DELHI TECHNOLOGICAL UNIVERSITY

COMPUTER SCIENCE DEPARTMENT

Bawana Road, New Delhi, 110042

B. TECH PROJECT-II REPORT

(CO-402)

Yoga Guru: A Study of Pose Estimation and Machine and Deep Learning Methods For Yoga Pose Correction

Proposed by:

Dhananjay Sharma, 2K18/CO/119

Harshil Panwar, 2K18/CO/142

Harshit Goel, 2K18/CO/143

Under the guidance of:

Prof. Rahul Katarya

(supervisor)

DECLARATION

We hereby attest that the work presented as Major Project II, entitled, "**Yoga Guru: A deep learning pipeline for yoga/gym pose estimation and correction**," to satisfy the requirement for the award of the Degree of Bachelor of Technology in **Computer Science and Engineering** and submitted to the department of Computer Science and Engineering is an authentic record of our own work, carried out from January, 2022 to April, 2021, under the supervision of our respected faculty, Prof. Rahul Katarya. The content of this report has not been presented by any of the members of this group for the award of any other degree in any other institute. The work has been submitted in an SCI/SCI Expanded/SSCI/Scopus Indexed Journal or Peer-Reviewed Scopus Indexed Conference with the following details:

Title of Paper: Deep-Learning for Self-Learning in Yoga and Fitness: A Literature Review

Author Names (In order): Dhananjay Sharma, Harshil Panwar, Harshit Goel, Rahul Katarya

Name of Conference/Journal: Springer IICS

Conference Dates with Venue (If Applicable): 17-18 December 2021, Online Mode

Have you registered for the conference: Yes

Status of the Paper: Accepted and Presented

Date of Paper Submission: 20th November, 2021

Date of Paper Acceptance: 5th December, 2021

Date of Paper Presentation: 18 December, 2021

Date of Publication: To be decided.

Dhananjay Sharma 2K18/CO/119	Harshil Panwar 2K18/CO/142	Harshit Goel 2K18/CO/143

DELHI TECHNOLOGICAL UNIVERSITY
Bawana Road, New Delhi, Delhi-110042

SUPERVISOR'S CERTIFICATE

To the best of my knowledge, the above-mentioned work has not been submitted in part or full for any Degree/Diploma to this institute or elsewhere. I further certify that the publication and indexing information given by the student is correct.

Place: DELHI

PROF. RAHUL KATARYA
(SUPERVISOR)

ABSTRACT

With the rising levels of stress in the hustle and bustle of modern life, physical exercise has become imperative in order to maintain a sound body and soul. Yoga is one such art form that has received a lot of approval and has become well sought-after in recent times because of its physical, mental and spiritual benefits. Sports and other gym exercises also provide similar benefits. Moreover, in light of this raging pandemic it has become almost impossible to find a reliable institute to learn such artforms regularly due to the government-imposed lockdown and social distancing norms. Thus, our paper attempts to analyse the significance of modern deep learning techniques in making cost-effective self-learning aids for the aforementioned activities.

Moreover, we have experimented with various potential pipelines and classification techniques on self-scraped databases to develop an end-to-end product capable of successfully classifying the posture, analysing it and generating a feedback score to assist with correction.

ACKNOWLEDGEMENT

We would like to use this opportunity to express our deep sense of gratitude and appreciation to our respected faculty, Professor Rahul Katarya, for his profound guidance, consistent monitoring, motivation and extremely helpful advice throughout the course of this project. We are extremely gratified and indebted to have received such support and mentorship from the teacher and the faculty of the computer science and engineering department. This allowed us to complete our major project and this report, while also enabling us to pursue our interests in the field of research in the correct manner. We are highly obliged to the faculty and the mentors of the computer science and engineering department for indebting us with their time and knowledge. Their immense cooperation throughout the course of this project helped us in numerous ways.

Table of contents

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

ABBREVIATIONS

1. CNN- Convolutional Neural Network
2. LSTM- Long Short-Term Memory
3. RCNN- Region-based convolutional neural network
4. FPN- Feature Pyramid Network
5. ResNet- Residual Neural Network
6. MAE- Mean Absolute Error
7. MSE- Mean Squared Error
8. KNN- K nearest neighbours
9. SVM- Support Vector Machine
10. IOT- Internet of Things
11. WSN- Wireless Sensor Network
12. PCP- Percentage of Correct Parts
13. PDJ- Percentage of Detected Joints
14. PCK- Percentage of Correct Key Points
15. mAP- Mean Average Precision
16. AUC- Area Under the Curve
17. NLP- Natural Language Processing

CHAPTER 1

INTRODUCTION

1.1 BRIEF OVERVIEW

Yoga, calisthenics, sports and minor gym exercises have slowly started finding their way into everyone's daily routines due to their numerous benefits and minimal prerequisites. One major factor that determines the effectiveness of the aforementioned exercises is posture correctness. The correct posture will not only allow the user to reap maximum benefits but it will also prove to be instrumental in preventing injuries. Unfortunately, professional instructors and institutes providing such guidance to beginners are often very expensive and have a very tight schedule due to their popular nature. This paper aims to summarize the role and scope of deep learning in building assistive technology to provide efficient and inexpensive alternatives for the same.

Pose estimation [1] in deep learning is the sub domain associated with analysing and approximating various key-points on a human being in order to estimate the pose. It broadly follows two approaches; the bottom-up approach where each joint is estimated and then connected to form a pose and the top-down approach where a human's bounding box is estimated and then joints are approximated within the region. Pose estimation today forms the base for many other research fields such as human activity recognition, pedestrian analysis for self-driving cars, etc. This paper analyses the use of various pose estimation techniques in building posture correction technology for the various forms of exercise mentioned above.

Nowadays, most end-to-end models in the domain use libraries/hardware such as OpenPose [2], Kinect, etc along with convolutional neural networks (CNN) [3] to detect key-points on a human being. These pose analysis libraries are often paired with LSTMs [4] and/or CNNs to classify the exact pose being performed ("asana" in the case of yoga). In the case of video input, the combination of both proves to work the best as LSTMs help record the temporal relation between the frames while CNNs record the spatial data in every frame thus combining to produce fast and accurate real time results. Once the pose has been classified and the joint locations have been estimated using key points, various mathematical and trigonometric models are employed to perform a comparison between the obtained joint locations and ideal joint locations for a particular pose. This data is then used to generate feedback reports allowing users to fix their posture in real-time.

Thus, by utilizing deep learning techniques, many assistive solutions have been and can be developed to tackle the problem of ensuring posture correctness during exercises. These solutions are not only accurate, they provide real time feedback and are affordable enough to provide a better alternative to posh institutes and personal trainers.

Not only have we thoroughly studied and analysed these pipelines, we have also experimented with and proposed the best ones in our opinion. We use Open Pose, CNNs, Transfer Learning, various classifiers, etc for the same.

1.2.1 PROBLEM STATEMENT

Over the years, most developed countries have recorded a rise in the stress levels of the average population. Moreover, an accompanying increase in average obesity rate and overall decrease in the conditions of health and fitness can be observed. The increasing competitive nature of the modern corporate world and the struggle to survive in this cutthroat environment might have a direct relation to these problems.

Since there seems to be no obvious effort from the government or the corporate sector’s side, everyone must find ways to keep their mental and physical health intact. Yoga, exercising, going to the gym, etc can provide such therapeutic experiences that allow one to cope with daily hustle-bustle both physically and mentally.

As mentioned before, order of exercises, joint angles and body posture can help increase the impact of such exercises a lot. Correct posture can also help prevent injury. Sadly, institutes providing such coaching facilities are often very expensive, not very flexible with schedules and vulnerable to covid related lockdown and social distancing norms. Thus, an alternate way is required.

1.1 MOTIVATION

In today’s digital age, modern facilities like phones, computers, the internet, etc have become more and more accessible. These facilities, if employed correctly, can be used to guide the general population and help them learn and develop new hobbies/routines with ease at no added expenses as well. Moreover, with the recent advancements in Deep Learning under the domain of activity tracking and human pose analysis, it is only natural that an alternative to expensive coaching institutes be developed. An efficient and accurate online pipeline developed with the help of deep learning algorithms capable of analysing a user’s pose and suggesting corrections could help everyone improve their lifestyle significantly. This is what served as the motivating factor and hence, the authors decided to carry out a thorough research study on the progress being made and introducing novelties of their own.

CHAPTER 2

LITERATURE REVIEW

2.1 GENERIC POSE CORRECTION PIPELINE

The general approach that has been adopted by most researchers in the domain comprises two major steps. Firstly, a pose detection algorithm is employed to detect key points on a human being. A pose detection network first localizes human body joints and then group them into valid pose configuration.

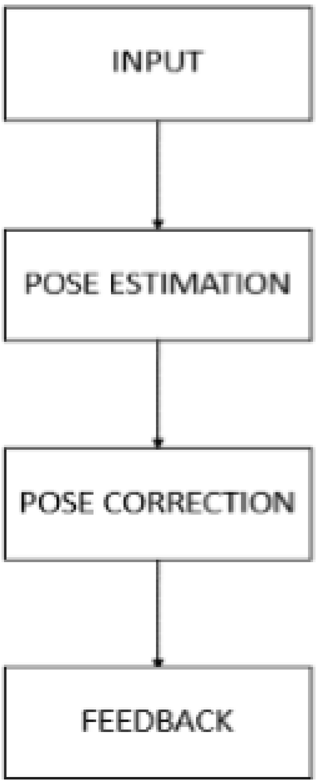


Fig 1. Generic Self-Learning Pipeline

2.1.1 BACKBONE ARCHITECTURE

The backbone architecture used in these networks primarily range from AlexNet [5], to the recently developed ones such as Fast R-CNN [6], Mask R-CNN [7], feature pyramid networks (FPN) [8]. However, VGG [9] and ResNet [10] remain the popular choice.

2.1.2 LOSS FUNCTIONS

Loss functions are at the core of any machine-learning or deep learning model to learn from the dataset. In case of human pose estimation models, Cross-Entropy loss, Mean Absolute Error (MAE), Mean Squared Error (MSE) are most used loss functions. MAE or L1 loss function is measured as mean of absolute error between prediction and true value. Being outlier insensitive, this loss function is more robust.

$$L1 = 1/n\sum |yi - f(xi)|\; n\; i = 1 \tag{2.1}$$

MSE or L2 loss function is mean of squared sum of errors between true and predicted value. This loss function penalizes outliers in a dataset.

$$L2 = 1/n\sum (yi - f(xi))2\; n\; i = 1 \tag{2.2}$$

For a classification model with probability of output between 0 and 1, cross entropy loss is used for measuring performance. Similar to other loss function, cross entropy loss increases for increasing deviation from true value.

$$Logloss = -(yilog(f(xi)) + (1 - yi)log(1 - f(xi))) \tag{2.3}$$

2.1.3 EVALUATION METRICS

For performance comparison of human pose estimation models, several custom evaluation metrics have been developed. These are:

Percentage of Correct Parts (PCP):

It focuses on correct prediction of limbs. A body limb is identified, if distance between two joint positions is less than half in comparison to true value between these key points. Hence due to scaling issues, it penalizes for shorter limbs.

Percentage of Detected Joints (PDJ):

This metric is based on distance between true and predicted joint within a threshold of torso diameter, thus achieving localization precision over former metric.

Percentage of Correct Key-Points (PCK):

It measures if true joint and predicted key point are within a certain threshold. Due to smaller head-bone connections and torsos, it overcomes the shorter body limb problem.

The estimated body joint key point data or the frame itself is then passed through a classifier network that identifies the yoga pose or exercise being performed.

In the second step, the joint body key-points are then used to compute the joint angles using various trigonometric functions. These angle values are then compared to the angle values obtained from the instructor (the ideal values, in this scenario). This comparison helps generate a feedback score and detect flaws in the user's posture, if any. This feedback score is then utilized to generate a report to help users learn by themselves. These approaches have been summarized in table 1.

Table 1. Analysis of existing posture correction pipelines

Authors	Pose Estimation Techniques	Correction Techniques
J. Palanimeera, K. Ponmozhi [11]	KNNs [12], SVMs [13], Logistic Regression [14] and Naive Bayes [15] are used on data set of 12 joints obtained from 17 detected key points.	Only pose classification approaches mentioned.
Munkhjargal Gochoo, Tan-Hsu Tan, Shih-Chia Huang, Tsedevdorj Batjargal, Jun-Wei Hsieh, Fady S. Alnajjar, Yung-Fu Chen [16]	Introduces an IOT based approach using a very deep CNN along with a low-resolution infrared sensor. This infrared sensor is based a wireless sensor network (WSN). WSN is used on the client side to scan the user's movements and then employ a deep CNN to detect the yoga poses.	Only pose classification approaches mentioned.
Maybel Chan Thar, Khine Zar Ne Winn, Nobuo Funabiki [17]	Open Pose is used along with a CNN to detect yoga poses accurately.	Ideal pose angle values are compared with estimated joint angles The output pose is displayed using a greed-red gradient to display degree of correctness in posture
Fazil Rishan, Binali De Silva, Sasmini Alawathugoda, Shakeel Nijabdeen, Lakmal Rupasinghe, Chethana Liyanapathirana [18]	Open pose and mask-RCNN are used as key point estimators in video frames. Time distributed CNNs are used to capture spatial features. An LSTM is then used to capture temporal and spatial changes based on the features. A SoftMax layer then uses these changes to display final pose probabilities.	Only pose classification approaches mentioned.

Manisha Verma, Sudhakar Kumawat, Yuta Nakashima, Shanmuganathan Raman [19]	A novel fine grain hierarchical classification method based on visual similarity of poses is used. Many famous CNN architectures along with a modified dense net 201 with hierarchical connections is proposed. Increased classification accuracy and exemplar performance is observed on an 82-pose dataset.	Only pose classification approaches mentioned.
Rehnhai Huang, Jiqing Wang, Haowei Lou, Haodang Lu, Bofei Wang [20]	Kinect combined with Open Pose is used to improve upon previous results.	Joint angles are calculated by finding the tan inverse of the value of the vector difference between the two vectors joining to make a joint. These angles are used for comparison with ideal values.
Ian Gregory, Samuel Mahatmaputra Tedjojuwono [21]	Open pose is used to detect 18 key points from multiple angles.	Cosine-distance formula is used to obtain joint angles. The difference between the user's joint angles and instructor's joint angles are then used to provide a feedback score.
Edwin W. Trejo, Peijiang Yuan [22]	Kinect v2 is paired with the adaboost algorithm to implement a yoga pose classifier for 6 poses for up to 6 people. Adaboost selects the final dataset for training in order to obtain max accuracy.	Uses joint angle comparisons based on joint coordinates obtained with the help of Kinect v2.
Grandel Dsouza, Deepak Maurya, Anoop Patel [23]	CNN trained on different body part images is used.	Comparison of shoulder and joint angles with that of an athlete's are displayed on a graph over time.
Yuxin Hou, Hongxun Yao, Haoran Li, Xiaoshuai Sun [24]	Convolutional Pose Machine [25] and Faster RCNN [26] are used to detect a person and map their position.	An action guided network is used to rate how accurately a pose is being performed.
Amit Nagarkoti, Revant Teotia, Amith K. Mahale, Pankaj K. Das [27]	Real time two dimensional multi person Pose Estimation is performed. The proposed approach is based on Part Affinity Fields.	For comparison, Dynamic Time Warping (DTW) between each frame of the user and trainer is done. For output feedback used affine transformations to solve camera problems.

Jianbo Wang, Kai Qiu, Houwen Peng, Jianlong Fu, and Jianke Zhu [28]	Region based networks are used to detect humans in an image and isolate them. A custom single person tracking algorithm based off of Siamese tracking [29] is used. A custom pose estimation algorithm is built to analyze both spatial and temporal key points.	Based on the features obtained, a classifier identifies all bad poses so athletes can work on their posture.

In the above table, the paper employing a novel IOT method in order to reduce privacy related issues associated with the use of Kinect and other camera devices makes stellar progress in the self-learning and posture correction domain. It averages an F1 score of 0.93 along all axes and an accuracy of 99.45 percent. The paper conducting a comparison study between open pose and mask-Rcnn as pose analysis modules also achieves excellent results. Open pose as the pose module gives an accuracy of 99.91 percent whereas the mask RCNN algorithm surpasses that result and achieves a 99.96 percent accuracy on the test dataset. The paper employing the convolutional pose machine and faster RCNN also achieved a mean accuracy precision value of 99.9 percent. Lastly the paper using regional based networks also improves upon its predecessor on the same dataset by achieving an F1 score of 0.71. As other papers in table 1 have showcased their results by using image displays and predicted feedback scores of their pipeline, it is hard to summarize the results by using mathematical metrics. Thus, it can be seen despite their being quite a few interesting pipelines already in play, there are a plethora of sub domains where one could expand upon.

2.2 UNTAPPED POSE ESTIMATION ARCHITECTURES

Further, we have also analysed recent advancements in the field of pose detection. These cutting-edge algorithms, while exceptional in analysing the human pose, have not yet been utilized for self-learning and posture correction in the field of exercise. The appropriate implementation of these angles would enable such exercise-oriented pipelines to be much faster. The new state of the art pipelines would also be able to detect humans and poses more accurately, thus, generating a much more relevant feedback report than their predecessors. These new algorithms and networks have been summarized in table below:

Table 2. Comparison of Pose Estimation Algorithms

Model	Year	Backbone Architecture	Approach	Loss Function	Evaluation Metrics	Training Datasets
DeepPose [30]	2014	AlexNet	Top-down	L2	PDJ, PCP	LSP, FLIC
DeeperCut [31]	2016	ResNet	Bottom-Up	L1, Cross-Entropy	AP, mAP, AUC, PCKh@0.5	MPII, COCO, LSP
Convolutional Pose Machines	2016	VGG	Top-down	L2	PCKh@0.1,PCKh@0.2, PCKh@0.5	FLIC, LSP, MPII
Rmpe: Regional Multi-Person Pose Estimation [32]	2017	VGG, ResNet	Top-down	L2	mAP	MPII, COCO
DensePose [33]	2018	FCN, Mask-RCNN	Bottom-up	Cross-Entropy	AP, PCP, GPS	COCO
High-Resolution Network (HRNet) [34]	2019	ResNet	Bottom-up	L2	AP, mAP, PCKh@0.5	MPII, COCO
Human Pose Estimation for Real-World Crowded Scenarios [35]	2019	ResNet	Top-down	L2	AP, OKS	CrowdPose, JTA

For each recent human pose estimation model, parameters such as backbone architecture, loss function have been weighed in Table 2. The majority of models opt for Res-Net as the backbone architecture due to its ability to negate the vanishing gradient problem, hence providing a deeper model with greater accuracy. Apart from Average Precision (AP), Mean Average Precision (mAP) many of the models prefer PCKh@0.5 as a good evaluation metric since it directly deals with human images. For training purposes, apart from the standard cross-entropy loss, most models are shifting towards Least Square Errors(L2) even though it is sensitive towards outliers. Last but not the least, in these recent works COCO and MPII have been, by far, default choices of Dataset with richer training data. Hence, from table 2, it is quite clear that there are numerous network architectures that have not been employed for the purpose of real time pose detection and correction mechanisms. Doing so would allow one to create new and improved state of the art pipelines, thus, revolutionising the domain of self-help in yoga and fitness.

CHAPTER 3

METHODOLOGY

For the development of an end-to-end pipeline and a finished product we chose two broad avenues. Firstly, we built a heavy-duty product where the classification accuracy surpasses previously attained benchmarks. Secondly, we focused on a light weight network where we compromised on the classification accuracy but developed a faster and cheaper alternative. Even so, the light weight network performs exceedingly well on the relatively clearer pictures in our database

3.1 Dataset

Our dataset comprises of 5 classes as of now, each representing a different yoga pose. There are about 2000 images in total. The database has been scraped, analysed and refined by the members of this group for the purpose of this project.

To analyse alternate training routes, two other databases were developed, databse-black and database-openpose, Database-Black contains the estimated human skeletons of all images on a black background. Database-OpenPose contains the results of Database-Black super imposed on the original database.

Another approach that we developed was identifying the joint-coordinates on a human body and then calculating the joint angles using custom python scripts. A fourth and final database was used for storing this.

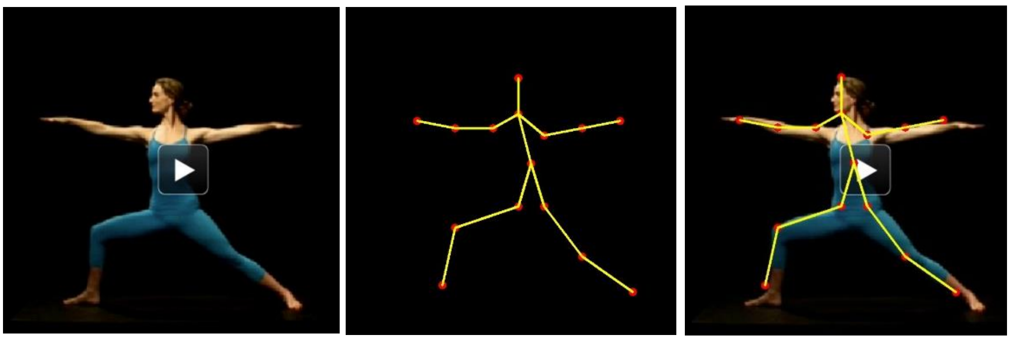


Fig 2. Sample Image from All 3 Image datasets

3.2 Yoga Guru (Full-Scale)

For the full-scale version of our pipeline, we utilized a VGG-16 pre-trained on the Imagenet dataset. A unique thing about VGG-16 is that it follows the same dimensions of convolutional and pooling layers across the depth of the architecture. We incorporated a custom classifier and trained a model capable of identifying the yoga pose being performed. The model was trained on all three of the image-based datasets mentioned above. Based on the speed and accuracy of each model, we opted to select the plain database for our final product.

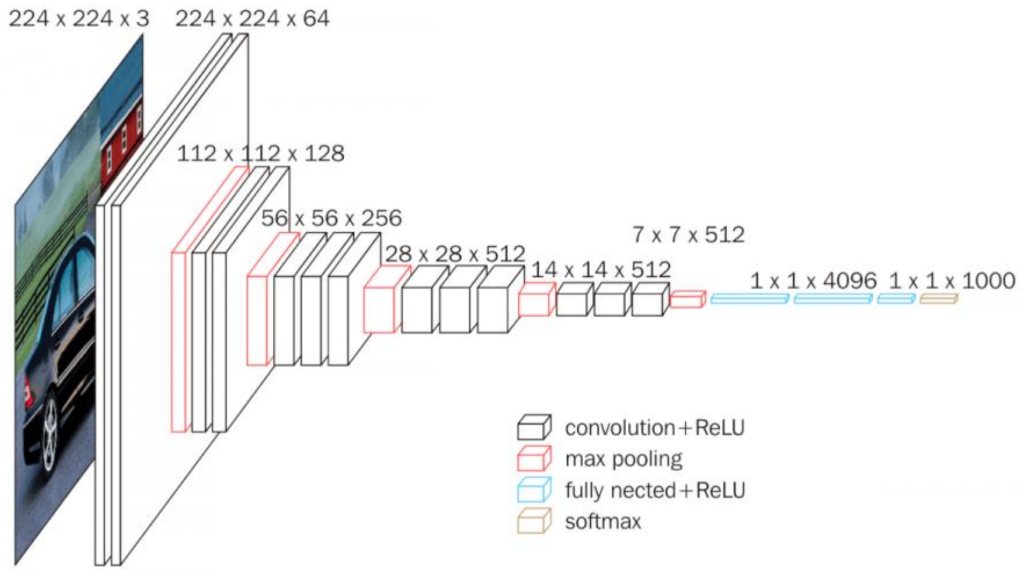


Fig.3 VGG – 16 Feature Extractor Architecture

Once the model accurately identifies the pose being performed, pose estimation is used on the image to locate joint keypoints. For this purpose, OpenPose has been employed, which is advantageous in real-time conditions. It can detect 2-D as well as 3-D multi-person poses, also in crowded scenes.

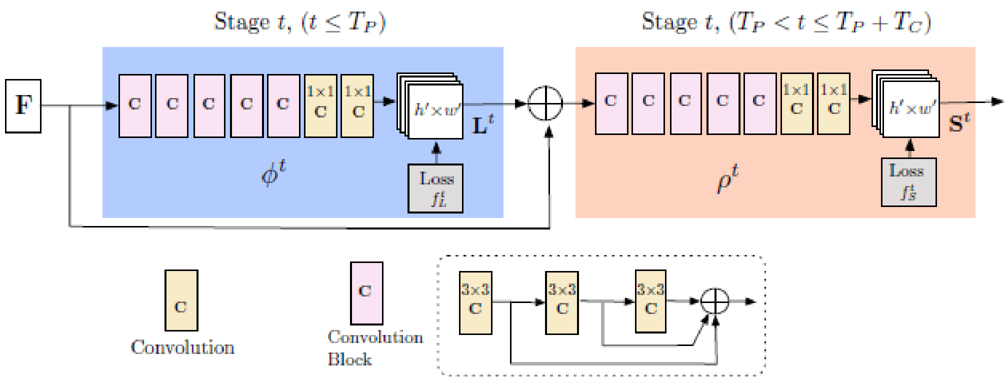


Fig.4 OpenPose Multi-Stage CNN Architecture

Our work uses pre-trained OpenPose on MPII dataset, which outputs a total of 15 keypoints corresponding to different joints of body. These joint keypoints and calculated body joint angles are then used to compare the user's pose with an ideal pose already stored in our database. This comparison is used to generate a feedback score for the same.

Table 3. Estimating joint angles based on OpenPose indexing

Angle Name	Body Parts			MPII Label Index
elbow-r	Right Shoulder	Right Elbow	Right Wrist	[2,3,4]
elbow-l	Left Shoulder	Left Elbow	Left Wrist	[5,6,7]
knee-r	Right Hip	Right Knee	Right Ankle	[8,9,10]
knee-l	Left Hip	Left Knee	Left Ankle	[11,12,13]
neck-r	Head	Neck	Right Shoulder	[0,1,2]
neck-l	Head	Neck	Left Shoulder	[0,1,5]
leg-r	Chest	Right Hip	Right Knee	[14,8,9]
leg-l	Chest	Left Hip	Left Knee	[14,11,12]
back	Head	Neck	Chest	[0,1,14]

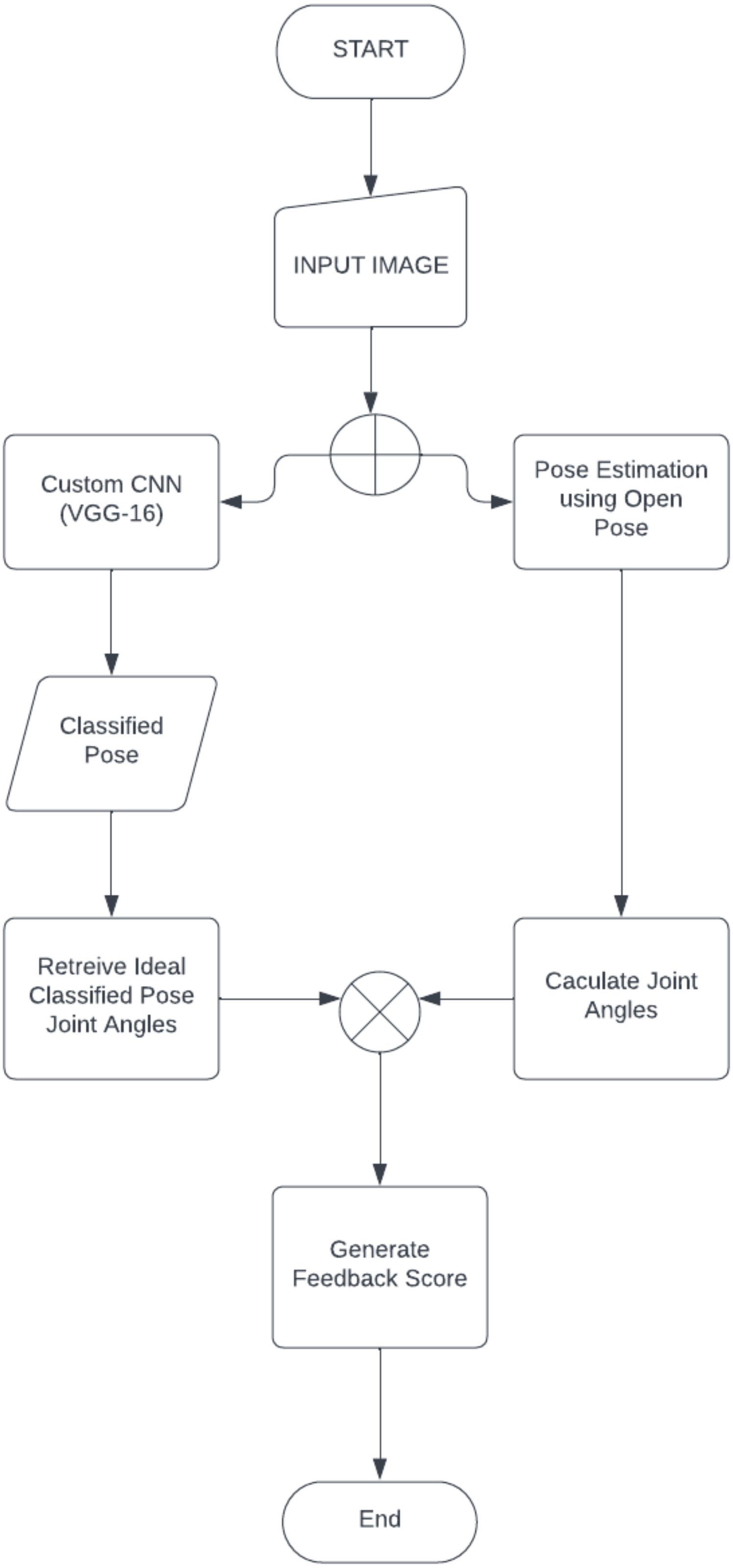


Fig 5. Yoga Guru Full Scale Pipeline

3.3 Yoga Guru (Light Weight)

For the light weight version of our pipeline, we trained numerous classifiers on our fourth dataset, the one comprising of angle values. We found that the random forest classifier performed best given the lack of obvious trends present in our dataset. None the less, we built a complete product keeping this model at the centre.This time, the input image is sent to the pose estimator right away, the joint key points are obtained and the body angles are calculated. These angle values are fed into the trained classifier and the output class is obtained. Similar to the previous network, this output class and angle values are then used to generate a feedback score by comparing with the ideal pose values.

The key differentiator here is the size and speed of the classifier. Trained random forest weights require 10 times lesser space than the VGG16 based classifier, thus while the model may compromise on accuracy its size and speed makes it extremely portable for mobile application and web application development.

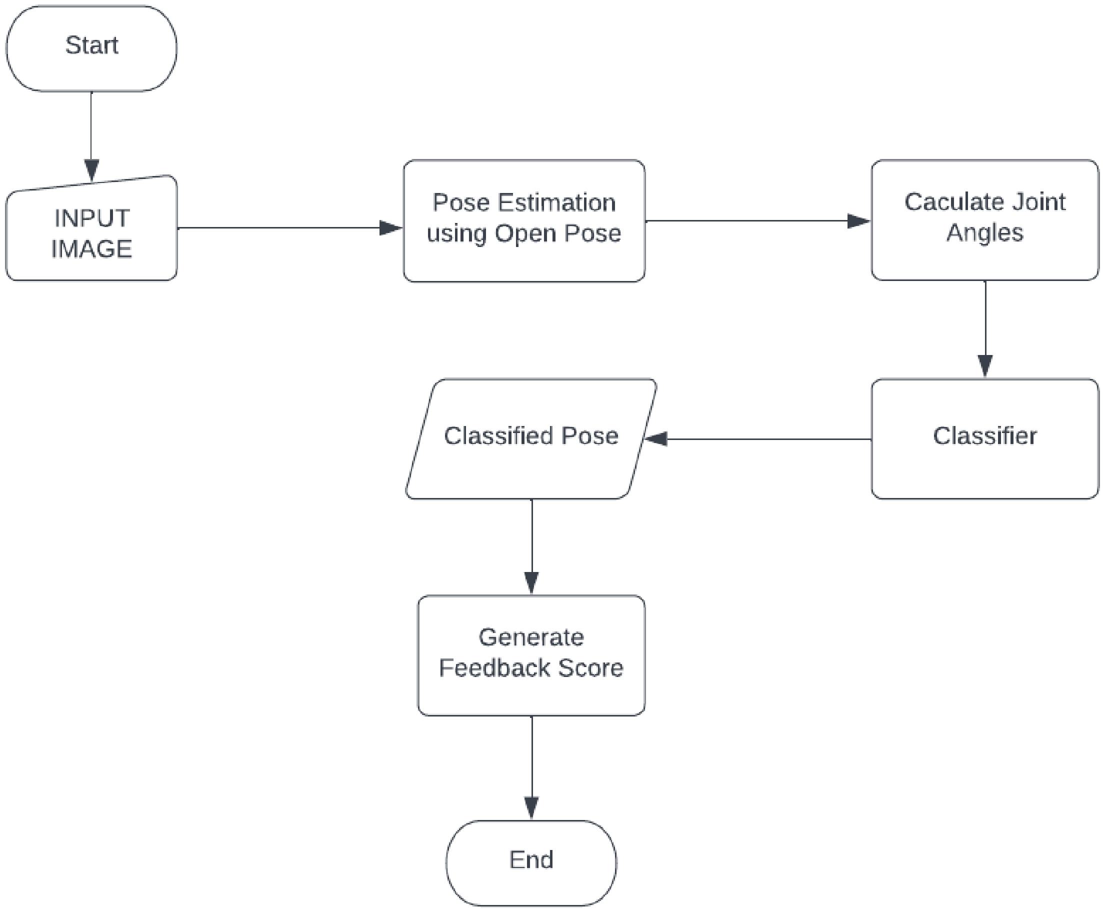


Fig 6. Yoga Guru Light Weight Pipeline

CHAPTER 4

EXPERIMENTATION AND RESULTS

4.1 Technical Specifications

1. Operating System: Windows 10, Home Edition
2. Coding Language: Python 3.x
3. IDE and Platforms: Jupyter Notebook and Google Colab
4. Libraries and Packages:
    a. Numpy: A python library capable of handling and performing complex mathematical operations on multi-dimensional arrays.
    b. Open CV: A python library useful in opening, viewing, saving and manipulating images.
    c. Tensorflow 2: A python framework allowing users to create multi-layer perceptron and import pre-loaded models with ease.
    d. Flask: A python module that helps develop web applications to demo python backends with ease.

4.2 Yoga Guru Full Scale Results

As mentioned above, for training the full-scale model, 3 databases were used. A CPU compatible version of the various python libraries was implemented on an Intel I5 7th gen chipset. RMS Prop was used as the optimizer and categorical cross entropy was used as the loss function. The training function was run for 100 epochs on a batch size of 20. The limited number of runs were set due to the small dataset size. The results are shown below:

Table 4. Dataset wise accuracy for VGG16 based transfer learning model.

Datasets	Train Accuracy	Test Accuracy
Plain Images	98.32%	93.56%
Pose Estimation on black background	85.21%	56.44%
Super imposed Images	98.06%	91.47%

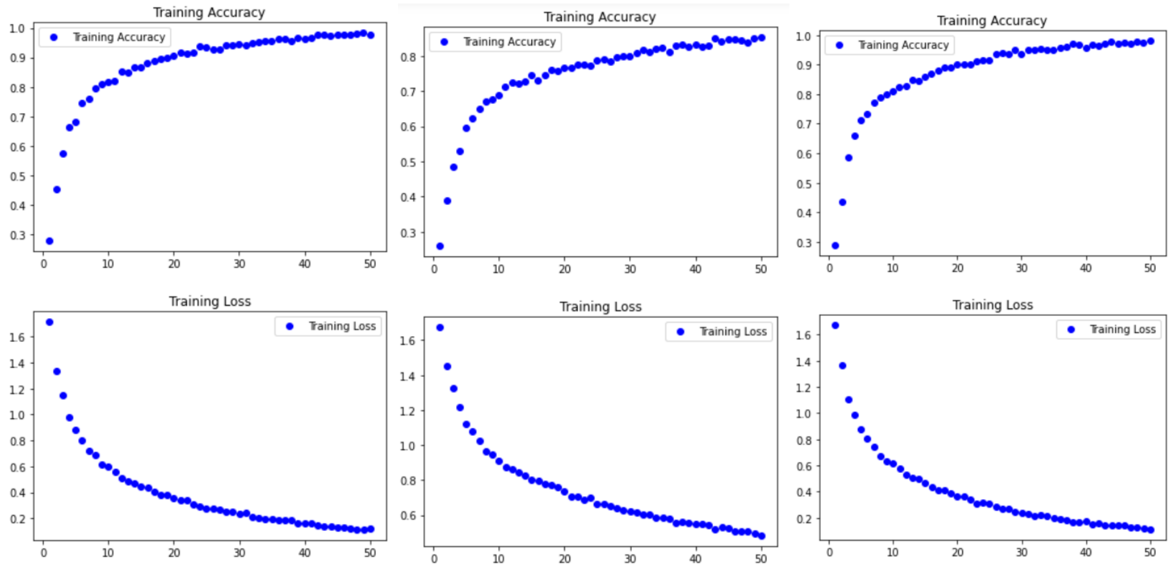


Fig 7. Training accuracy and training loss for the three datasets in order of table

As we can see from the afore-mentioned results, our best choice in terms of test results appeared to be the base model.

4.3 Yoga Guru Light Weight Results

As mentioned above, for training the light weight model, the numerical angle-based dataset was used. A CPU compatible version of the various python libraries was implemented on an Intel I5 7^{th} gen chipset. Various classifiers including KNNs, SVMs, Decision Trees, Random forests and ANNs were used.

4.3.1 Result of KNN Classifier

As the raw dataset yielded poor accuracy, we removed the null values from our dataset, dealt with outliers, normalized the independent variables and ran again. The results achieved this time were quite satisfactory considering the speed and space advantage simpler machine learning models provide. From the figure below, it is quite clear that we proceeded with n neighbours = 1.

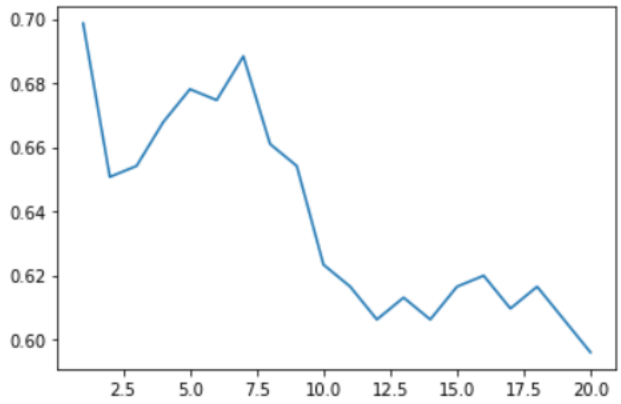


Fig 8. Accuracy score vs N-Neighbours for the KNN model

4.3.2 Result of SVM Classifier

After performing the same data transformations as our previous model, we chose to implement the support vector classifier on the dataset. Here, we recorded the accuracies for the various kernel choices. From the results presented below, we can se that the 'rbf' kernel was our best choice.

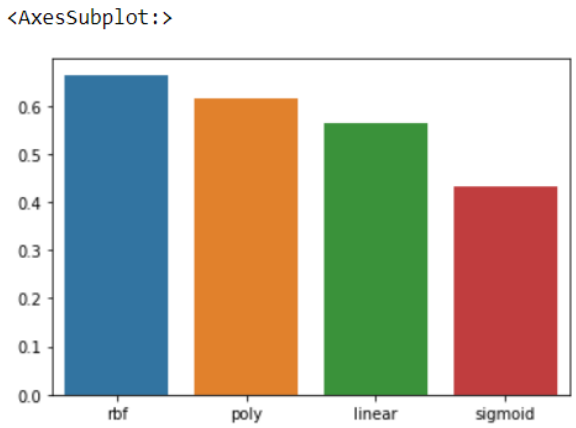


Fig 9. Accuracy v/s choice of kernel in SVM Model

4.3.3 Result of Artificial Neural Network

Once again, using the cleaned dataset for improved accuracy, we implemented various artificial neural networks. We changed the model structure, epochs, and loss functions. Finally, we found that the ADAM optimizer works best for the task at hand. From the various combinations of results given below, we chose the 3rd and 4th model as it gives the best test accuracy. The models after it move towards extreme overfitting.

Table 4. ANN Structure wise accuracy comparison

Model Structure	Train Accuracy	Test Accuracy
ReLu(16,12), Softmax(5)	73%	64%
ReLu(16,12, 8), Softmax(5)	78%	64%
ReLu(20, 16,12, 8), Softmax(5)	85%	64%
ReLu(20, 20, 16,12, 8), Softmax(5)	92%	66%
ReLu(24, 20, 20, 16,12, 8), Softmax(5)	97%	65%

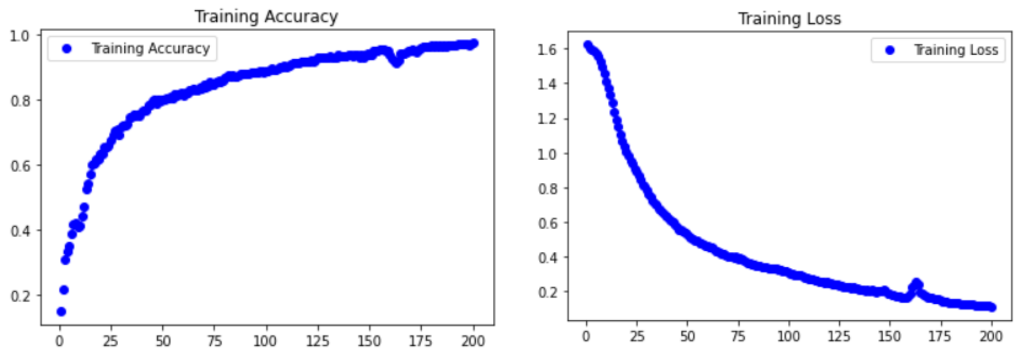


Fig 10. Training Accuracy v/s Loss for chosen ANN

4.3.4 Result of Decision Tree Classifier

The decision tree algorithm performed better on the raw dataset. As this provides additional robustness to the model and prepares it for abnormal test entries we proceeded with the raw model. As we can see, the tree stagnates after max depth = 15.

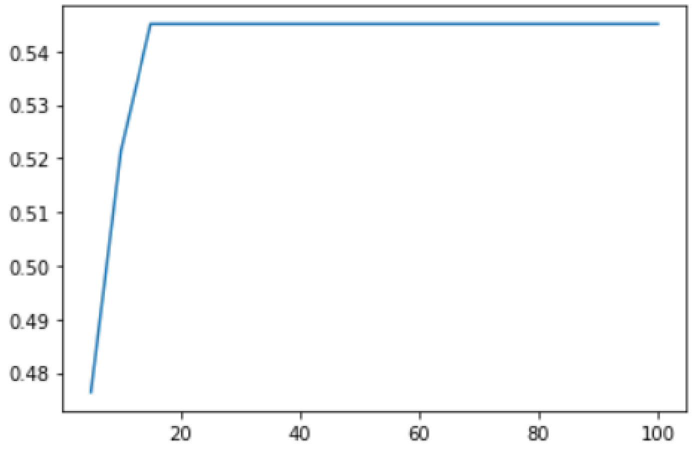


Fig 11. Accuracy v/s depth for decision tree model

4.3.5 Result of Random Forest Classifier

The random forest algorithm also worked the best on raw data. Once again, this scenario was preferred as it increased the model's robustness and performance against unseen data. As we can see below, the model peaks out at about 71 percent accuracy and thus this is the best choice from our models mentioned. Also, this seems to be the best choice overall for building our light weight pipeline.

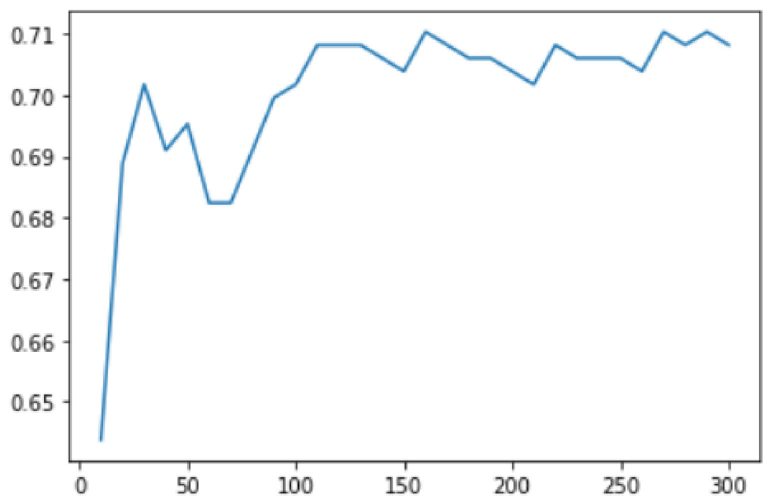


Fig 12. Accuracy score v/s number of estimators for the Random Forest model

4.4 Summarized Result

The table below showcases the top model choice for each algorithm achieved by tuning the hyperparameters. From the table, it is quite clear, we chose the random forest classifier.

Table 5. Comparison of classifier results

Algorithm	Best Test Accuracy
KNN	69.86%
SVM	66.43%
ANN	66.78%
Decision Tree	54.50%
Random Forest	71.03%

CHAPTER 5

DEPLOYMENT

1.1 General Flow

For the purpose of creating a proof of concept, we created a flask-based app. The app comprises of a home page allowing the user to select whether they want to pick the lite version or the full-scale version.

After determining the version, they simply need to upload a photo of them performing a yoga pose. Once uploaded, the pipeline will predict the pose and generate a feedback score for the same. This will be visible on the output screen.

The user may simply try again after as many times as they please.

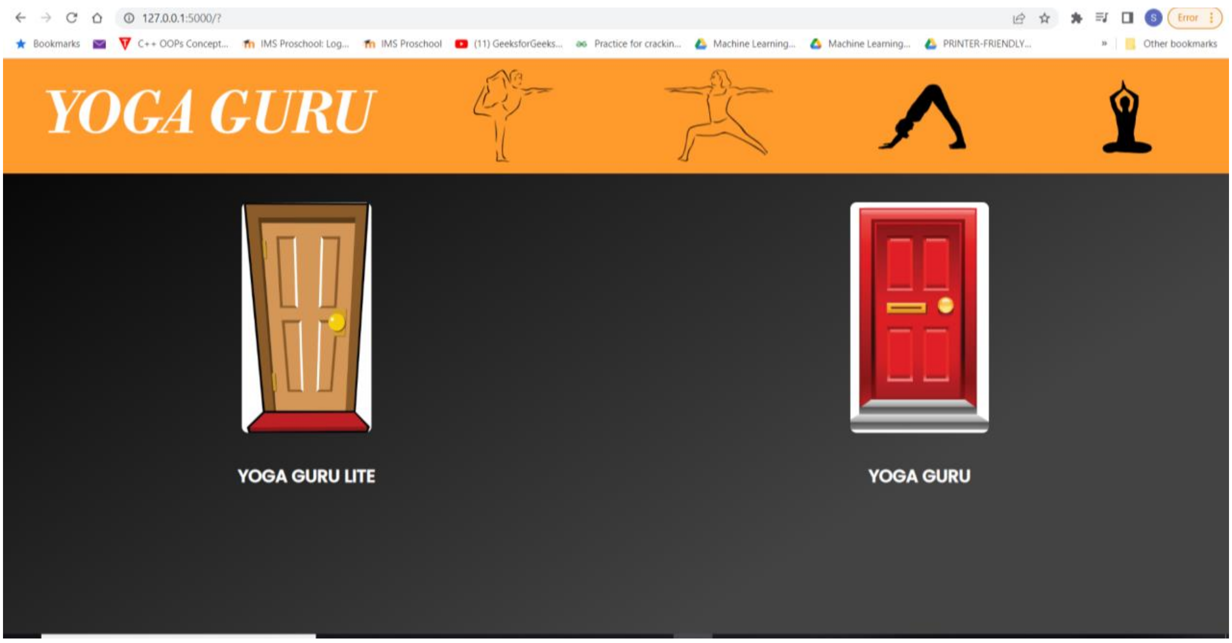


Fig 13. Home Page for Yoga Guru web application

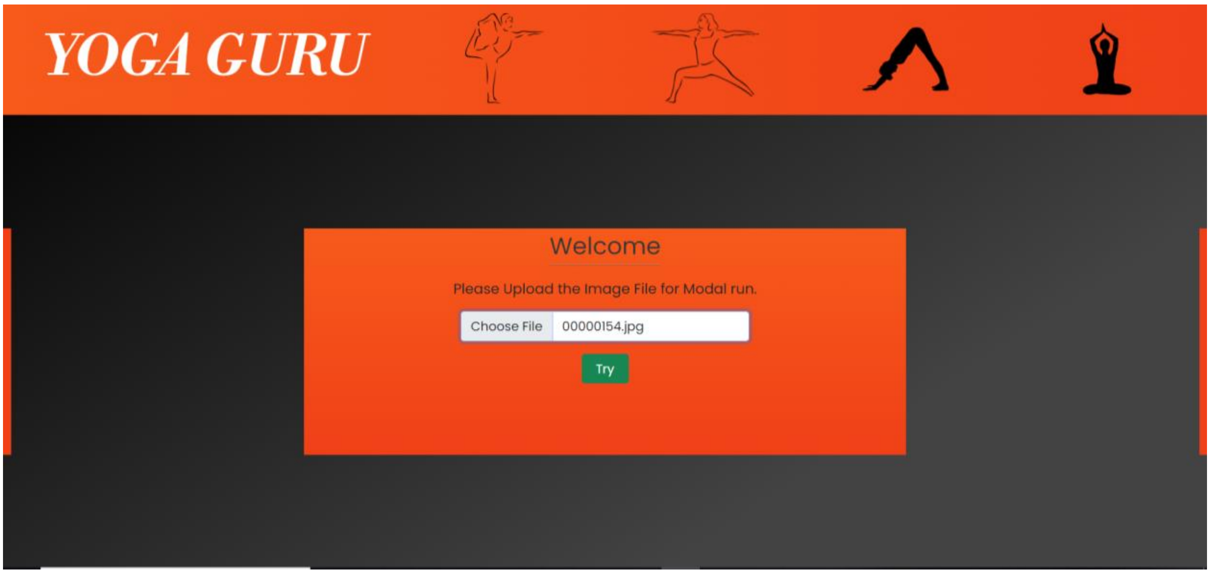
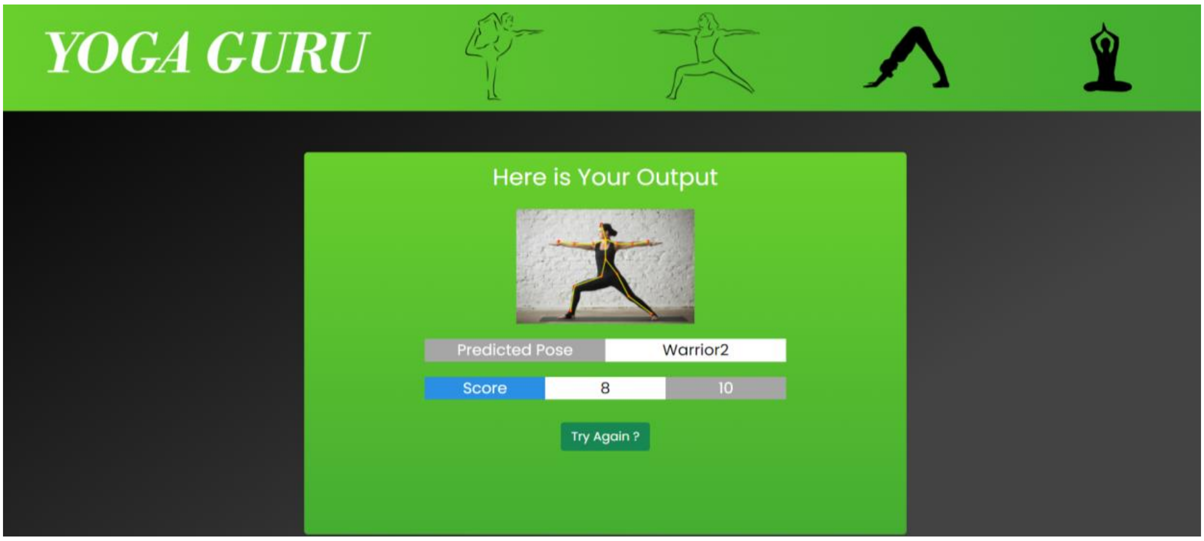


Fig 14. Input Screen for Yoga Guru web application



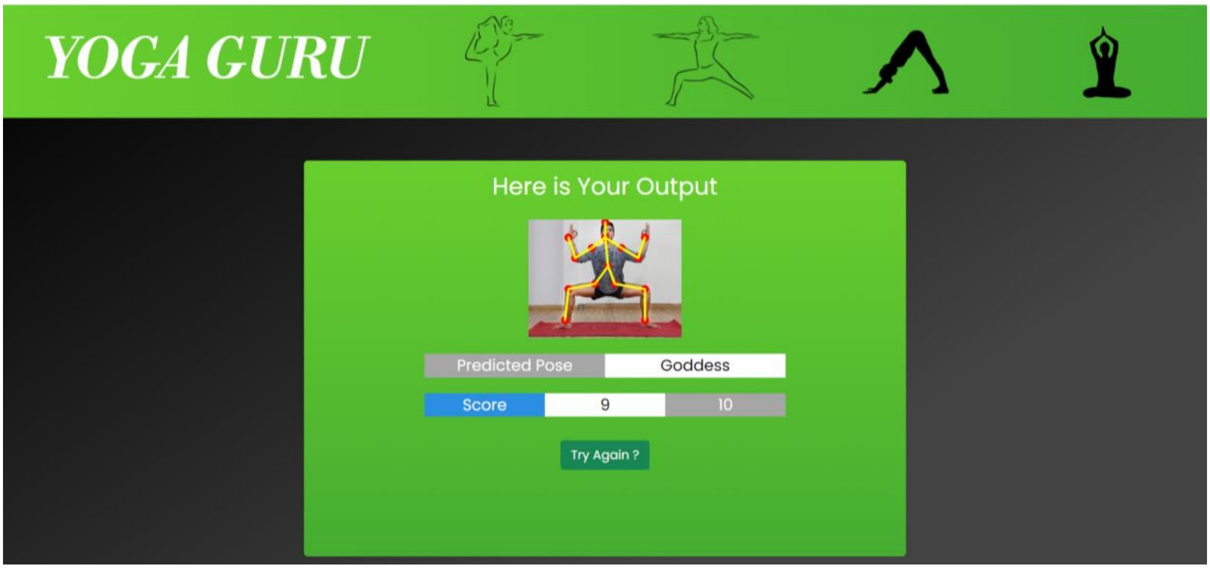


Fig 15. Output page of yoga guru web application

CHAPTER 6
CONCLUSION AND FUTURE WORK

6.1 CONCLUSION

Thus, this work manages to highlight the increasing importance of yoga, calisthenics, gyms and general fitness in the daily life of an average person. Furthermore, this work neatly summarizes the relevant works and benchmarks that have been set by past researchers in the literature review section. Moreover, we manage to experiment with multiple classification and pose estimation algorithms and develop end to end pipelines capable of predicting the yoga pose being performed by a user and suggesting performance scores accordingly.

Also, this work manages to summarize all the cutting-edge pose detection networks that have been created in the past few years. These networks, while much better than their predecessors, have not yet been applied in the field of posture correction and self-learning and posture correction in exercise and fitness.

Lastly the paper also breaks down the pipeline in a systematic format and manages to point out the scope for improvement and improvisation along each step of the pipeline for future researchers to pick up on.

6.2 FUTURE WORK

As far as future scope in the domain is concerned, there is potential for the introduction of numerous improvements and improvisations along every step of the pipeline. Firstly, a novel IOT method was proposed to capture user input in the place of a traditional camera or a Kinect to tackle privacy issues that their predecessors were not even aware of. Further improving this approach or finding better and safer methods is one track future authors could research along.

Pose detection networks, while extremely accurate now, still have to overcome many challenges. To this day, multi person pose detection, dynamic activity tracking etc. have not yet reached their saturation points. Thus, improving upon this part of the pipeline is also a track researchers could entertain.

Moreover, the aforementioned pose analysis networks have successfully surpassed their predecessors in terms of accuracy, efficiency and raw speed. Still, these networks have not yet been tested in the domain of self-learning and posture correction for exercise and fitness. Thus, by utilizing the potential of these models, future researchers could help make accurate and truly real time posture classification and analysis possible.

Adding the capability to deal with video inputs might make this pipeline a viable solution for an actual real-world problem. Using LSTMs, or simply creating a probability threshold on every frame might help the algorithm predict which frames to consider and provide feedback on.

Also, by using Natural Language Processing (NLP), rather than tree-based hard coded correction output mechanisms the pipeline could be perfected in a more end to end manner. The use of NLP could allow the model to give a personalized and more relevant feedback instead of relying on a few pre-coded statements. This will help create impactful products that the end users could truly benefit from.

REFERENCES

[1] T. L. Munea, Y. Z. Jembre, H. T. Weldegebriel, L. Chen, C. Huang and C. Yang, "The Progress of Human Pose Estimation: A Survey and Taxonomy of Models Applied in 2D Human Pose Estimation," in IEEE Access, vol. 8, pp. 133330-133348, 2020, doi: 10.1109/ACCESS.2020.3010248.

[2] 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. 172-186, 1 Jan. 2021, doi: 10.1109/TPAMI.2019.2929257.

[3] Yamashita, R., Nishio, M., Do, R.K.G. et al. Convolutional neural networks: an overview and application in radiology. Insights Imaging 9, 611–629 (2018).

[4] S. Hochreiter και J. Schmidhuber, 'Long Short-term Memory', *Neural computation*, τ. 9, σσ. 1735–1780, 12 1997.

[5] A. Krizhevsky, I. Sutskever, και G. E. Hinton, 'ImageNet Classification with Deep Convolutional Neural Networks', στο *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, και K. Q. Weinberger, Επιμ. Curran Associates, Inc., 2012, σσ. 1097–1105.

[6] R. Girshick, "Fast R-CNN," 2015 IEEE International Conference on Computer Vision (ICCV), 2015, pp. 1440-1448, doi: 10.1109/ICCV.2015.169.

[7] K. He, G. Gkioxari, P. Dollár and R. Girshick, "Mask R-CNN," 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 2980-2988, doi: 10.1109/ICCV.2017.322.

[8] T. J. Watson, 'An empirical study of the naive Bayes classifier', 2001

[9] T. -Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan and S. Belongie, "Feature Pyramid Networks for Object Detection," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 936-944, doi: 10.1109/CVPR.2017.106.

[10] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.

[11] J. Palanimeera, K. Ponmozhi, Classification of yoga pose using machine learning techniques, Materials Today: Proceedings, Volume 37, Part 2, 2021, Pages 2930-2933, ISSN 2214-7853

[12] Guo, Gongde & Wang, Hui & Bell, David & Bi, Yaxin.. KNN Model-Based Approach in Classification.(2004)

[13] T. Evgeniou και M. Pontil, 'Support Vector Machines: Theory and Applications', στο *Machine Learning and Its Applications*, 2001.

[14] C.-Y. J. Peng, K. Lee, και G. M. Ingersoll, 'An Introduction to Logistic Regression Analysis and Reporting', *The Journal of Educational Research*, τ. 96, σσ. 14–13, 2002.

[15] T. J. Watson, 'An empirical study of the naive Bayes classifier', 2001.

[16] M. Gochoo et al., "Novel IoT-Based Privacy-Preserving Yoga Posture Recognition System Using Low-Resolution Infrared Sensors and Deep Learning," in IEEE Internet of Things Journal, vol. 6, no. 4, pp. 7192-7200, Aug. 2019, doi: 10.1109/JIOT.2019.2915095.

[17] M. C. Thar, K. Z. N. Winn and N. Funabiki, "A Proposal of Yoga Pose Assessment Method Using Pose Detection for Self-Learning," 2019 International Conference on Advanced Information Technologies (ICAIT), 2019, pp. 137-142, doi: 10.1109/AITC.2019.8920892.

[18] F. Rishan, B. De Silva, S. Alawathugoda, S. Nijabdeen, L. Rupasinghe and C. Liyanapathirana, "Infinity Yoga Tutor: Yoga Posture Detection and Correction System," 2020 5th International Conference on Information Technology Research (ICITR), 2020, pp. 1-6, doi: 10.1109/ICITR51448.2020.9310832.

[19] M. Verma, S. Kumawat, Y. Nakashima and S. Raman, "Yoga-82: A New Dataset for Finegrained Classification of Human Poses," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2020, pp. 4472-4479, doi: 10.1109/CVPRW50498.2020.00527.

[20] R. Huang, J. Wang, H. Lou, H. Lu and B. Wang, "Miss Yoga: A Yoga Assistant Mobile Application Based on Keypoint Detection," 2020 Digital Image Computing: Techniques and Applications (DICTA), 2020, pp. 1-3, doi: 10.1109/DICTA51227.2020.9363384.

[21] I. Gregory and S. M. Tedjojuwono, "Implementation of Computer Vision in Detecting Human Poses," 2020 International Conference on Information Management and Technology (ICIMTech), 2020, pp. 271-276, doi: 10.1109/ICIMTech50083.2020.9211145.

[22] E. W. Trejo and P. Yuan, "Recognition of Yoga Poses Through an Interactive System with Kinect Device," 2018 2nd International Conference on Robotics and Automation Sciences (ICRAS), 2018, pp. 1-5, doi: 10.1109/ICRAS.2018.8443267.

[23] G. Dsouza, D. Maurya and A. Patel, "Smart gym trainer using Human pose estimation," 2020 IEEE International Conference for Innovation in Technology (INOCON), 2020, pp. 1-4, doi: 10.1109/INOCON50539.2020.9298212.

[24] Y. Hou, H. Yao, H. Li and X. Sun, "Dancing like a superstar: Action guidance based on pose estimation and conditional pose alignment," 2017 IEEE International Conference on Image Processing (ICIP), 2017, pp. 1312-1316, doi: 10.1109/ICIP.2017.8296494.

[25] S. Wei, V. Ramakrishna, T. Kanade and Y. Sheikh, "Convolutional Pose Machines," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 4724-4732, doi: 10.1109/CVPR.2016.511.

[26] Ren, Shaoqing & He, Kaiming & Girshick, Ross & Sun, Jian. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence. 39. 10.1109/TPAMI.2016.2577031.

[27] A. Nagarkoti, R. Teotia, A. K. Mahale and P. K. Das, "Realtime Indoor Workout Analysis Using Machine Learning & Computer Vision," 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 2019, pp. 1440-1443, doi: 10.1109/EMBC.2019.8856547.

[28] Jianbo Wang, Kai Oiu Houwen Peng, Jianlong Fu, and Jianke Zhu. 2019. AI Coach: Deep Human Pose Estimation and Analysis for Personalized Athletic Training Assistance. In Proceedings of the 27th ACM International Conference on Multimedia (MM ’19).

[29] Bertinetto, Luca & Valmadre, Jack & Henriques, Jao & Vedaldi, Andrea & Torr, Philip. (2016). Fully-Convolutional Siamese Networks for Object Tracking. 9914. 850-865. 10.1007/978-3-319-48881-3_56.

[30] A. Toshey and C. Szegedy, “DeepPose: Human Pose Estimation via Deep Neural Networks,” 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp, 1653-1660, doi: 10.1109/CVPR.2014.214.

[31] Insafutdinov, Eldar & Pishchulin, Leonid & Andres, Bioern & Andriluka, Mykhaylo & Schiele, Bernt. (2016). DeeperCut: A Deeper, Stronger, and Faster Multi-person Pose Estimation Model. 9910. 34-50. 10.1007/978-3-319-46466-4_3.

[32] Fang, Hao-Shu & Xie, Shuqin & Tai, Yu-Wing & Lu, Cewu (2017). RMPE: Regional Multi-person Pose Estimation. 2353-2362. 10.1109/ICCV.2017.256.

[33] R. A. Güler, N. Neverova and I. Kokkinos, “DensePose: Dense Human Pose Estimation in the Wild,” 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 7297-7306, doi: 10.1109/CVPR.2018.00762.

[34] Wang, J., Sun, K., Cheng, T., Jiang, B., Deng, C., Zhao, Y., Liu, D., Mu, Y., Tan, M., Wang, X., Liu, W., & Xiao, B. (2020). Deep High-Resolution Representation Learning for Visual Recognition.

[35] T. Golda, T. Kalb, A. Schumann and J. Beyerer, "Human Pose Estimation for Real-World Crowded Scenarios," 2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), 2019, pp. 1-8, doi: 10.1109/AVSS.2019.8909823.

PROOF OF PRESENTATION IN IICS

IICS2021 submission 139 External Inbox ×

IICS2021 <iics2021@easychair.org>
to me

Dear authors,

We received your submission to IICS2021 (International conference on Intelligent Computing and Security: A Paradigm Shift):

Authors : Dhananjay Sharma, Harshil Panwar, Harshit Goel and Rahul Katarya
Title : Deep-Learning for Self-Learning in Yoga and Fitness: A Literature Review
Number : 139

The submission was uploaded by Dhananjay Sharma <dhananjaysharma_2k18co119@dtu.ac.in>. You can access it via the IICS2021 EasyChair Web page

https://easychair.org/conferences/?conf=iics2021

Thank you for submitting to IICS2021.

Best regards,
EasyChair for IICS2021.

IICS2021 (author)

New Submission | Submission 139 | IICS2021 | Conference | News | EasyChair

My Submissions for IICS2021

Using the submission author environment you can view or manage your submissions to IICS2021. You can make new submissions or update your previous submissions.

To **make a new submission** click on "New Submission".

To **view or update your existing submission** click on the corresponding "view" icon.

Additional information about submission to IICS2021 can be found at the IICS2021 Web page.

Questions about submissions can be sent to the conference contact email sanjeev.kumar@glbitm.ac.in.

Please note that if you do nothing (not even click on the menu) for more than two hours, your session will expire and you will have to log in again.

#	Authors	Title	View	paper	Program
139	Dhananjay Sharma, Harshil Panwar, Harshit Goel and Rahul Katarya	Deep-Learning for Self-Learning in Yoga and Fitness: A Literature Review			

GL BAJAJ
Institute of Technology & Management
FIND YOUR SPARK
Approved by AICTE & Affiliated to AKTU

INTERNATIONAL CONFERENCE
on
INTELLIGENT COMPUTING AND SECURITY: A PARADIGM SHIFT (IICS2021)
Organised by
Department of Master of Computer Application
G.L. Bajaj Institute of Technology & Management, Greater Noida
December 17th - 18th, 2021

Springer
Scopus

CALL FOR PAPERS | IMPORTANT DATES | PAPER SUBMISSION | REGISTRATION | SPEAKERS | CONTACT US

- HOME
- ABOUT THE COLLEGE
- PATRONS & CHAIRS
- EDITORIAL BOARD
- ADVISORY COMMITTEE
- ORGANIZING COMMITTEE
- TECHNICAL PROGRAM COMMITTEE

Important Dates

Paper Submission Started:	15 September, 2021
Submission deadline:	15 November, 2021
Notification of acceptance:	5 December, 2021
Submission deadline for revised papers:	10 December, 2021
Date of Conference:	17-18 December, 2021

GL BAJAJ
Institute of Technology & Management
FIND YOUR SPARK
Approved by AICTE & Affiliated to AKTU

Springer

INTERNATIONAL CONFERENCE
on
INTELLIGENT COMPUTING AND SECURITY: A PARADIGM SHIFT (IICS2021)
December 17th - 18th 2021

CERTIFICATE

Certified that Ms./Mr./Dr. *Dhananjay Sharma*

from *Delhi Technological University, Delhi*

has participation / presented paper entitled *Deep-Learning for Self-Learning in Yoga and Fitness: A Literature Review*

in Two Days Springer Sponsored Intelligent Computing and Security: A Paradigm Shift (IICS2021) on December 17th - 18th 2021 *organized by Department of Master of Computer Applications, G.L. Bajaj Institute of Technology & Management Greater Noida, (U.P.) India.*

Convener	Conference Chair	General Chair
Dr. Sanjeev Kumar	Prof. (Dr.) Madhu Sharma Gaur	Prof (Dr.) Rajeev Agrawal