MediaPipe: Yoga Pose Detection Using Deep Learning Models

Sweety Patel[1,2, a)] and Dr. Amit Lathigara[2, b)]

[1]*Ph.d Scholar, RK University, Rajkot, India*
[2]*Team Lead-Learning and Development, Tatvasoft, Ahmedabad, India*
3Dean, RK University, Rajkot, India

a) Corresponding author: sweetyp.ait@gmail.com
b)amit.lathigara@rku.ac.in

Abstract: In this contemporary lifestyle human beings choose a healthy lifestyle and one of them is yoga. Yoga focuses on physical and mental and spiritual well being. It requires discipline in order to perform, in fast paced life people prefer self-taught, however beginners are not able to recognize their incorrect posture during the initial stage and consequences lead to one's physique and brain injuries. For that reason, it is mandatory to have proper body alignment from the beginning. Using Deep learning algorithms, a technique to detect inappropriate yoga postures of a user. The first real-time multi-person system,Open pose, transformed the area of estimating a human body stance.With minimal latency, the proposed algorithm estimates and categorizes yoga positions into five main groups. Its intention is to act as a source for both academics and industry researchers looking into different Yoga position identification technologies. We evaluated the most recent and promising deep learning algorithms for Yoga stance monitoring and recognition.Using the media pipe and the OpenCV library, the input is preprocessed as an image, the object is recognised, and the core of the human body are identified. Training and testing logistic regression models.Media pipe and the OpenCV library are used to identify any photos that have been preprocessed as an input form.At the end of the paper, barriers and future developments are discussed. This survey can help researchers better understand current systems and propose new ways by solving the stated difficulties.

INTRODUCTION

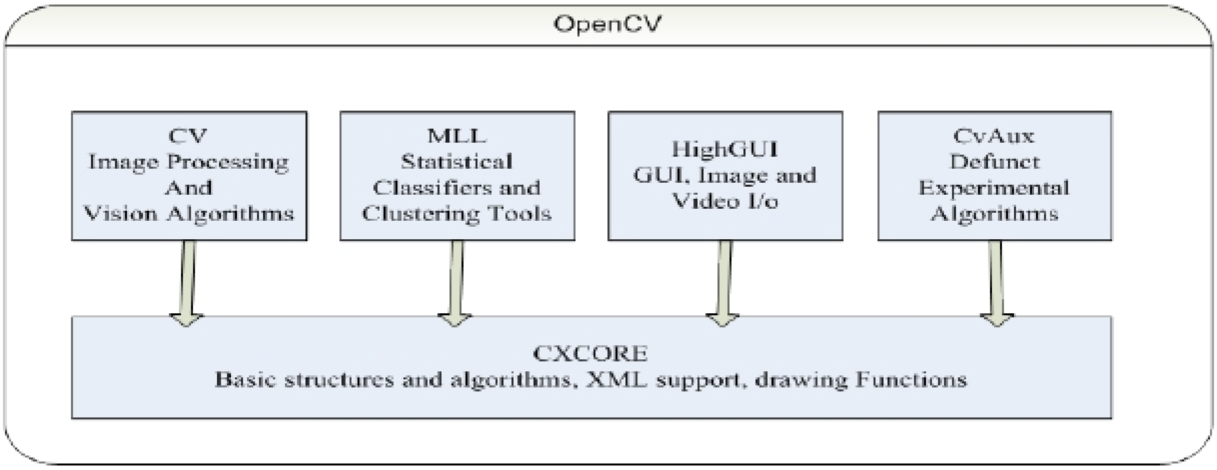
Human posture identification is a challenging and time-consuming operation in the field of computer vision.It is concerned with the localisation of human joints in a photograph or video in order to create a skeleton representation. Finding a user's activities in a photograph may be difficult since it depends on a number of factors such as image scale and determination, lighting fluctuation, backdrop confusion, venture variations, and human contact with the environment. The concern with yoga is that, like any other workout, it is critical to perform it correctly, since any wrong posture during a yoga session is generally ineffective and sometimes harmful. As a result, an instructor must be present to monitor the session and maintain good posture.

Human posture estimation has improved immensely from machine learning in recent years, with huge improvements in accuracy. This study aims to gain understanding by researching various approaches to yoga position categorization. This project includes referrals from unpublished articles, technical reports, and newspapers. The second section discusses cause estimation and extensively covers many types of cause estimate methods, going one level deeper to explain statistical formulas. Different methods for cause extraction are then discussed in combination with machine learning-based models Like logistical regression and python libraries like MediaPipe, OpenCV for cause estimate.

PROPOSED METHODOLOGY

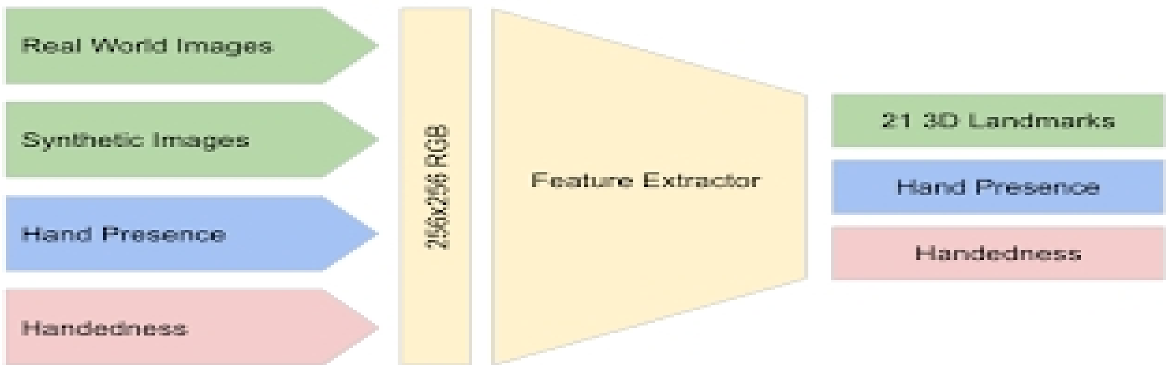
OpenCV

OpenCV is written in the Python open-source library used for computer vision in applications such as artificial intelligence, machine learning, and face recognition. It performs upon each picture frame. It includes library files that are used to compute angles. It operates a camera that detects the difference between a genuine image and an unknown one.



MediaPipe

MediaPipe Posture is a machine learning technique for high-fidelity body pose tracking that uses RGB video frames to infer 33 3D landmarks and a background segmentation mask over the entire body. The pipeline initially locates the human region-of-interest (ROI) inside the frame using a detector. Following that, the tracker masks within the ROI uses the cropped picture as input MediaPipe. Pose may be used to generate a full-body segmentation mask.



Requirements

```
pip install mediapipe
pip install keras
pip install tensorflow
pip install opencv-python
pip install numpy
```

```python
for i in os.listdir():
        if i.split(".")[-1] == "npy" and not(i.split(".")[0] == "labels"):
                if not(is_init):
                        is_init = True
                        X = np.load(i)
                        size = X.shape[0]
                        y = np.array([i.split('.')[0]]*size).reshape(-1,1)
                else:
                        X = np.concatenate((X, np.load(i)))
                        y = np.concatenate((y, np.array([i.split('.')[0]]*size).reshape(-1,1)))

                label.append(i.split('.')[0])
                dictionary[i.split('.')[0]] = c
                c = c+1


for i in range(y.shape[0]):
        y[i, 0] = dictionary[y[i, 0]]
y = np.array(y, dtype="int32")


y = to_categorical(y)

X_new = X.copy()
y_new = y.copy()
counter = 0

cnt = np.arange(X.shape[0])
np.random.shuffle(cnt)
```

```
for i in cnt:
        X_new[counter] = X[i]
        y_new[counter] = y[i]
        counter = counter + 1


ip = Input(shape=(X.shape[1]))

m = Dense(128, activation="tanh")(ip)
m = Dense(64, activation="tanh")(m)

op = Dense(y.shape[1], activation="softmax")(m)

model = Model(inputs=ip, outputs=op)

model.compile(optimizer='rmsprop', loss="categorical_crossentropy", metrics=['acc'])

model.fit(X_new, y_new, epochs=80)


model.save("model.h5")
np.save("labels.npy", np.array(label))
```

IMPLEMENTATION

Image Capture

In the beginning stage, use an RGB camera to capture the image.
The RGB camera is used to collect colour and depth pictures. The camera is installed and fixed on a tripod with a suitable frame that centres the person executing the yoga positions. The camera and the user are kept at a distance of 4 to 5 metres.

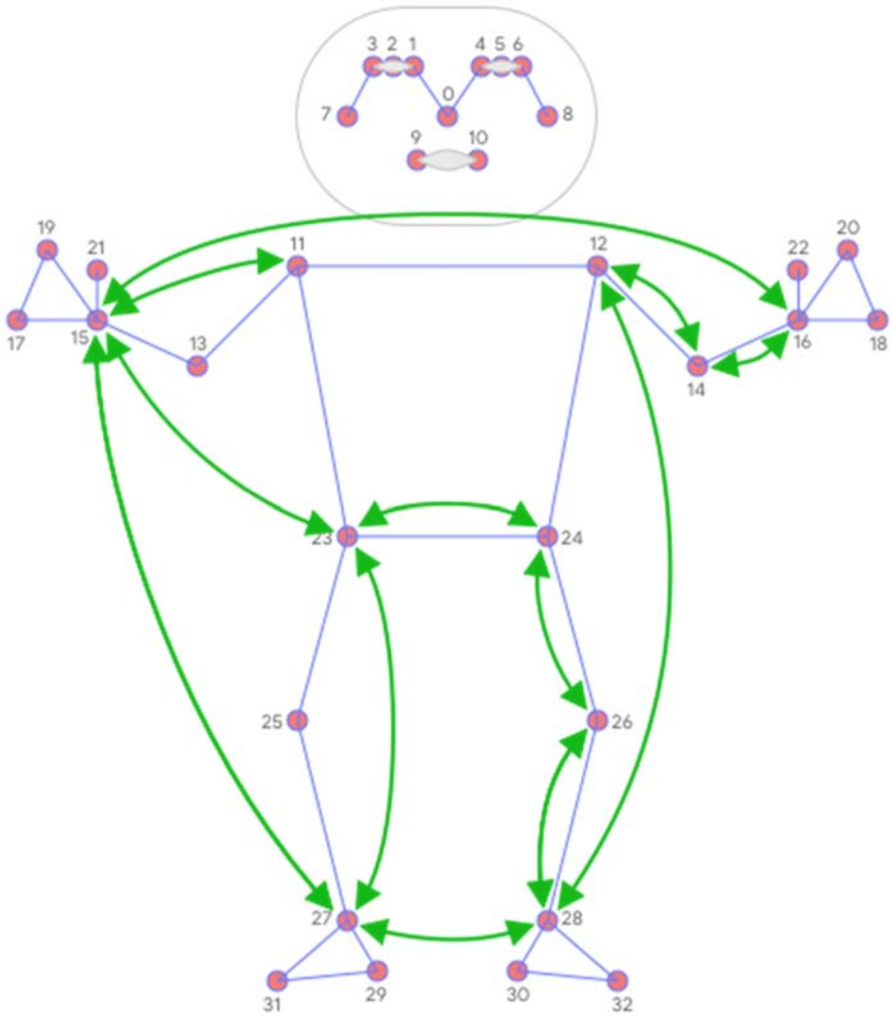
View Image

In the second step, use the function to take a sample image. Opencv will be used to read the picture.

Carry out Landmark Detection

In the third step, a human skeleton of the human practicing the yoga positions is created using mediapipe, and the findings are presented as 33 basic essential points:

Following the pose identification, a total collection of thirty-three points identifying the main person's body joint positions in the image is generated. Each landmark contains:
x: The picture width normalizes the landmark x-coordinate to [0.0, 1.0].
The picture height normalizes the landmark y-coordinate to [0.0, 1.0].
z: The z-coordinate of a landmark adjusted with the same level as x. It represents the depth of the landmark, with the origin being the halfway of the hips, therefore the lesser the number of z, the near the position to the camera.



Pose Classification with Angle Heuristics

Classify various yoga positions using estimated angles of various joints in the fourth step.
The first point is the starting point of the first line, the second point is considered as the ending point of the first line as well as the starting point of the second line, and the third point is the ending point of the second line.

Pose Classification

Finally, the stance can be categorized based on a different combination of body part angles. Initialize the pose's label. At this point, it is not recognised as 'Unknown Pose.' Measure the necessary angles, and if the stance is properly identified, change the color with which the caption will be printed on the image.

Modeling and analysis

Our method seeks to detect the user's Yoga asanas from real-time footage. The procedure is divided into four major phases. First, data collection is undertaken, which is a real-time procedure that occurs concurrently with detection. Second, Mediapipe is utilised to determine the joint positions to detect the visible joints and estimate the location of the non-visible joints using Leonardo's Virtuvian man idea. The discovered keypoints are fed into our model, where logistic regression detects patterns and examines their evolution over time. Next, the model and training technique of framewise prediction and polling strategy for distinct images with probability range is predicted as output with the name Asanas are explained.

PREPROCESSING

The Python library is used for data preparation and includes features such as comprehensive file format support, fast internal representation, thumbnail creation, picture file format conversion, and image filtering.

FEATURE EXTRACTION

Mediapipe and the OpenCV library are used to extract features. We must follow the procedures during feature extraction. 1. We collect picture samples from the target workouts and use pose prediction to determine their poses. 2. We must convert the obtained posture positions to a structure appropriate for the learner and create a training set from these vectorized keypoints. 3.We then did categorization.

MODELING

The structure of each yoga posture is determined by the x, y, and z values of the joint locations. To categorise data and detect the yoga stance, we employed a logistic regression model. The x, y, and z values are provided to the model as X as an input and Y as an output respectively. 70% of the data was utilised for training and 30% for testing. Our model is nearly perfect.

CONCLUSION

A yoga posture classifier that works properly on photos, static video, and live video of any user was successfully built in this study. The research begins with the construction of an environment and then moves on to data collecting from open data sources. The Mediapipe posture estimating library is used for human position estimate, which produces body key points, which serve as the foundation for a new dataset. The target variables are then adjusted during data preparation. Following this, data is normalised for improved performance of machine learning algorithms, and feature engineering of features begins, with different joint angles of the body determined using the method.Because the data has been thoroughly preprocessed, it is eventually supplied to machine learning models. These models are evaluated on test data and compared based on their accuracy score. Among all classifiers, the logistic regression classifier has the highest score of 94%. For categorization, a threshold value of 97% is employed.

FUTURE WORKS

The models' performance is determined on the accuracy of OpenPose posture estimation, which may fail in circumstances of overlap between participants or overlap among body parts. This system may be equipped with a portable device for self-training and real-time forecasts. This paper exhibits the use of activity recognition in real-world situations. A similar method may be used for posture identification in applications such as sports, monitoring, and healthcare. Multi-person pose estimation is a completely new subject with a lot of room for investigation. There are several instances in which a single person posture estimate would not enough; for example, pose estimation in overcrowded scenarios would need tracking and detecting the pose of each participant.Many of the characteristics highlighted previously in this survey, such as backdrop, lighting, overlapping figures, and so on, would make multi-person posture estimate even more difficult.

REFERENCES

1. Rahul Ratusaria, 1Tushar Baghel, 1Ayush Chander Vanshi and 1 Neeraj Garg "GYM REP TRACKER USING MEDIAPIPE AND PYTHON" 2021
2. Ms. P Charith, Mr. K R Rohit Srivatsa, Mr. Prajwal Kumar B R, Ms. Niharika D "HUMAN POSE ESTIMATION IN FITNESS TRACKING AND GUIDANCE"
3. Valentin Bazarevsky, Ivan Grishchenko. Karthik Raveendran, Tyler Zhu, Fan Zhang, Matthias Grundmann "BlazePose: On-device Real-time Body Pose tracking" 2020
4. Alex Moran, Bart Gebka, Joshua Goldshteyn,Autumn Beyer, Nathan Johnson, and Alexander Neuwirth "Muscle Vision: Real Time Keypoint Based Pose Classification of Physical Exercises"
5. Indriani1 Moh.Harris1 Ali Suryaperdana Agoes1, "Applying Hand Gesture Recognition for User Guide Application Using MediaPipe"
6. Debabrata Swain, Santosh Satapathy, Pramoda Patro, Aditya Kumar Sahu "Yoga Pose Monitoring System using Deep Learning" 2022
7. Yubin Wu 1, Qianqian Lin 1, Mingrun Yang 1, Jing Liu 1, Jing Tian 1, Dev Kapil 2 and Laura Vanderbloemen "A Computer Vision-Based Yoga Pose Grading Approach Using Contrastive Skeleton Feature Representations" 2022
8. Swapnil Dawange, Akash Chavan, Abhijit Dusane, H.P.Bhabad "Workout Analysis Using Mediapipe BlazePose and Machine Learning" 2021
9. 1Kashish Jain, 2Jignasha Jadav, 3Manashvini Yadav and 4Dr Yogita Mane "AI FITNESS TRAINER" 2014
10. Renhao Huang, Jiqing Wang, Haowei Lou, Haodong Lu, Bofei Wang "Miss Yoga: A Yoga Assistant Mobile Application Based on Keypoint Detection"
11. Pooja Rakate "A Deep Learning Framework to Classify Yoga Poses Hierarchically"
12. Abhishek Sharma, Yash Shah, Yash Agrawal, Prateek Jain "REAL-TIME RECOGNITION OF YOGA POSES USING COMPUTER VISION FOR SMART HEALTH CARE" 2022
13. Alexander Toshev, Christian Szegedy "DeepPose: Human Pose Estimation via Deep Neural Networks"
14. Anna Lai, Bhargav Reddy, Bruis Van Vlijmen "Yogi.ai: Deep Learning for Yoga"
15. Dr Anil Kumar Dubey, 2Sameeksha Agarwal, 3Tarandeep Singh, 4Vaibhav Sharma "YOGA POSE CORRECTION USING POSENET" 2022 https://www.researchgate.net/publication/362280366
16. Dr. Maya Bembde1, Swapnali Barude2, Pradnya Shinde2, Tejaswini Thorat2, Deepak Thakar2 "Yoga Posture Detection and Correction System" 2022
17. Krushnkant Somwanshi1, Nayan Jagtap2, Abhishek Badadal3 "Classification of Yoga Posture Using POSENET" 2022
18. Deepak Kumar, Anurag Sinha "Yoga Pose Detection and Classification Using Deep Learning" November 2020 https://www.researchgate.net/publication/346659912
19. CE ZHENG,WENHAN WU,CHEN CHEN,MUBARAK SHAH "Deep Learning-Based Human Pose Estimation: A Survey" Jan 2022
20. Mahendran N "Deep Learning for Fitness" Sep 2021