2D to 3D Human pose reconstruction in the wild

Dr.Snehasis Mukherjee
IIIT Sricity, India
snehasis.mukherjee@iiits.in

Siddharth Kumar
IIIT Sricity, India
siddharth.k16@iiits.in

Shobhit Malarya
IIIT Sricity, India
shobhit.m16@iiits.in

ABSTRACT

Although significant improvement has been achieved recently in 3D human pose estimation,we propose a new pipeline and architecture for 3D pose estimation of a human for in-the-wild images. The main objective is to minimize the re-projection loss of key points, which allows our model to be trained using in-the-wild images that only have ground truth 2D annotations.
We use different networks at different stages of pipeline in our model.We have used publically available dataset COCO[1] for training and validating our results. The Pipeline of our model consists of detecting and segmenting a person from an image using yolo v3 [2] ,scaling the image without disturbing the aspect ratio of the image ,inferring 17 key points of a human body using posenet.
We have also discussed issues regarding a subset of images in which the 2d key point detection is not performing well and have also proposed another addition to the model,a regression network,to overcome this issue.

CCS CONCEPTS

• **Computing methodologies** → *Machine Learning*; **Neural networks**.

KEYWORDS

CPM (Convolutional pose machine)[3],HMR
(Human mesh Reconstruction)[4], SSD (Single shot detection)[5], YOLO (You only look once)[2]

1 INTRODUCTION

A Human Pose Skeleton represents the orientation of a person in a graphical format. Essentially, it is a set of coordinates that can be connected to describe the pose of the person.Each co-ordinate in the skeleton is known as a part (or a joint, or a key point). A valid connection between two parts is known as a pair (or a limb). Note that, not all part combinations give rise to valid pairs.Knowing the orientation of a person opens avenues for several real-life applications, some of which are Activity Recognition, Motion Capture and Augmented Reality, Training Robots,Motion Tracking for Consoles e.t.c.For example, a very popular Deep Learning app HomeCourt uses Pose Estimation to analyse Basketball player movements Several approaches to Human Pose. Estimation were introduced over the years. The earliest (and slowest) methods typically estimating the pose of a single person in an image which only had one person to begin with. These methods often identify the individual parts first, followed by forming connections between them to create the pose.

Challenges faced with Pose Estimation?
Strong articulations, small and barely visible joints which leads to difficulty in inferring the key points of human body like nose,eyes e.t.c, clothing, and lighting changes make this a difficult problem, specially when dealing with the images in-the-wild.
Occlusions are also one of the major problems in pose estimations and also a very challenging task to deal with.

1.1 Contributions

The key contributions of this work is given as follows:

- A new pipeline: We present a new pipeline in fig 1 which includes image segmentation,inferring 2D joint key points and projecting them to 3D.
- We also add a regression network,our novelty, which fine tunes the ankle joint points and and adjust them to their correct place forming 2D belief map.

The remaining paper is organized as follows: Section 2 presents the state-of-art methods for 3D pose estimation. We present a brief overview about the proposed pipeline and network in section 3, followed by a brief of the dataset used. In section 4, the detailed methodology of our approach. Section 5 explains the results of all the models and presents all ablation studies followed by discussions and future work in section 6.

2 RELATED WORK

In this section, we briefly provide a review of the various works done in context of 2D to 3D pose estimation.

2.1 Different approaches to 2D Human Pose Estimation

- **Classical approaches**
  The classical approach to articulated pose estimation is using the pictorial structures framework. The basic idea here is to represent an object by a collection of parts arranged in a deformable configuration (not rigid). A part is an appearance template which is matched in an image. Springs show the spatial connections between parts. When parts are parameterized by pixel location and orientation, the resulting structure can model articulation which is very relevant in pose estimation.[6]
  This method, however, comes with the limitation of having a pose model not depending on image data. As a result, research has focused on enriching the representational power of the models.

- **Deep Learning based approaches**
  The classical pipeline has its limitations and Pose estimation has been greatly reshaped by CNNs. With the introduction of "Deep Pose" by Toshev et al [7], research on human pose estimation began to shift from classic approaches to Deep

Learning. Most of the recent pose estimation systems have universally adopted Conv-Nets as their main building block, largely replacing hand-crafted features and graphical models; this strategy has yielded drastic improvements on standard benchmarks.

So we propose a new pipeline which helps us in detecting the key points more efficiently and accurately. Different stages in our pipeline combines various networks and uses them to infer cues that helps in prediction of the 17 key points which then can be used for 3D pose estimation.

2.2 End-to-end Recovery of Human Shape and Pose

HMR [4] presents an end-to-end framework for recovering a full 3D mesh of a human body from a single RGB image. They use the generative human body model, SMPL [8], which para-meterizes the mesh by 3D joint angles and a low dimensional linear shape space. Their approach is similar to 3D interpreter networks in the use of re-projection loss and the more recent adversarial inverse graphics networks for the use of the adversarial prior. They go beyond the existing techniques in multiple ways:

- They infer 3D mesh parameters directly from image features, while previous approaches infer them from 2D key points. This avoids the need for two stage training and also avoids throwing away valuable information in the image such as context.
- Going beyond skeletons, They output meshes, which are more complex and more appropriate for many applications. Again, no additional inference step is needed.
- framework is trained in an end-to-end manner.They outperform previous approaches that output 3D -meshes in terms of 3D joint error and run time.
- They show results with and without paired 2D-to-3D data. Even without using any paired 2D-to-3D supervision, Their approach produces reasonable 3D reconstructions. This is most exciting because it opens up possibilities for learning 3D from large amounts of 2D data.

Since there are no datasets for evaluating 3D mesh reconstructions of humans from in-the-wild images, they are bound to evaluate their approach on the standard 3D joint location estimation task. Their approach out performs previous methods that estimate SMPL [8] parameters from 2D joints and is competitive with approaches that only output 3D skeletons.

2.3 CPM : Convolutional Pose Machines

[3] Convolutional Neural Network is naturally a sequence of stages if multiple losses and predictors are inserted at the intermediate layers.

In addition, the stacked convolutional layers' perceptual field increases as deepening, which means that more contextual infomation is taken into consideration helping refine the output.

CPM [3] proposed a sequential architecture that composed of convolutional networks which directly operate on belief maps from previous stages, producing increasingly refined estimates for part locations.

When training, every stage has its own loss function to predict parts. These losses work similar with the auxiliary classifiers in GoogleNet[9], which helps alleviate the problem caused by the vanishing of gradient. The network can be trained end-to-end. Compared with traditional pose machine, CPM[3] is much easier to train.

Therefore it addresses the vanishing gradients problem during training by providing a natural learning objective function that enforces intermediate supervision.

3 PROPOSED NETWORK FOR GETTING 2D BELIEF MAPS

3.1 Dataset

COCO is a large-scale object detection, segmentation, and captioning dataset. COCO has several features: Object segmentation, Recognition in context, Superpixel stuff segmentation, 330K images (>200K labeled), 1.5 million object instances, 80 object categories, 91 stuff categories, 5 captions per image, 250,000 people with keypoints. It is one of the best image datasets available, so it is widely used in cutting edge image recognition artificial intelligence research. It is used in open source projects such as Facebook Research's Detectron, Matterport's Mask R-CNN, endernewton's Tensorflow Faster RCNN for Object Detection, and others.

COCO has five annotation types: for object detection, keypoint detection, stuff segmentation, panoptic segmentation, and image captioning.Out of which we use keypoint detection.

Keypoints add additional information about a segmented object. They specify a list of points of interest, connections between those points, where those points are within the segmentation, and whether the points are visible.

As of the 2017 version of the dataset, there is only one category ("person") in the COCO dataset with keypoints, but this could theoretically be expanded to any category that might have different points of interest.

Number of keypoints is specified in sets of 3, (x, y, v) where x and y indicate pixel positions in the image. v indicates visibility— v=0: not labeled (in which case x=y=0), v=1: labeled but not visible, and v=2: labeled and visible.

In the case of a person, keypoints indicate different body parts. The skeleton indicates connections between points. Among the 91 categories we use the images related to sports category with subcategory related to tennis.

4 METHODOLOGY

In this section we describe the models used and created on each step of the pipeline along with the regression network we use to fine tune our joint point in 2D belief map..

4.1 Image Segementation

Since we are dealing with images in-the-wild, so the first step which we include in our pipeline is segmenting the human from the given image.

Since a lot of images have noise in the background so segmenting

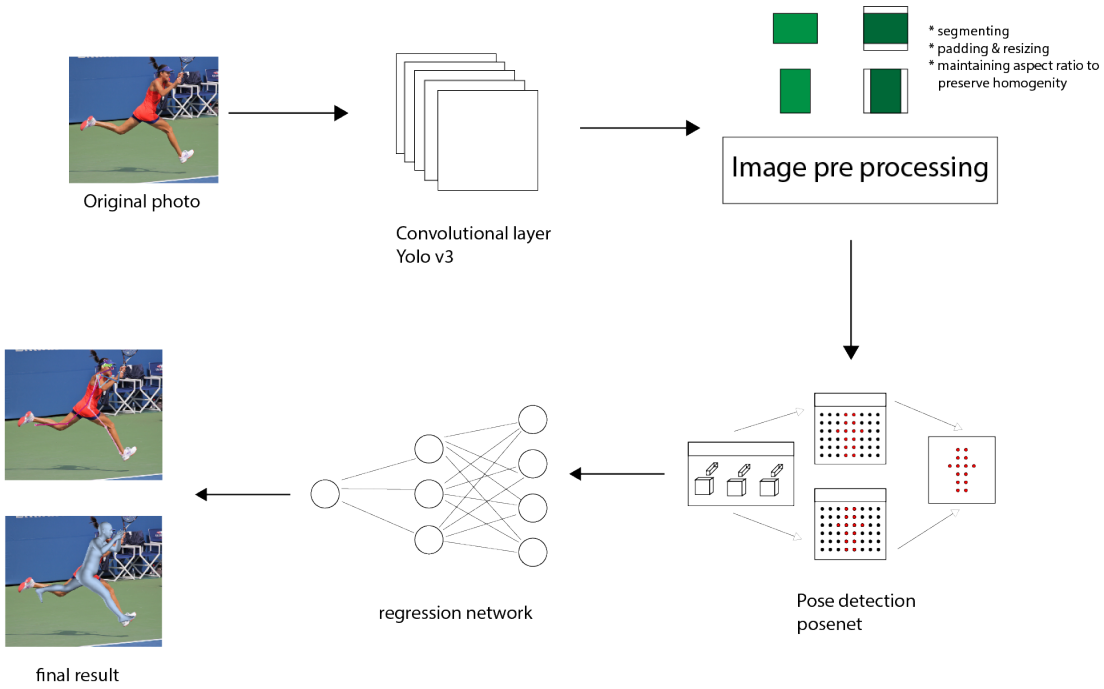

Figure 1: Block diagram of proposed multi-layer attention network on multi-modality input features

plays a very important role in estimation of 2D joint keypoints. For this purpose we use a pretrained model of yolov3 which is trained for 80 classes on Imagenet dataset.
YOLO performs a linear regression using two fully connected layers to make 7 x 7 x 2 boundary box predictions. To make a final prediction, we keep those with high box confidence scores (greater than 0.25) as the final predictions
We get the output bounding box co-ordinates,crop the image and pad the image to get a nxn sized image taking into consideration the aspect ratio.

4.2 Keypoint Detection: 2D belief mapping

In order to infer the the 2D keypoints from the segemented image we have used posenet.
At a high level the pose estimation in PoseNet happens in two phases:

(1) An input RGB image is fed through a convolutional neural network.
(2) Either a single-pose or multi-pose decoding algorithm is used to decode poses, pose confidence scores, keypoint positions, and keypoint confidence scores from the model outputs.

Following is the Explanation of above key points:

- the output of posenet is (x,y,c) where x,y are co-ordinates of keypoints and c is their confidence.
- Pose confidence score — this determines the overall confidence in the estimation of a pose. It ranges between 0.0 and 1.0. It can be used to hide poses that are not deemed strong enough.
- Keypoint — a part of a person's pose that is estimated, such as the nose, right ear, left knee, right foot, etc. It contains both a position and a keypoint confidence score. PoseNet currently detects 17 keypoints.
- Keypoint Confidence Score — this determines the confidence that an estimated keypoint position is accurate. It ranges between 0.0 and 1.0. It can be used to hide keypoints that are not deemed strong enough.

Problems faced with posenet and keypoint detection
The challenge we faced was a subset of images in which there is a considerable distance between the ankle and knee key points.
On these subset of images the PoseNet fails to detect the key points properly or confuses between the key points. So in order to get rid of this problem we have come up with a model pipeline which is shown in figure 1. The addition to the pipeline is a regression network,discussed in next section, which takes the output joints ,17 key points, as input and predicts the correct positions of the key points for pose estimation using co-relation between joint points.

4.3 Regression Network

In order to deal with the problem with posenet key point detection we have come up with this novel approach which uses regression between keypoints to predict the joint points.

- We use regression of ankle key points in order to improve the result of keypoint detection.
- For feature extraction we use correlation matrix, shown in fig 2, of keypoints to find the linearly dependent features.

- We have also taken in account the problem of multi-collinearity between the features.
- We have trained on 700 images and tested on 100 images from COCO dataset.

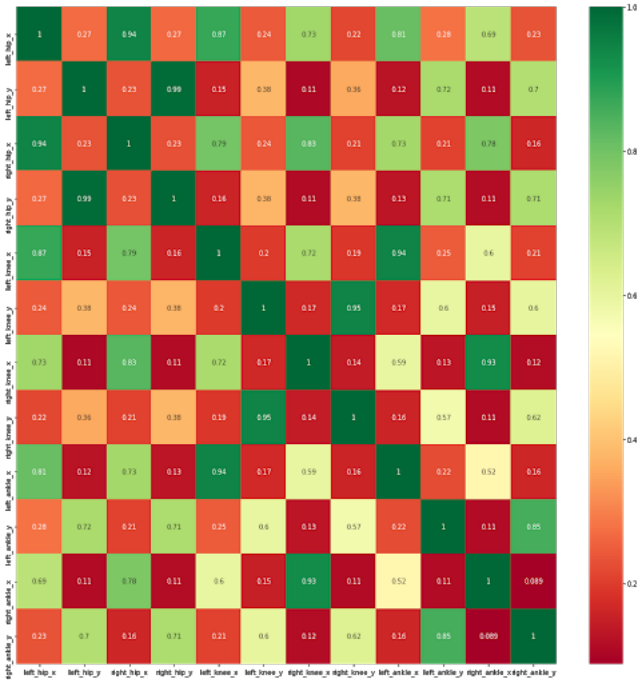

Figure 2: Co-relation matrix found between the ankle, knee and hip points)

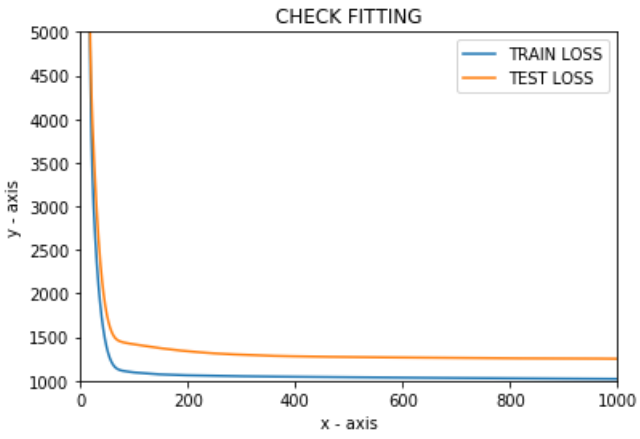

Figure 3: Train test graph

We have used the following formulation for joint correction:

(1) Right ankle x = A0 * Right knee x + C
(2) Right ankle y = A0 * Right knee y + A1 * right hip y + C
(3) Left ankle x = A0 * Left knee x + C
(4) Left ankle y = A0 * Left knee y + A1 * left hip y + C

Where C,A0 and A1 are weights associated with their corresponding features and Right ankle x refers to the x co-ordinate of pixel corresponding to the same and similarly for the rest other joint points as well.
For regression we have used a feed forward network having three dense layer with 500,200,1 hidden units respectively,with relu used as activation function.
We have used Mean Squared error as the loss function which takes loss between the predicted point and the annotated ground truth. The train test graph is fig 3.

5 RESULTS

Figure 4: 3D mesh rendering on an image without segmentation

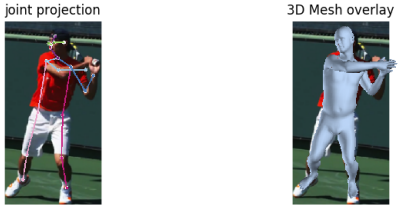

Figure 5: 3D mesh rendering on an image with segmentation

Figure 6: 2D results from hmr [4]

Figure 7: 2D results from posenet

Figure 8: ankle correction by our network

Figure 9: 2D results from hmr
[4]

Figure 10: ankle correction by our network

Figure 11: final result

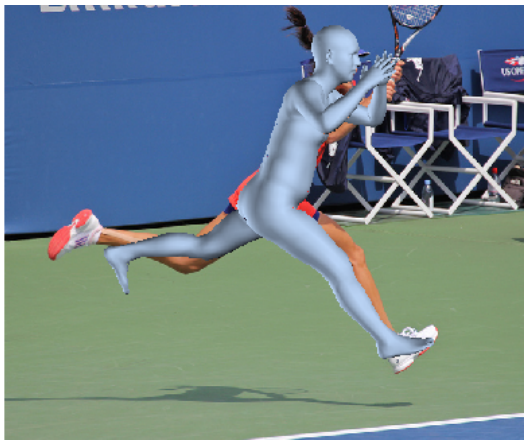

Figure 12: final result

6 DISCUSSIONS AND FUTURE WORK

This paper proposes a novel approach to improve the 2D belief map of the keypoints,which involves a regression network and equation formed based on co-relation matrix of joint points. For this task we observed that there is high co-relation between some joint points which we made use of in our network to predict the correct 2D joint points. In our approach we are able to predict the 2D keypoints better than the posenet, for the images in which there is a certain distance between the ankle points.

An illustrated example is the fig 6 which shows the 2D annotations by hmr, it is clear that the left knee and ankle are misplaced/wrongly predicted which will ultimately result in a bad 3D model, The fig 7 shows the result from posenet, the left ankle is correctly predicted by now the right ankle is misplaced. The fig 8 shows the correction of the right ankle by our network.

Another example is fig 9 - result of hmr and fig 10- result of posenet with our correction network.

As future work, we want to improve the accuracy of the joint point prediction by using other methods as an addition to our current approach.We can extend this work with videos, which is also a very challenging task.We would also like to explore more methods and alterntive approaches which can help us improving the 2D joint points prediction.

REFERENCES

[1] T.-Y. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *ECCV*, 2014.

[2] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779–788, 2015.

[3] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4724–4732, 2016.

[4] A. Kanazawa, M. Black, D. W. Jacobs, and J. Malik, "End-to-end recovery of human shape and pose," 12 2017.

[5] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," 2015, cite arxiv:1512.02325Comment: ECCV 2016. [Online]. Available: http://arxiv.org/abs/1512.02325

[6] Y. Yang and D. Ramanan, "Articulated human detection with flexible mixtures of parts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, pp. 2878–2890, 2013.

[7] A. Toshev and C. Szegedy, "Deeppose: Human pose estimation via deep neural networks." *CoRR*, vol. abs/1312.4659, 2013. [Online]. Available: http://dblp.uni-trier.de/db/journals/corr/corr1312.html#ToshevS13

[8] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "SMPL: A skinned multi-person linear model," *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, vol. 34, no. 6, pp. 248:1–248:16, Oct. 2015.

[9] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Computer Vision and Pattern Recognition (CVPR)*, 2015. [Online]. Available: http://arxiv.org/abs/1409.4842