Human Posture Detection

Sethu Nandan.O.G (193079011) Nihar Shah (193079018)
R.V.Satwik (193079009) Deval Patel (19307R011)

8^{th} December, 2020.

`https://github.com/niharhshah/CS725-Course-Project`

1 Introduction

In this report we present a human posture detection task using deep learning techniques. We show the block diagram of our task in below figure.

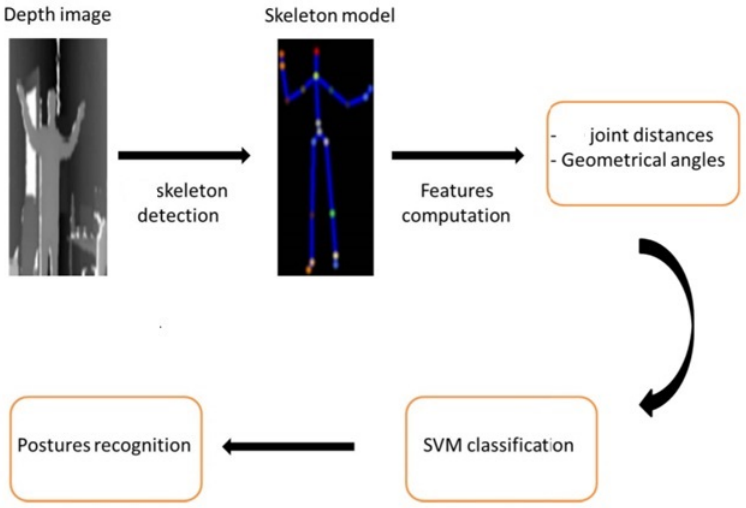


Figure 1: Flow diagram of the implementation
`https://www.seminarsonly.com/computer%20science/human-posture-recognition-system.php`

Pose Estimation is a general problem in Computer Vision where we detect the position and orientation of an object. This usually means detecting keypoint locations that describe the object.

2 Existing models

The main approaches for generating human pose recognition can be broadly classified into a top-down and a bottom-up approach. In the top-down approach, a boundary box of the person is detected. The image of the person mapped within the boundary box is processed further to generate key points (location of joints) and pose lines (lines connecting the joint locations) are detected to generate a stick-man mapping of the human in the picture. Mask-RCNN and AlphaPose are examples of the top-down approach implementation. On the other hand, the bottom-up approach detects key points first and then estimates the person's pose by joining the key points. Various procedures can be followed for estimating the pose from key points and among them, the bipartite graph using a greedy algorithm is well known in the art. The OpenPose model is an exemplary implementation of the bottom-up approach.

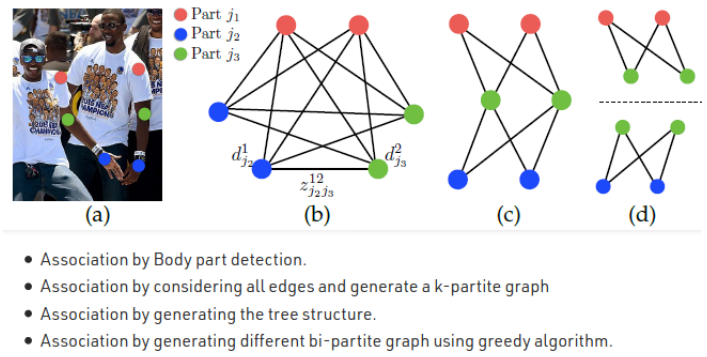


Figure 2: Various methods of matching key points in an image
`geeksforgeeks.org/openpose-human-pose-estimation-method/`

3 Overview and Hypothesis

The dataset used in the current implementation is a mixture of the COCO dataset and Stanford action 40 datasets. The final dataset for training and testing is generated using the annotations and the labels provided in either dataset. A principal component analysis was conducted on the final dataset to see if the data can be separated using a boundary, possibly of higher dimensions.

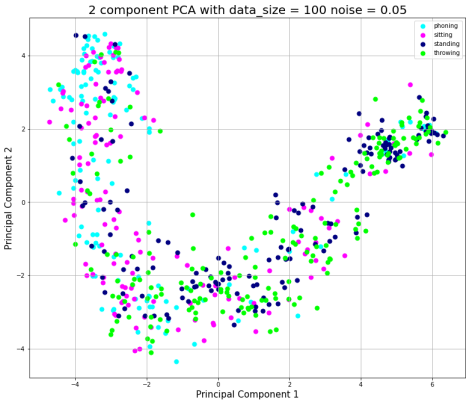


Figure 3: Principal component analysis of a sample dataset

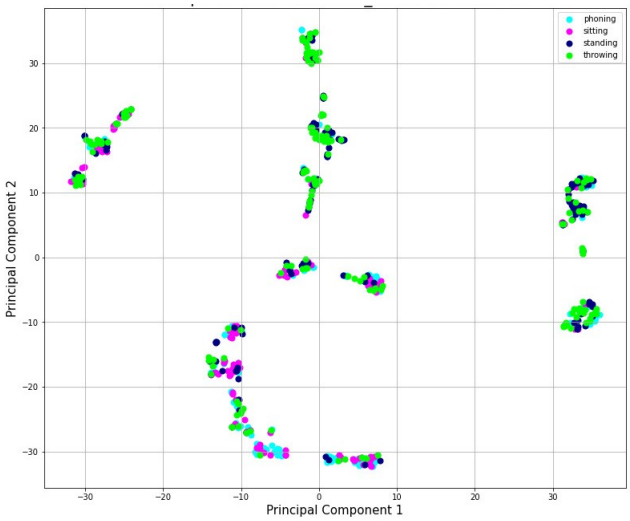


Figure 4: t-SNE analysis of sample dataset

From the image, the data appears to be separable if we use a higher-order curve for separation boundary. And it can also be seen from the image that the certain poses are so close that it can be very hard to separate them, as is the case with a cluster near (6,2) in the image for standing and throwing. A further analysis using t-Distributed Stochastic Neighbor Embedding (t-SNE) was conducted on the final dataset.

The form of any human pose is usually such that the location of joints and the orientation of joints are to be relatively in fixed states from the center of the person. The pose of a human while running would have knees and ankles located farther from the head than the case when the human is in a squatting pose. This brings to the hypothesis we tried to test, to see if this pattern in data can be used for estimating a human pose. A

major unforeseen hurdle was the presence of tilted images, images with multiple people, and images with partial body hidden or covered either by the presence of objects or due to the orientation of the image such as a side view image of a person where the second half of the person is hidden from the image.

The approach used in the project is the use of the OpenPose model for determining the key points in an image and then use the determined key points to train a support vector machine (SVM) model. The images are sorted into the necessary labeled folders which serve as the targets for the images. The OpenPose model API (in the form of a docker model) is used for determining the key points and pose lines in an image. A standard sklearn SVM model was used during the training and predictions.

As we did see above, the dataset needs very high order boundary functions to effectively predict the pose of the image. Hence a high order polynomial kernel and a radial basis function kernel were chosen for the training of the SVM. The polynomial kernel functions serve as an example of a finite-dimensional kernel function whereas the radial basis function serves for an exemplary implementation of the infinite-dimensional kernel. The optimal order for the polynomial kernel function was determined using a grid search algorithm on the training and test accuracy as metrics for measurement.

```
param = [
    {
        "kernel": ["poly"],
        "degree":[0,1,2,3,4, 5, 6]#, 7]#, 8,9,10]
    }
]

# request probability estimation
svm_model = SVC(max_iter=10000, gamma='scale', probability=True)
clf = GridSearchCV(svm_model, param, cv=5, n_jobs=2, verbose=3)
clf.fit(x_train, y_train)
```

Figure 5: Conducting grid search on polynomial parameter

Since the location of the person in an image is very arbitrary, one approach known in the art to easily overcome the need for image processing to identify and narrow down the region is the use of polar coordinates relative to the center of the person. The location of joints expressed in relative position to the center of the person serves as automatically centering upon the person. The joint locations are therefore considered in polar coordinates from the center of the person (not the center of the image). The center of the person approximately is the average of the coordinates of the joint locations. The use of polar/radial coordinates makes it easier in case of need of rotation of the stick man mapping.

The polar coordinate representation of the stick-man mappings of the various images categorized as 'standing' is shown in the figure

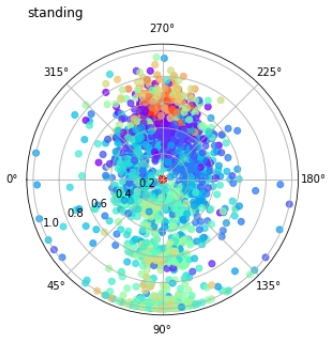


Figure 6: Polar co-ordinates of images with category as 'standing'

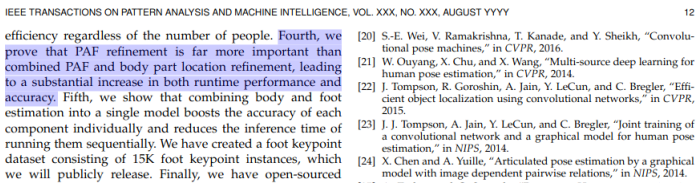
3.1 Dataset preparation

The first hurdle we faced was to find a clean dataset. COCO dataset had some figures of objects mixed up with humans. The only obvious pose for an object is 'standing'. The model we made was not flexible with such kind of predictions. So we took the stanford 40 actions dataset which was relatively less co-related in itself than COCO but it still needed some modifications. There were some poses like *riding a boat* and *using a computer* which can be merged with sitting.

COCO dataset: `https://cocodataset.org/#download`
Stanford-40-Action Dataset: `http://vision.stanford.edu/Datasets/40actions.html`

4 Learnings

One of the key scopes where the prediction of the model can be improved was by using the orientation of pose lines along with the joint locations. This part wasn't successfully implemented at the time of writing this report. This is because pose lines need extra information to be linked along with the orientation, without which they essentially behave as positional vectors originating at the origin. Without such an information storage method, the same set of pose lines can be arranged in multiple ways to achieve different poses. But the reason why using the pose lines along joint locations hint at improving the results can be seen from the conclusions from one of the papers related to the OpenPose, in which the authors themselves agree that accuracy of the part affinity fields (that essentially identify pose lines in an image) is more significant compared to the accuracy of the confidence maps.



IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. XXX, NO. XXX, AUGUST YYYY 12

efficiency regardless of the number of people. Fourth, we prove that PAF refinement is far more important than combined PAF and body part location refinement, leading to a substantial increase in both runtime performance and accuracy. Fifth, we show that combining body and foot estimation into a single model boosts the accuracy of each component individually and reduces the inference time of running them sequentially. We have created a foot keypoint dataset consisting of 15K foot keypoint instances, which we will publicly release. Finally, we have open-sourced

[20] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," in *CVPR*, 2016.
[21] W. Ouyang, X. Chu, and X. Wang, "Multi-source deep learning for human pose estimation," in *CVPR*, 2014.
[22] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, and C. Bregler, "Efficient object localization using convolutional networks," in *CVPR*, 2015.
[23] J. J. Tompson, A. Jain, Y. LeCun, and C. Bregler, "Joint training of a convolutional network and a graphical model for human pose estimation," in *NIPS*, 2014.
[24] X. Chen and A. Yuille, "Articulated pose estimation by a graphical model with image dependent pairwise relations," in *NIPS*, 2014.

Figure 7: One of the conclusions from the paper OpenPose: Realtime Multi-Person 2D Pose-Estimation using Part Affinity Fields

Another reason for the low accuracy in the model is the inherent shortcomings in the key point identification by the OpenPose model. The OpenPose model gives incorrect key points when the ground truth pose (estimated from the image) is itself not well distinguishable. Secondly, the greedy algorithm based linking of the key points by the OpenPose model sometimes result in 'extended' stick man mappings for images with multiple persons (an example for this is provided in the false predictions section).

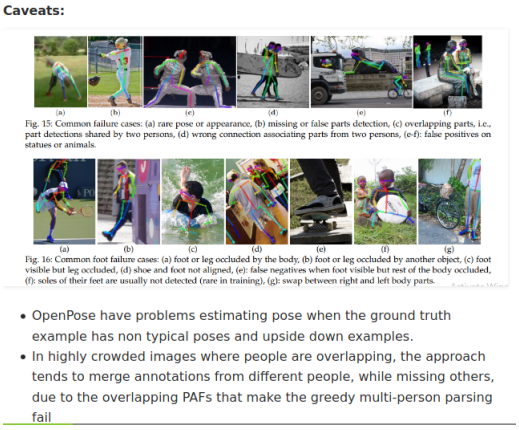
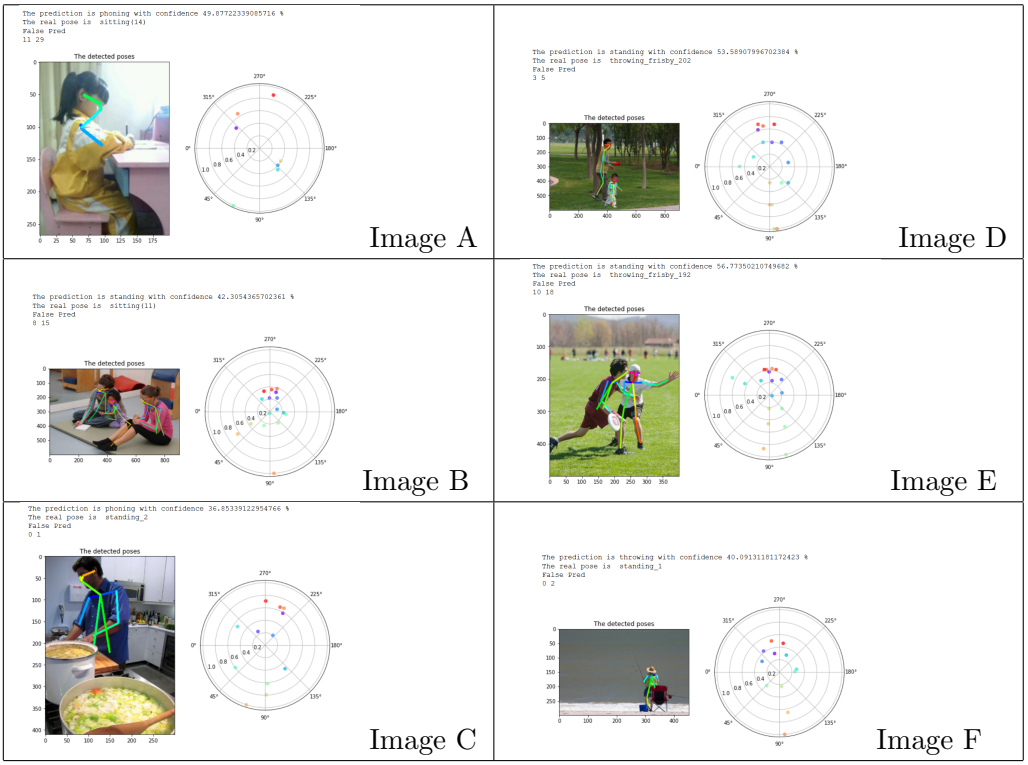


Figure 8: Unusual pose detection for images with multiple people and images with non standard pose (source: www.geeksforgeeks.org/openpose-human-pose-estimation-method/)

5 False predictions



In "Image A" we can see that only half of the body is visible. The partial body leads to the partial creation of the stickman and then leads to a false prediction of Phoneing when the correct posture is sitting. In "Image B" the greedy nature of OpenPose has made a stickman line stretching from one child to another girl. This stretching line cumulatively adds errors with multiple people. In "Image C, D" all persons are visible and a total stickman is made, but in both the image the person in the context in the image is not used to estimate the posture in case of image C the baby is standing which ultimately says that the person in the context is standing. But in the image person of context is a man throwing a frisbee. which is surely visible on a polar plot next to the image. In "Image C, F" a single person is not correctly detected as the part of the body or a limb is missing in the image, the rare occurrence is also a contributing factor.

6 Final Results and further scope

To get final results we changed some hyperparameters such as type of kernel and the degree of the polynomial this can affect testing accuracy and achieving greater accuracy. We have achieved a maximum Test accuracy of 53.19% for a polynomial kernel of degree 6. This test accuracy reduces significantly if we change our coordinate system to Cartesian. with Cartesian,

we have found accuracy to be of 40.42% with a polynomial kernel of degree 7.

7 Conclusion

This model can be used for a single person with the full-body image and regular pose which will have maximum accuracy. This needs an extremely clean dataset which will have all images to be with a single person and full-body visibility.

8 Acknowledgements

We express our special thanks to Prof. Preethi Jyoti our course instructor and all the TAs for helping us throughout the course, Prof. Amit Sethi of EE department for granting us access to MeDAL's System to run and test our model.

References

[1] https://github.com/IBM/yogait

[2] https://arxiv.org/pdf/1812.08008.pdf

[3] https://github.com/ildoonet/tf-pose-estimation