | Left Knee | Correct | Incorrect |

The movement that has the highest error rate is the Squat. This movement requires the human object to face sideways in a squatting position so that the condition of one of the main joints will be covered by parts of the human object's body and cause the program to not read one of the joints accurately. In other movements, the joint that causes the correction to be validated incorrectly is mostly a joint that is closed by another body part when the human object performs a movement that requires it to face sideways.

From all the results tested, there were nine data that were validated incorrectly from a total of 96 main joint data. So that the accuracy of the movement correction system testing is 90.62% in correcting four types of movements.

## V. Conclusion

We have built a fitness movements evaluating system to detect and correct a fitness practitioner movements when applying lifting weights, squat jumps and sit-ups movements based on computer vision and artificial intelligent approaches. The research and test results indicate that the system can distinguish the frame from the camera's acquired image, the system can classify 4 types of fitness poses with the accuracy of 96.87% using SVM with RBF kernel type, and a combination of frame coordinate features. joint and skeletal angle values can be used as a fitness pose classification dataset. The system also can correct the predicted movement with the accuracy of 90.62%. For future work, we will need to expand the datasets and adding a lot more variety of fitness movement types.

## VI. References


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

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

[3] 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.

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

[5] Vapnik, Vladimir N. “The nature of statistical learning”. Theory. 1995.

[6] Durgesh, K. SRIVASTAVA, and B. Lekha. "Data classification using support vector machine." Journal of theoretical and applied information technology. Vol. 12, No. 1: pp. 1-7. 2010.

[7] Han, Kun, Qiongqian Yang, and Zefan Huang. “A two-stage fall recognition algorithm based on human posture features”. Sensors. Vol. 20, No. 23: pp. 6966. 2020.

[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.




Authorized licensed use limited to: Dongguk University. Downloaded on December 30,2023 at 08:56:11 UTC from IEEE Xplore. Restrictions apply.