POSE TRAINER

Harit Gedia (B18EE017)
Anshika Gupta (B18EE004)
Supervised by : Dr. Himanshu Kumar

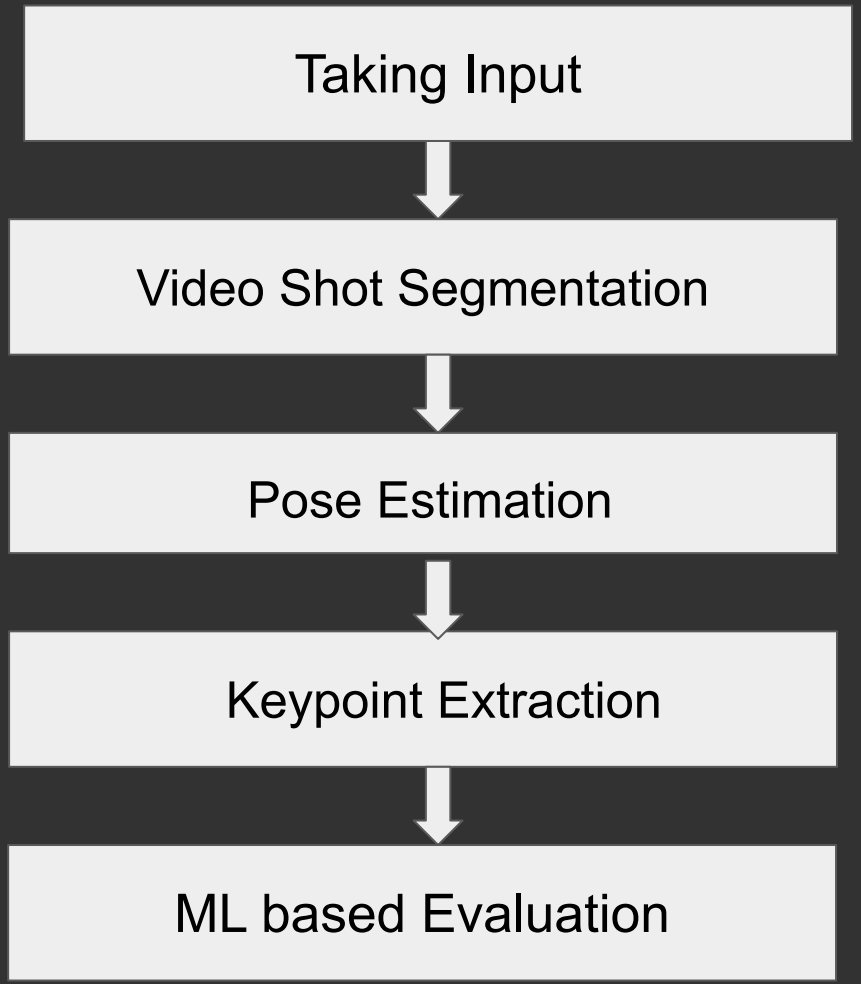
WHAT DOES POSE TRAINER DO?

It takes the recorded video of Exercise as input and tells us where we did go wrong.

MOTIVATION

- In many areas, there is a shortage of trainers to guide people to proper fitness.
- Doing exercise in the wrong manner may affect adversely.
- This application tries to reduce human interdependence.

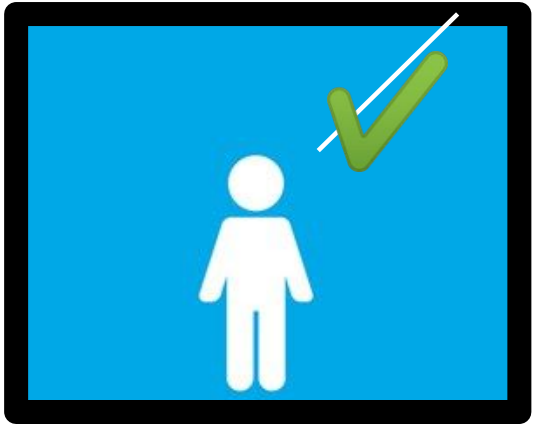
PROPOSED APPROACH

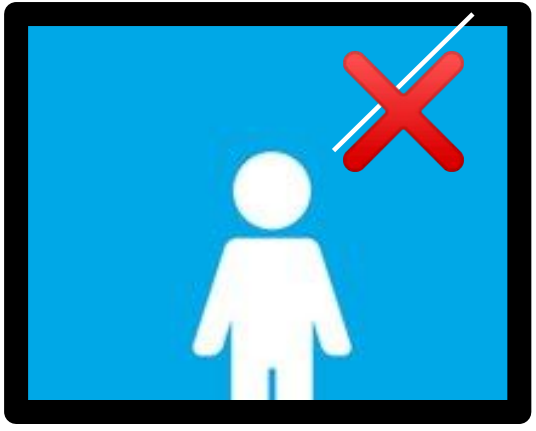


TAKING INPUT

We can give inputs in any video format.

Our whole body should be visible and facing front.(standing at the center is preferred)





COLLECTION OF DATASET

One Yoga Exercise (Prayer Pose)

5-Correct examples

30-Incorrect examples (30fps, 60fps)

VIDEO SHOT SEGMENTATION

We can select important Key frames of the poses as our Reference and find out the frame of input with maximum similarity.

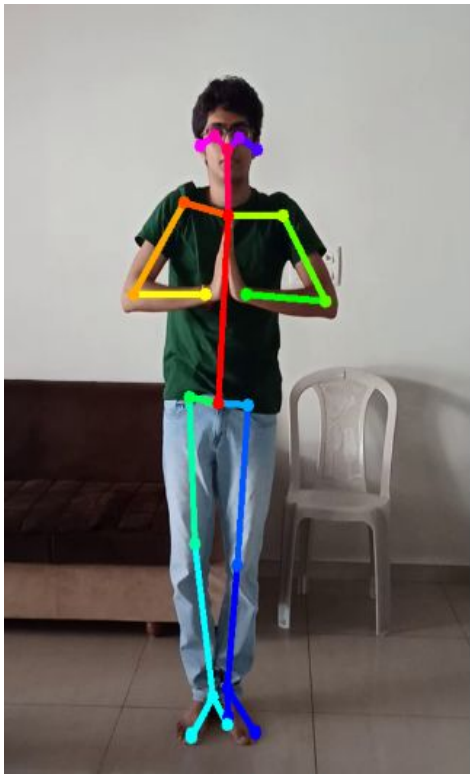
However, as we have not analysed it yet, we are comparing every frame of reference with the input Video.

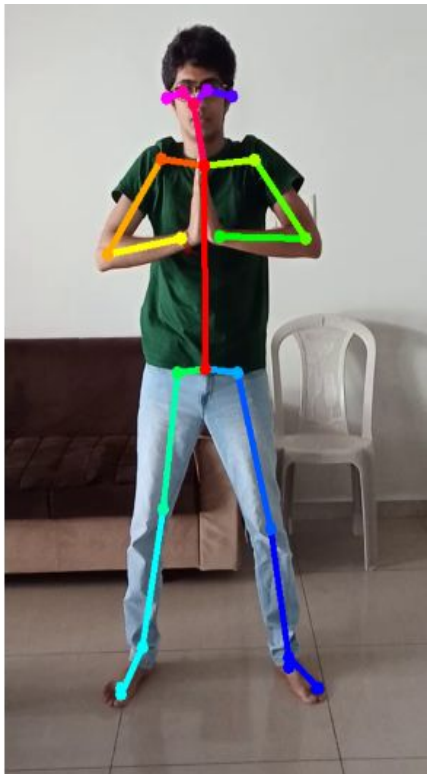
POSE ESTIMATION

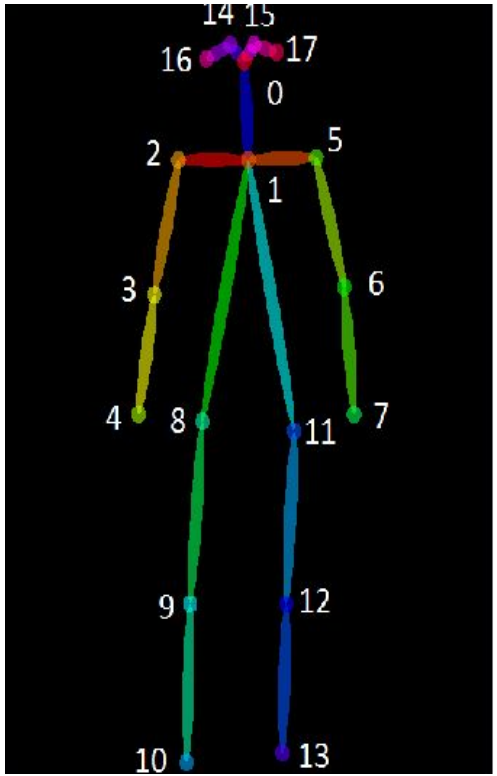
As we are evaluating the Exercise by comparing the poses, Estimation of Pose is an important step.

It is hard to find the poses of the same body types. Thus, we are using a pre-trained model Openpose to estimate the pose in skeletal form.

OUTPUT OF OPENPOSE





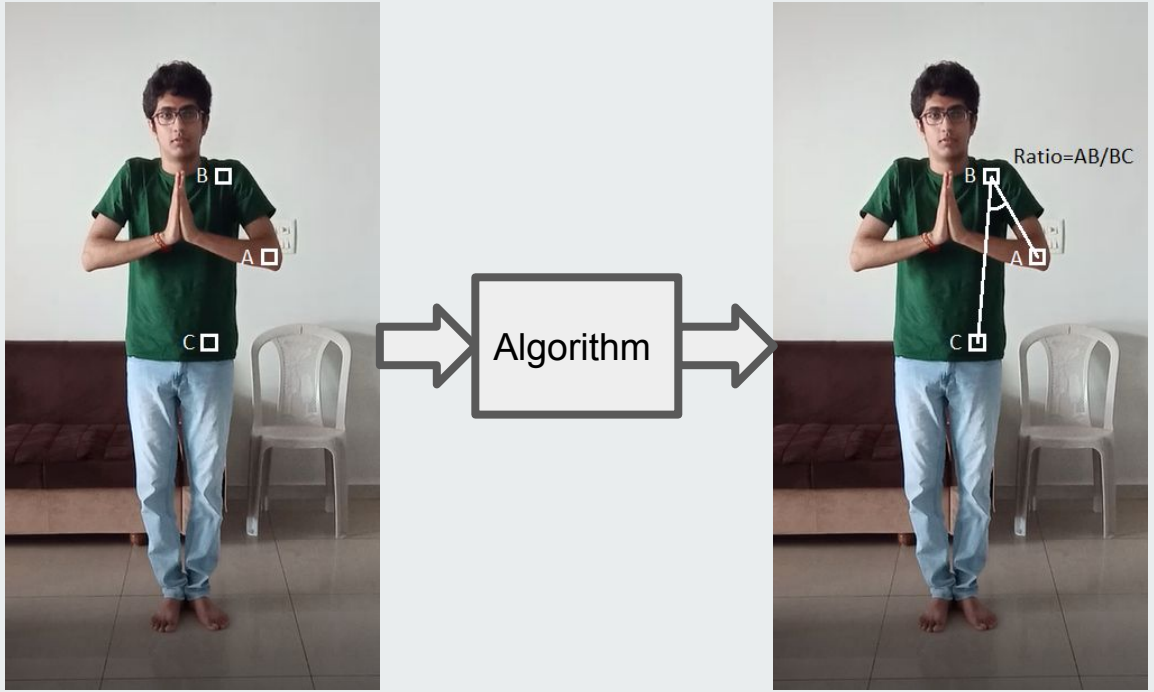


DATA EXTRACTION

- OpenPose gives the raw input in terms of location of Key-points.
- We measure the euclidean distances between the associated points.
- We use those distances to find the ratios and angles between the joints

For Example:

Thus, instead of storing (x,y,acc) of three points, only (ratio, angle) of one joint is stored



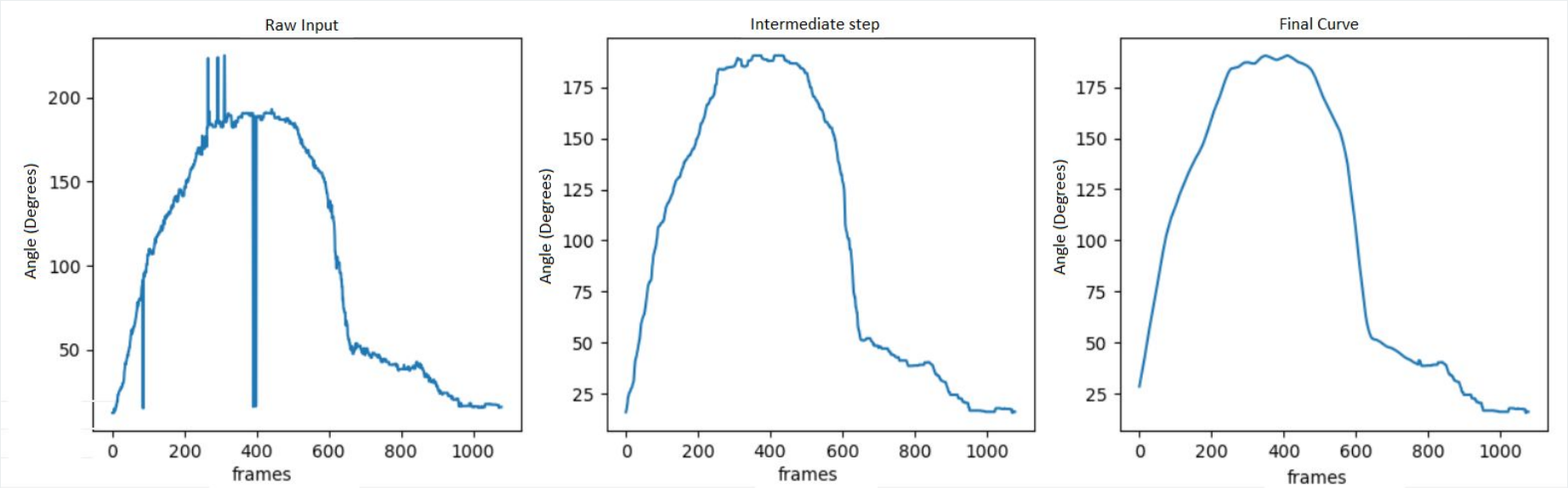
NOISE REMOVAL:

Output of Openpose is fluctuating, and consist of irregularities, which is needed to be removed

To remove fluctuations, we use Median

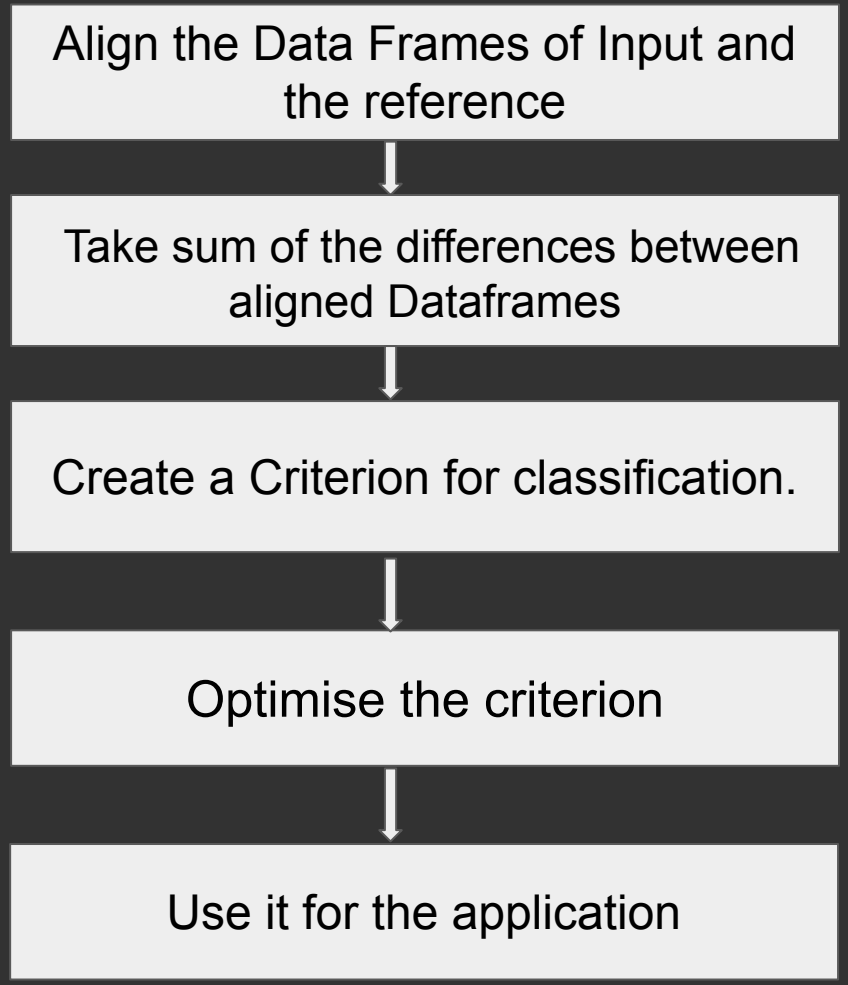
To smoothen the curve, we use Mean.

RESULT



Left-shoulder v/s frames

WORKFLOW OF ML BASED EVALUATION

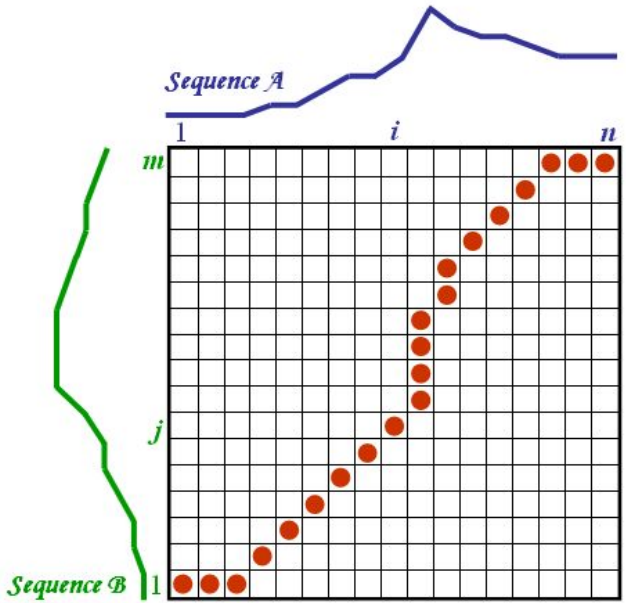


DTW ALGORITHM

Dynamic Time Warping is an algorithm which measures similarity between two sequences which may differ in time by comparing their Euclidean Distance.

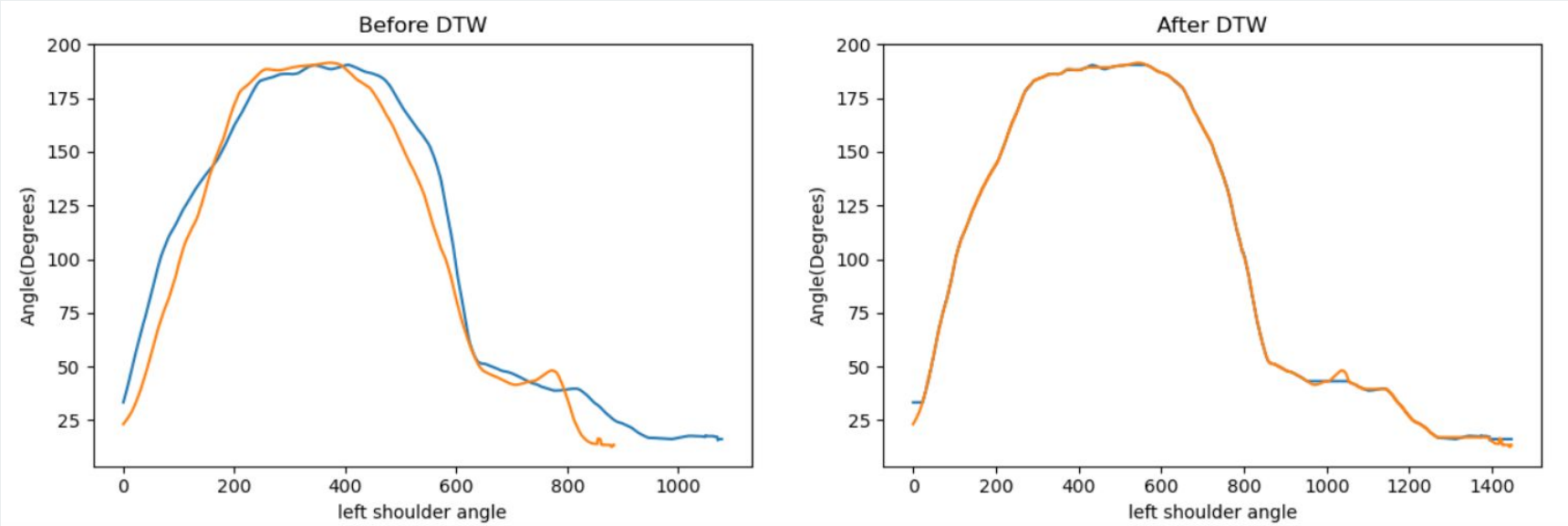
Here we use DTW matrix.

It takes the distance between all the co
points into account and chooses the pa
best alignment.



$$D[i, j] = |xi - yi| + \min(D[i-1, j-1], D[i, j-1], D[i-1, j])$$

DTW ALGORITHM



ML BASED EVALUATION

To remove Discrepancy in the length of Videos, we align the Data-frames w.r.t. each other using a Dynamic Time Warping.

We Calculate the sum of the absolute difference between aligned Dataframes.

Let,

d = Sum of differences previously obtained

N = Number of frames.

We define classification constant ‘C’, such that if

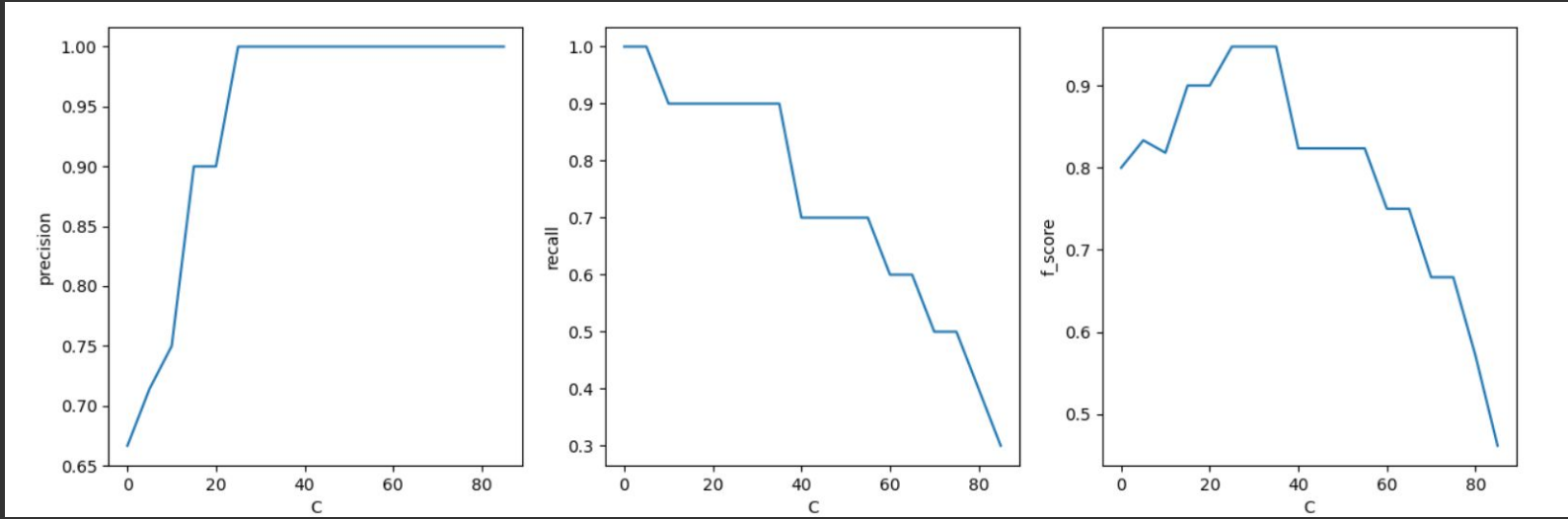
$$\frac{d * 100}{N} > C$$

The variable is considered to be deviated from the reference.

SELECTING VALUE OF ‘C’

We analyse the Training dataset and calculate precision, recall and F1 score for different values of C

We consider the value of C where the F1 score is maximum.



From upper diagram, we can notice that as we increase C, precision increases, However, recall decreases.

Thus, we use F1 score to find out the optimum value of C.

FINAL RESULTS

We test our model on 15 incorrect examples.

Score using sum of Euclidean Distance

Relevant Variables	Precision	Recall	F_score	No. of Examples ('0'-Correct '1'-Incorrect)
l_shoulder_angle	0.666	1.000	0.799	'0'>11, '1'>4
r_shoulder_angle	0.666	1.000	0.799	'0'>11, '1'>4
l_elbow_angle	0.800	0.800	0.800	'0'>10, '1'>5
r_elbow_angle	0.800	0.800	0.800	'0'>10, '1'>5
l_hip_angle	0.600	0.600	0.600	'0'>10, '1'>5
r_hip_angle	0.574	0.666	0.611	'0'>9, '1'>6
l_shoulder_ratio	1.000	1.000	1.000	'0'>11, '1'>4
r_shoulder_ratio	1.000	1.000	1.000	'0'>11, '1'>4
l_elbow_ratio	0.666	0.800	0.726	'0'>10, '1'>5
r_elbow_ratio	0.666	0.800	0.726	'0'>10, '1'>5

FINAL RESULTS

As the lower body remains still during this exercise, instead of using sum of Euclidean distance, we just use the difference of Means.

Score using difference of Mean

Relevant Variables	Precision	Recall	F_score	No. of Examples ('0'-Correct '1'-Incorrect)
l_hip_angle	0.666	0.800	0.726	'0'>10, '1'>5
l_hip_angle	0.714	0.833	0.769	'0'>9, '1'>6

The test took approx. 30 sec. for each variable of every example having around 1000 frames each. Thus it takes approx. 1 second to process 500 frames.

LIMITATIONS

- As our algorithm is processing each and every frame, it is slower as compared to other key-frames.
- Currently, it is only able to process for one person per frame.
- The Datasets were taken in ideal conditions, the performance may vary in real-life usage.
- We need to take different approach for different exercise. So it is not a generalized solution.

Conclusion & Future Work

We have developed the application, Pose Trainer, which calculates the error in the exercise performed concerning the ideal way of performing it. We have used the output of pose estimation to evaluate videos of activities through human pose keypoints and further using machine learning algorithms to determine the correctness of the posture concerning the dataset.

We have further planned to extend this from a yoga exercise to a larger dataset. We are also planning to Use KNN to identify the exercise from the larger Dataset of Yoga Exercises.

Thank You!