Improved 3D Human Pose Estimation using RGB-Depth Data

Anonymous CVPR submission

Paper ID 6099

Abstract

Human Pose Estimation is estimating the joint positions, more formally the pose, of a person from an image or a video. The arrival of cheap RGB-depth devices, like Microsoft Kinect, has widened the research landscape in 3D Human Pose Estimation. Both RGB and Depth data have certain limitations, but can together complement each other for an improved model. With the goal of creating a multi-modal 3D Pose estimation pipeline, we have outperformed the best reported results on Human3.6M and SURREAL. Our results indicate the vast potential in working with both RGB and Depth data together, and suggests directions to further advance the state of the art in 3D Human Pose Estimation.

1. Introduction

Human Pose estimation is a key component in many computer vision applications, including video surveillance [62], smart environments [66], assisted living [14, 16], advanced driver assistance systems (ADAS) [22], and sport analysis [37]. A majority of initial research in the field was focused towards 2D Pose Estimation using RGB sensors. However, for a large number of these applications, giving the ability to understand complex spatial arrangements and depth ambiguities to machines is crucial, thus introducing the need of 3D Pose Estimation. The introduction and popularity of low-cost RGB-depth (RGB-D) cameras (such as the Kinect [1]) has given path to development of algorithms that use depth cues and has immensely influenced a majority of Computer Vision application, including but not limited to Pose estimation, Object tracking, Human detection, orientation detection, action recognition etc. [24]

There is a vast range of definitions for human pose. Formally speaking, the pose refers to a set of semantically important points on the human body, usually physical joints, forming a tree, that defines the geometry of our complex articulated body structure [70, 58, 46, 34]. The points could be landmarks that can be easily distinguished from their appearances, e.g., eyes or nose on human face [30].

In this paper, we will focus on 3D Pose estimation from a single RGB and the corresponding depth image. Recently, some systems have explored the possibility of directly inferring 3D poses from images with end-to-end deep architectures [57, 48]. However, as studied in detail in Martinez et al. [42], the decoupling of 3D pose estimation into the well studied problems of 2D pose estimation, and then using its result (plus the depth image in our case) to do 3D pose estimation gives us the possibility of exploiting existing state of the art 2D pose estimation systems.

Earlier methods of 2D pose estimation used features such as silhouettes [2], shape context [44], SIFT descriptors [9] or edge direction histograms [52]. Recently though, data-hungry deep learning systems, specifically Deep Convolutional Neural Networks (CNNs) [65, 59, 49, 45, 17, 15, 13], have outperformed all other approaches. Cao et al. [13] proposed a pose estimation framework which combines an iteratively correcting variation of the confidence map based joint detection architecture with a part affinity field regression to enforce inter-joint consistency, which won the 2016 COCO person keypoints challenge [36].

Our main contribution to this problem is the design and analysis of a Deep Learning pipeline that is intuitive, easy to reproduce and performs better than the current state-of-the-art in 3D Human Pose Estimation. The increase in accuracy can be attributed to using a state of the art 2D pose estimation model by Cao [13] at the start of our pipeline and incorporating both RGB and Depth data for final 3D pose estimation. We will also be releasing our code, replicating the pipeline mentioned in this paper.

2. Related Work

2D pose estimation : The research in the field of pose estimation, especially 2D pose estimation, is really vast with a lot of surveys available, focused on different sections of the problem statement. For example, in [35], the focus of the survey is on algorithms for high-level crowd scene understanding. The survey presented in [67] summarises the advances in human body parts tracking for rehabilitation purposes. The work in [24] reviews recent Kinect-based applications in computer vision, including a very brief survey of RGB-D based trackers. In [38], a general review of multiple human tracking is presented. The review in [12] is focused on multiple human tracking using RGB-Depth data. The methods in [64, 55, 5, 21] uses only depth cues for pose estimation.

2D pose to 3D joints : The problem of inferring 3D joints from 2D pose can be traced back to the work of Lee and Chen [33]. There has been a lot of traditional Computer Vision techniques used for lifting the 2D pose to 3D joints [3, 10, 23, 29, 51, 61, 69, 70]. The recent advancements are however mainly in the Deep Learning based approaches [48, 43] which don't actually predict the final 3D joint positions directly. A major motivation behind these approaches is the idea that predicting 3D keypoints directly from 2D detections is inherently difficult. However, recent work by Martinez et al. [42] contradicts the idea and got great performance boost in trying to directly predict the 3D joint positions.

2D pose to 3D angular pose : There is a different set of algorithms for inferring 3D pose which estimate the body configuration in terms of angles (and sometimes body shape) instead of directly estimating the 3D position of the joints [7, 10, 47, 68]. This certainly reduces the dimensionality of the problem, due to the constrained mobility of human joints and the resulting estimations are forced to have a human-like structure. However these mappings between the 2D pose and the corresponsing 3D body configuration parameters is highly nonlinear and makes learning and inference harder and more computationally expensive [42].

3D pose from Depth image : Since a major drawback of using just the RGB image was occlusions, there were certain Depth, otherwise known as time of flight, based approaches introduced, with a camera in an overhead position in order to improve the detection [4, 11, 18, 19, 8, 28, 55, 20].

Multiple approaches have been developed during the last decades to incorporate the time of flight data. Most of them are focused towards using feature based Computer vision algorithm [50, 39, 40, 41]. While there were a few approaches leaning towards Machine Learning methods [6], not much has been done in terms of Deep Learning approaches for extracting 3D poses from Depth images.

3. Proposed Model

Our goal is to estimate 3D joint positions given a RGB and a corresponding calibrated Depth image. Formally, our input is -

1. A 2D matrix of size $w \times h$ where each cell is a three valued tuple, each value ranging from 0 to 255 (RGB representation).

2. A 2D matrix, again of size $w \times h$ where each cell is a float value representing its depth from the camera.

Our final output is an array of 3n float values (n is the number of joints present in the kinematic model), representing the x, y and z coordinate of each joint.

We aim to use an existing state-of-the-art 2D pose estimation model at the start of our pipeline. The results from this model, along with the depth data, will then be used to predict the final 3D joint position. The complete pipeline is discussed in detail below.

3.1. 2D Pose Estimation

For the 2D pose estimation part of our pipeline, we used the model by Cao et al. [13]. Their complete model takes, as input, a color image of size $w \times h$ and produces, as output, the 2D locations of keypoints for each person in the image. However, we only replicate their model upto the point where they provide the final heatmaps, also called confidence maps. Fig 1 illustrates the exact architecture of the 2D Pose estimation model that we used.

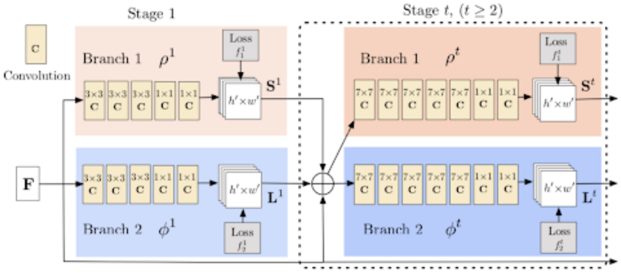

Figure 1: Architecture for the multi stage CNN. At every stage, the first branch outputs S, the heatmaps and the second branch outputs L, the PAFs. Both of these, along with the image features, are concatenated for the next stage input. Image courtesy Cao et al [13]

A feedforward network simultaneously predicts a set of 2D confidence maps S of body part locations and a set of 2D vector fields L of part affinities. The set $S = (S_1, S_2, .., S_J)$ has J confidence maps, one per joint, where $S_j \in R^{w \times h}, j \in \{1...J\}$. The set $L = (L_1, L_2, ..., L_c)$ has C vector fields, one per limb, where $L_c \in R^{w \times h \times 2}, c \in \{1...C\}$, each image location in L_c encodes a 2D vector. [13]

The input RGB image is first passed through a CNN (initialized by the first 10 layers of VGG-19 [53] and then fine-tuned), generating feature maps F which is input to the first stage of each branch. The architecture can be explained by the following equations,

$$S^1 = \rho^1(F) \tag{1}$$

$$L^1 = \phi^1(F) \tag{2}$$

$$S^t = \rho^t(F, S^{t-1}, L^{t-1}) \quad \forall t \geq 2 \tag{3}$$

$$L^t = \phi^t(F, S^{t-1}, L^{t-1}) \quad \forall t \geq 2 \tag{4}$$

where S^t and L^t are the set S and L at stage t and ρ^t and ϕ^t are the CNNs for inference at Stage t [13]. The final heatmaps are present in S^T (where 'T' is the final stage) that will be further used by our 3D pose estimation model.

To iteratively predict confidence maps and PAFs, the loss functions is applied at the end of each stage. They used L_2 loss between the estimated predictions and the groundtruth maps. Specifically, the loss function at stage t is,

$$f_S^t = \Sigma_{j=1}^{J} \Sigma_p \| \, S_j^t(p) - S_j^*(p) \, \|^2 \tag{5}$$

$$f_L^t = \Sigma_{c=1}^{C} \Sigma_p \| \, L_c^t(p) - L_c^*(p) \, \|^2 \tag{6}$$

$$f^t = f_S^t + f_L^t \tag{7}$$

where S_j^* and L_c^* are the groundtruth confidence maps and part affinity fields respectively [13]. The intermediate supervision at each stage addresses the vanishing gradient problem [63]. The overall loss function is,

$$f_{2D} = \Sigma_{t=1}^{T} f^t \tag{8}$$

3.2. 2D Pose to 3D Joints

The second half of our pipeline is based on a simple, deep, convolutional neural network with batch normalization [26], dropout [54] and an architecture similar to AlexNet [32]. The final fully connected layer produces output of size 3n. Our architecture benefits from the availability of an initial 2D pose and the perception of distance between different joints, provided by the depth image. As we will demonstrate, a multi-modal pipeline like ours can provide significant improvement over existing single modal techniques.

The input Depth image is first passed through a CNN (again initialized by the first 10 layers of VGG-19 [53] and then fine tuned), generating feature maps DF which, along with heatmaps generated previously, are fed to a pipeline containing seven CNN layers.

The pipeline contains a total of nine layers with weights; the first seven are convolutional and the remaining two are fully-connected. The output of the last fully-connected layer is of size 3n, for the x, y and z coordinate of each of the n joints. The neurons in the fully-connected layers are connected to all neurons in the previous layer. Max-pooling layers follow the first, third, fifth and seventh convolutional layer. The ReLU non-linearity is applied to the output of every convolutional but not the fully connected layers.

The first CNN layer filters the input with 64 kernels of size 7 × 7 × 3. The second CNN layer takes as input the (pooled) output of the first CNN layer and filters it with 128 kernels of size 5 × 5 × 64 followed by the third CNN layer with 128 kernels of size 5 × 5 × 128. The fourth and fifth CNN follow a similar architecture as second and third, with the input as the (pooled) output of the third CNN layer and 128 kernels of size 3 × 3 × 128 for both the layers. The sixth and seventh CNN also follow a similar architecture, with the input as the (pooled) output of the fifth CNN layer and filters it with 256 kernels of size $3 \times 3 \times 128$ in the sixth layer and 256 kernels of size 3 × 3 × 256 in the seventh layer. All the CNN layers have a stride of 1.

The first fully-connected layer takes the (pooled) output of the seventh CNN layer as its input and contains 100 neurons. The second fully-connected layer contains 3n neurons (n is the number of joints). Fig 2 illustrates the exact architecture that we used. We used L_2 loss at the final output of our model. The loss function can be defined as,

$$f_{3D} = \Sigma_{j=1}^{J} \{ (x_j^p - x_j^*)^2 + (y_j^p - y_j^*)^2 + (z_j^p - z_j^*)^2 \} \tag{9}$$

where x_j^p, y_j^p, z_j^p represent the predicted coordinate values and x_j^*, y_j^*, z_j^* represent the groundtruth coordinate values.

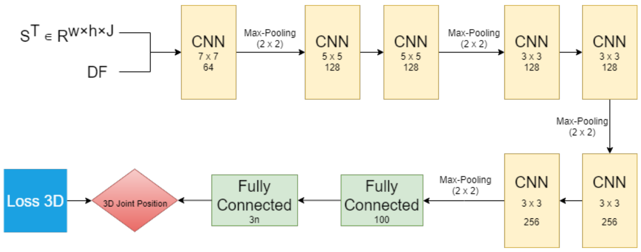

Figure 2: Architecture for our 3D pose estimation model from 2D pose. The heatmaps produced by the 2D pose estimation model, along with depth image features, are concatenated and fed as input to our network.

3.3. Complete Pipeline

A complete overview of our can be seen in Fig 3. Due to an integration of different models present in our pipeline, a strict protocol was followed for the training of the model. The following steps were taken (in this order) to achieve the mentioned performance of our model,

1. The 2D pose estimation model was seperately trained on COCO keypoints dataset using the loss function f_{2D}.

2. These learned weights were used to initialise the first half of our final pipeline. The model is now trained on a 3D pose estimation dataset like Human3.6M or SURREAL, using a combination of loss from both 2D and 3D models. The loss function used is,

$$f = \alpha \times f_{2D} + \beta \times f_{3D} \tag{10}$$

where α, β are hyperparameters tuned during training.

3.3.1 Training across datasets

The three datasets that we are working with are all built on different kinematic groundtruth models. COCO dataset

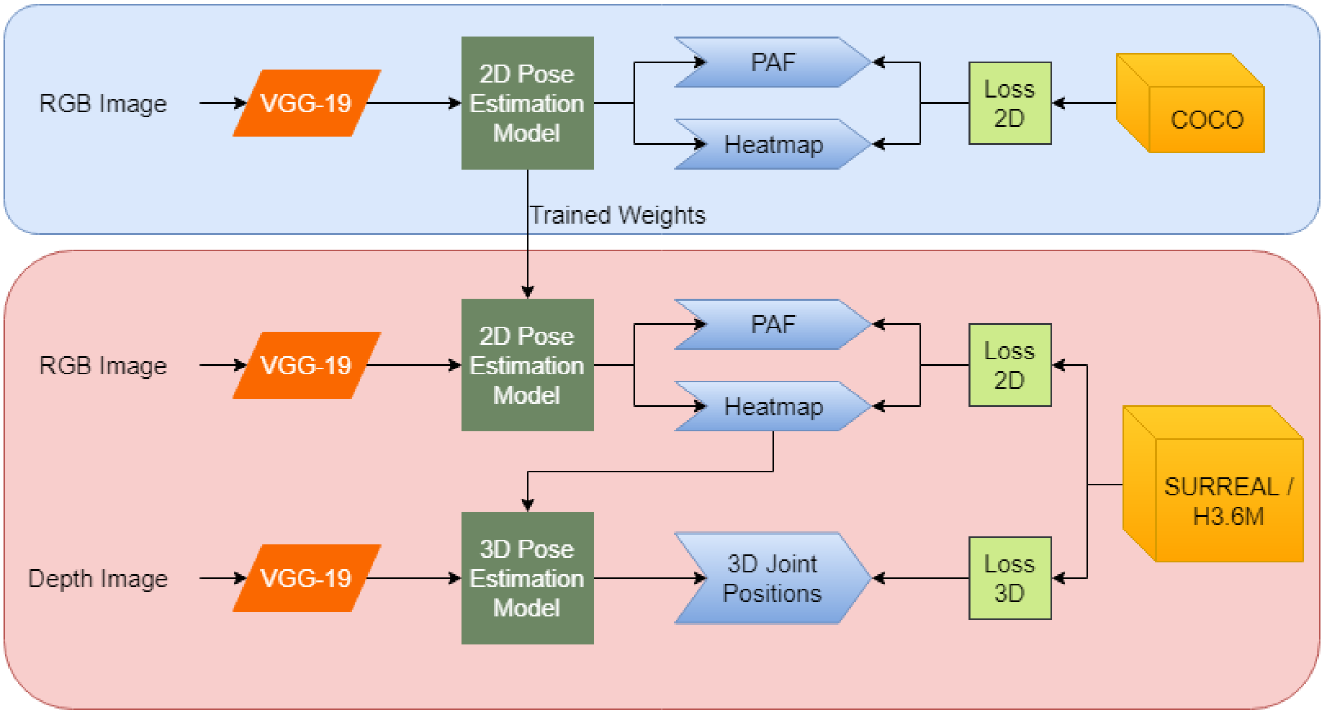

Figure 3: An overview of the complete pipeline and training structure of our model.

groundtruth is a 17 joint body model, while Human3.6M contains 32 and SURREAL contains 24 joints in their body models. Due to the architecture used by us to join the 2D and 3D pose estimation, the number of heatmaps taken as input (representing the body model of 2D dataset) and the number of 3D joint positions predicted (representing the body model of 3D dataset) are not dependent on each other. This allows us to overlook the differences between the kinematic models across 2D and 3D datasets.

Now among 3D datasets, training across different datasets cannot be done until a uniform body model is agreed upon. Since SURREAL contains a less detailed model, we brought down the body model of Human3.6M from 32 joints to 24 joints. The final 24 joint model used across all 3D datasets can be seen in Fig 4.

3.3.2 Design Choices

The motivation behind using the model by Cao et al. [13] for 2D pose estimation was to use a pipeline that outputs confidence maps for 2D pose. We hypothesize that these confidence maps, by using the depth maps, can correct themselves and shift a little, ultimately providing us with more accurate 3D joint positions.

Another choice that we had was generating outputs as 3D joint positions instead of 3D probabilities [48], 3D motion parameters [68] or basis pose coefficients and camera parameter estimation [3, 10, 51, 69, 70]. The motivation behind this was the work of Martinez et al. [42], which showed the easily trainable relation between the 2D pose and the 3D joint positions.

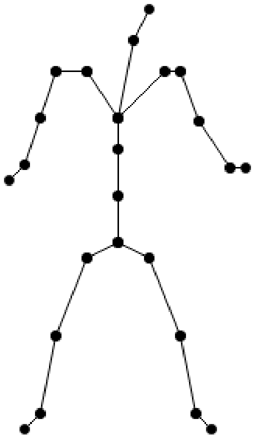

Figure 4: 24 joint skeleton : The final output of our pipeline

There has been a lot of work in recent times regarding the design of Deep Learning networks, specially in the domain of Computer Vision, like ResNet [25], Inception [56]. However, experimenting with these diverse range of designs would constitute a problem statement different from what we are focusing on. Our aim was to create a multi-modal pipeline for 3D pose estimation and so we stuck with the

Table 1: Detailed 3D Pose estimation results on Human3.6M, and comparison with previous work. Model #1 is our model trained on Human3.6M, Model #2 is pretrained on COCO and then trained on Human3.6M and Model #3 is pretrained on COCO, followed by training on both Human3.6M and SURREAL. All the previous work results were obtained from [42]

Model	Direct.	Discuss	Eating	Greet	Phone	Photo	Pose	Purch.	Sitting	SitingD	Smoke	Wait	WalkD	Walk	Avg
Zhou et al. [68]	91.8	102.4	96.7	98.8	113.4	125.2	90.0	93.8	132.2	159.0	107.0	94.4	126.0	79.0	107.3
Pavlakos et al. [48]	67.4	71.9	66.7	69.1	72.0	77.0	65.0	68.3	83.7	96.5	71.7	65.8	74.9	59.1	71.9
Martinez et al. [42]	51.8	56.2	**58.1**	59.0	69.5	78.4	55.2	58.1	74.0	94.6	62.3	59.1	65.1	49.5	62.9
Model #1	59.3	65.2	66.9	58.8	75.4	73.7	57.5	64.1	70.3	97.9	61.8	66.8	70.9	57.7	65.7
Model #2	**45.4**	50.3	59.2	**49.3**	**62.9**	72.2	**50.7**	**52.7**	**68.6**	85.5	**59.1**	52.2	**57.1**	**46.2**	**57.2**
Model #3	46.7	**49.8**	61.3	50.5	63.5	**71.9**	51.3	52.9	69.3	**84.6**	63.2	**51.7**	59.2	46.5	58.1

VGG-19 [53] architecture, as used by Cao et al [13].

3.3.3 Training Details

We train our network using Adam [31], with a starting learning rate of 1e-4 and a step decay, using mini-batches of size 32. We implemented our code using Tensorflow and ran our code on two 2x NVIDIA K40 GPUs.

4. Experiments and Results

4.1. Datasets

We evaluate our method on two benchmarks for 3D pose estimation, Human3.6M [27] and SURREAL [60]. We also used COCO 2016 keypoints challenge dataset [36] for pretraining of our model. COCO is a multi person dataset and collects images in diverse scenarios that contain many real-world challenges such as scale variation, occlusion, contact etc.

Human3.6M and SURREAL are, to the best of our knowledge, the two largest publicly available datasets for single person 3D pose estimation. They contain 3.6 million frames and 6.5 million frames of Human poses respectively, where every frame consist of a RGB image and a corresponding Depth image, along with 2D and 3D ground truth joint positions.

As standard protocol suggests, the joint coordinates in the dataset were processed using the camera parameters and aligned to the root joint (central hip joint) before proceeding. Also, since we require both RGB and Depth image as an input for our model, we were able to train and evaluate on data from only one RGB camera (the one algined with the Depth camera), despite multiple cameras present in these datasets.

4.2. 2D Pose estimation

Since our final 3D pose estimation model also produces intermediate 2D pose results, we compared them with the existing state-of-the-art 2D pose estimation models. We trained two different 2D pose estimation models, Stacked Hourglass network [45] and Cao et al [13] using the SURREAL dataset. The results (Table 2) were compared with the 2D pose predicted by our complete pipeline trained on COCO and SURREAL.

Table 2: 2D Pose estimation results on SURREAL dataset

Model Name	AP	AP^{50}	AP^{75}
Stacked Hourglass [45]	59.1	77.7	61.2
Cao et al [13]	62.3	81.1	64.7
Our Model (Trained on COCO + SURREAL)	**65.3**	**82.1**	**69.9**

The results show an increase in the 2D pose estimation performance of our complete pipeline as compared to a standalone 2D pose estimation model of Cao et al [13] (which we have incorporated in our pipeline). This can be attributed to the 3D loss that gets backpropogated to the 2D pose estimation architecture, thus fine tuning the weights even further.

4.3. 3D Pose estimation

Value of α, β : The hyperparameters $\alpha.\beta$, as defined in equation 10, were chosen to ensure that the scaling of 2D and 3D losses remain the same. After reviewing the 2D loss and 3D loss during the final pipeline training, we picked the value as $\alpha = 7 \times 10^{-6}, \beta = 1$. Slight variations in the values of α does not substantially affect the performance of the model, however the parameter values were picked by doing a grid search (although not at a very granular level).

Most of the models proposed give good performances on datasets on which they were trained. However, switching to a different dataset requires retraining the model. So, with the intention of creating a single model that can, once trained, give competitive results across different datasets, we tried a variety of protocols, using combinations of all the datasets available to us. The compiled results can be found in Table 3. Detailed results for Human3.6M can be found in Table 1.

On Human3.6M we follow the standard protocol, using subjects 1, 5, 6, 7 and 8 for training, 9 and 11 for evaluation. The Mean average error across all the joint predictions is reported, after alignment of the root (central hip) joint. For SURREAL, the dataset is already partitioned into train, val and test and the same protocol, as mentioned above, was used for evaluation.

Our model achieved a peak performance of 57.2 mm of error on Human3.6M, when pretrained on COCO followed by training on Human3.6M. This results gives an improvement of 5.7 mm of error over best reported results that we

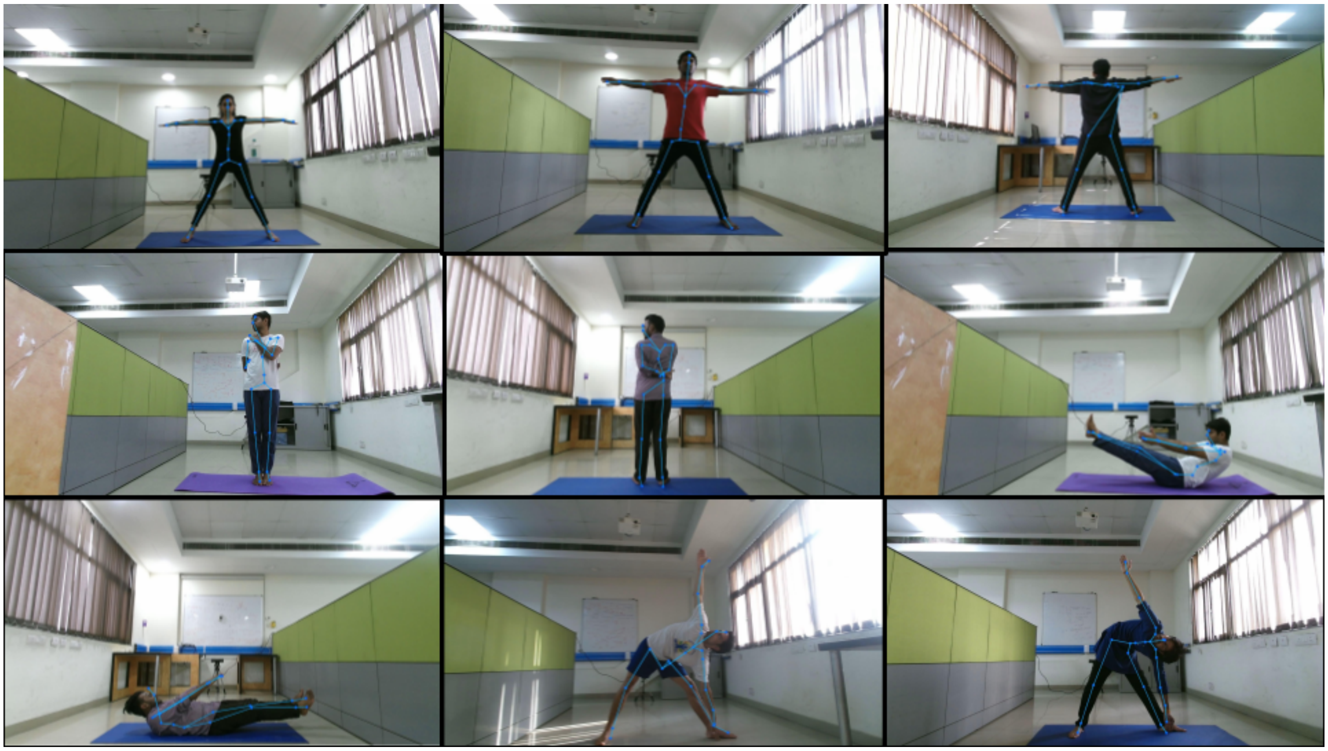

Figure 5: A qualitative analysis of our model. The images present here (left to right, top to bottom) represents, two images of subjects standing at a standard position from front camera and one from back camera, and then alternatively one image from front and one from back camera of subjects doing Katichakrasana and Naukasana respectively, followed by two images of subjects doing Trikonasana.

Table 3: 3D Pose estimation results, and comparison with previous work. All the previous work results were obtained from [42]

Model Name	**Trained on**			**Results (Error in mm)**	
	COCO	H3.6M	SURREAL	H3.6M	SURREAL
Pavlakos et al [48]		✓		71.9	N/A
Martinez et al [42]		✓		62.9	N/A
Martinez et al [42]			✓	N/A	82.3
Our Model		✓		65.7	86.4
Our Model			✓	65.8	84.7
Our Model	✓	✓		**57.2**	76.5
Our Model	✓		✓	60.4	**72.3**
Our Model	✓	✓	✓	58.1	72.5

are aware of [42]. Similarly, we achieved a peak performance of 72.3 mm of error on SURREAL, when pretrained on COCO followed by training on SURREAL. Again, the result gives an improvement of 10 mm of error over the best we are aware of [42].

However, the performance of the above two models on the datasets on which they were not trained, deterioates significantly. Thus, we trained a model on all the three datasets, pretrained on COCO and then trained on SURREAL and Human3.6M (in this order). The results obtained are clearly competitive with the results obtained on separate training. This model is hypothesized to perform better on unseen poses, since it has seen a wider range of poses during its training (both real and synthetic).

4.4. Qualitative Analysis

Finally, we show some qualitative results on images from our lab experiments in Figure 5. The figure shows estimated 3D poses projected onto the input images. The subjects in these images are performing different Yoga Asanas. The Human Pose during Yoga Asanas consist of unusual body extensions and heavy occlusions, never seen in any of the conventional datasets, like Human3.6M or SURREAL. We wanted to inspect the results of our model in case of such unusual poses. The results show that our model is capable of working with never seen before poses, and can handle such unusual extensions and light occlusions quite well.

However, there are certain cases visible, specifically containing heavy occlusions, in which our model fails to predict a reasonably correct pose, thus highlighting some of the limitations of our current approach. Also, it can be noticed that the poses estimated for the back camera are clearly not correct, even when the incorrect joint is actually clearly visible, but just from a different profile. This shows the importance of RGB features, or in other words visual features, in our approach, since our base 2D pose estimation model is based on just the RGB input. This indicates a possibility of approaching this problem by starting with Depth image and then incorporating RGB features, which can possibly get rid of such cases.

5. Discussion

5.1. Implications of our results

We have demonstrated that a naive, intuitive and easy to implement Deep CNN based approach of combining both RGB and Depth data for 3D pose estimation can achieve a remarkably low error rate. When working with a state-of-the-art 2D pose estimation model, our pipeline was able to achieve the best results in 3D Human Pose estimation to date.

Our results provide support to the range of algorithms that break down the 3D pose estimation problem into two separate problems, 2D pose estimation and obtaining 3D pose from 2D pose. This supports the results of Martinez et al [42], and stand in contrast to recent work, focused on end-to-end systems trained from pixels to 3d positions.

Another very important aspect of our pipeline is its multi-modality. Being able to incorporate both RGB image (in the form of 2D pose) and Depth image into a pose estimation pipeline, using a simple Deep CNN architecture gave us the best results to date. This immensely highlights the potential of such multi-modal pipelines. It would be interesting to see experiments with different ways of creating a multi-modal pipeline and choosing from a variety of CNN architectures available to boost the performance even further, which we leave for future work.

5.2. Further Improvement

The simplicity of the root ideas of our system springs open multiple directions of improvement to our model. For example, one aspect of research could be the architecture used. Different types of Deep CNN based architectures, for example Inception [56], Resnet [25] etc., can be tried to further improve the pose estimation performance.

Another aspect of improvement in our model is the way we used the depth data. We used the depth data in its raw form, letting our model decipher the necessary features from it. However, one can also preprocess the Depth images and create more meaningful features from it, which can be fed into our model. This can improve the performance and might even allow us to decrease the computational complexity of the model.

A post processing model, used after the final output prediction, is used to reject unnatural poses and possibly correct the predicted pose. No such post-processing pipeline is present in our model. Adding such a pipeline can also be a vast area of research, forcing the model to predict more human like poses.

6. Conclusion and Future Work

We have shown that a simple and intuitive Deep convolutional neural network can achieve surprisingly accurate results in 3D human pose estimation. The multi-modality of our pipeline, coupled with a state-of-the-art 2D pose estimation model, helps us create an easy-to-reproduce, yet high-performance baseline that outperforms the state of the art in 3D human pose estimation.

Our work in simply extending the 2D pose estimated using RGB images, into a 3D pose by using the depth image as an additional input, strongly suggests that the two types of input data, RGB and Depth, complement each other and can provide great leaps in 3D pose estimation and other related fields.

Moreover, we use 3D joint coordinates as our final output, suggesting that the more complex representations of the human body, like using joint angle representation, might either not be crucial, or have not been exploited to its full potential. Finally, given the simplicity and intuitive design of our pipeline, we do not think of it as a full-fledged 3d pose estimation system, but instead a start towards a new set of models and research. This ignites motivation to multiple directions of future work.

For one, our network currently does not incorporate any kind of temporal features, working on every image independently. This can sometimes cause jitter while predicting 3D pose in a video. Another possibility can be extending our model to multi person 3D pose estimation. This can have vast implications and is an important field of research. We only touched a little on the improvement of 2D pose estimation using depth images in this paper. However, the results were promising and incorporating both the modalities for 2D pose estimation can also be a promising prospective. These are all interesting areas of future work.

References

[1] Microsoft corporporation. kinect for xbox 360, 2009. 1

[2] A. Agarwal and B. Triggs. 3d human pose from silhouettes by relevance vector regression. In *null*, pages 882–888. IEEE, 2004. 1

[3] I. Akhter and M. J. Black. Pose-conditioned joint angle limits for 3d human pose reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1446–1455, 2015. 2, 4

[4] B. Antić, D. Letić, D. Ćulibrk, and V. Crnojević. K-means based segmentation for real-time zenithal people counting. In *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pages 2565–2568. IEEE, 2009. 2

[5] T. Bagautdinov, F. Fleuret, and P. Fua. Probability occupancy maps for occluded depth images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2829–2837, 2015. 2

[6] Y. Bar-Shalom, P. K. Willett, and X. Tian. *Tracking and data fusion*. YBS publishing Storrs, CT, USA:, 2011. 2

[7] C. Barrón and I. A. Kakadiaris. Estimating anthropometry and pose from a single uncalibrated image. *Computer Vision and Image Understanding*, 81(3):269–284, 2001. 2

[8] A. Bevilacqua, L. Di Stefano, and P. Azzari. People tracking using a time-of-flight depth sensor. In *Video and Signal Based Surveillance, 2006. AVSS'06. IEEE International Conference on*, pages 89–89. IEEE, 2006. 2

[9] L. Bo, C. Sminchisescu, A. Kanaujia, and D. Metaxas. Fast algorithms for large scale conditional 3d prediction. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008. 1

[10] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black. Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In *European Conference on Computer Vision*, pages 561–578. Springer, 2016. 2, 4

[11] Z. Cai, Z. L. Yu, H. Liu, and K. Zhang. Counting people in crowded scenes by video analyzing. In *Industrial Electronics and Applications (ICIEA), 2014 IEEE 9th Conference on*, pages 1841–1845. IEEE, 2014. 2

[12] M. Camplani, A. Paiement, M. Mirmehdi, D. Damen, S. Hannuna, T. Burghardt, and L. Tao. Multiple human tracking in rgb-depth data: a survey. *IET computer vision*, 11(4):265–285, 2016. 2

[13] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. *arXiv preprint arXiv:1611.08050*, 2016. 1, 2, 3, 4, 5

[14] F. Cardinaux, D. Bhowmik, C. Abhayaratne, and M. S. Hawley. Video based technology for ambient assisted living: A review of the literature. *Journal of Ambient Intelligence and Smart Environments*, 3(3):253–269, 2011. 1

[15] J. Carreira, P. Agrawal, K. Fragkiadaki, and J. Malik. Human pose estimation with iterative error feedback. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4733–4742, 2016. 1

[16] A. A. Chaaraoui, P. Climent-Pérez, and F. Flórez-Revuelta. A review on vision techniques applied to human behaviour analysis for ambient-assisted living. *Expert Systems with Applications*, 39(12):10873–10888, 2012. 1

[17] X. Chu, W. Ouyang, H. Li, and X. Wang. Structured feature learning for pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4715–4723, 2016. 1

[18] B.-K. Dan, Y.-S. Kim, J.-Y. Jung, S.-J. Ko, et al. Robust people counting system based on sensor fusion. *IEEE transactions on consumer electronics*, 58(3):1013–1021, 2012. 2

[19] L. Del Pizzo, P. Foggia, A. Greco, G. Percannella, and M. Vento. Counting people by rgb or depth overhead cameras. *Pattern Recognition Letters*, 81:41–50, 2016. 2

[20] A. Fernandez-Rincon, D. Fuentes-Jimenez, C. Losada-Gutierrez, M. M. Romera, C. A. Luna, J. M. Guarasa, and M. Mazo. Robust people detection and tracking from an overhead time-of-flight camera. In *VISIGRAPP (4: VISAPP)*, pages 556–564, 2017. 2

[21] B. Fosty, C. F. Crispim-Junior, J. Badie, F. Bremond, and M. Thonnat. Event recognition system for older people monitoring using an rgb-d camera. In *ASROB-workshop on assistance and service robotics in a human environment*, 2013. 2

[22] D. Geronimo, A. M. Lopez, A. D. Sappa, and T. Graf. Survey of pedestrian detection for advanced driver assistance systems. *IEEE transactions on pattern analysis and machine intelligence*, 32(7):1239–1258, 2010. 1

[23] A. Gupta, J. Martinez, J. J. Little, and R. J. Woodham. 3d pose from motion for cross-view action recognition via nonlinear circulant temporal encoding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2601–2608, 2014. 2

[24] J. Han, L. Shao, D. Xu, and J. Shotton. Enhanced computer vision with microsoft kinect sensor: A review. *IEEE transactions on cybernetics*, 43(5):1318–1334, 2013. 1

[25] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4, 7

[26] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015. 3

[27] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu. Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE transactions on pattern analysis and machine intelligence*, 36(7):1325–1339, 2014. 5

[28] L. Jia and R. J. Radke. Using time-of-flight measurements for privacy-preserving tracking in a smart room. *IEEE Transactions on Industrial Informatics*, 10(1):689–696, 2014. 2

[29] H. Jiang. 3d human pose reconstruction using millions of exemplars. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 1674–1677. IEEE, 2010. 2

[30] A. Jourabloo and X. Liu. Large-pose face alignment via cnn-based dense 3d model fitting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4188–4196, 2016. 1

[31] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[32] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 3

[33] H.-J. Lee, C. Zen, et al. Determination of 3d human-body postures from a single view. *Computer Vision Graphics and Image Processing*, 30(2):148–168, 1985. 2

[34] S. Li and A. B. Chan. 3d human pose estimation from monocular images with deep convolutional neural network. In *Asian Conference on Computer Vision*, pages 332–347. Springer, 2014. 1

[35] T. Li, H. Chang, M. Wang, B. Ni, R. Hong, and S. Yan. Crowded scene analysis: A survey. *IEEE transactions on circuits and systems for video technology*, 25(3):367–386, 2015. 1

[36] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 1, 5

[37] W.-L. Lu, J.-A. Ting, J. J. Little, and K. P. Murphy. Learning to track and identify players from broadcast sports videos. *IEEE transactions on pattern analysis and machine intelligence*, 35(7):1704–1716, 2013. 1

[38] W. Luo, X. Zhao, and T.-K. Kim. Multiple object tracking: A review. *arXiv preprint arXiv:1409.7618*, 1, 2014. 2

[39] J. MacCormick and A. Blake. A probabilistic exclusion principle for tracking multiple objects. *International Journal of Computer Vision*, 39(1):57–71, 2000. 2

[40] M. Marrón, J. C. García, M. A. Sotelo, D. Fernández, and D. Pizarro. " xpfcp": an extended particle filter for tracking multiple and dynamic objects in complex environments. In *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*, pages 2474–2479. IEEE, 2005. 2

[41] M. Marrón-Romera, J. C. García, M. A. Sotelo, D. Pizarro, M. Mazo, J. M. Cañas, C. Losada, and Á. Marcos. Stereo vision tracking of multiple objects in complex indoor environments. *Sensors*, 10(10):8865–8887, 2010. 2

[42] J. Martinez, R. Hossain, J. Romero, and J. J. Little. A simple yet effective baseline for 3d human pose estimation. In *International Conference on Computer Vision*, volume 1, page 5, 2017. 1, 2, 4, 5, 6, 7

[43] F. Moreno-Noguer. 3d human pose estimation from a single image via distance matrix regression. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 1561–1570. IEEE, 2017. 2

[44] G. Mori and J. Malik. Recovering 3d human body configurations using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(7):1052–1062, 2006. 1

[45] A. Newell, K. Yang, and J. Deng. Stacked hourglass networks for human pose estimation. In *European Conference on Computer Vision*, pages 483–499. Springer, 2016. 1, 5

[46] M. Oberweger, P. Wohlhart, and V. Lepetit. Hands deep in deep learning for hand pose estimation. *arXiv preprint arXiv:1502.06807*, 2015. 1

[47] V. Parameswaran and R. Chellappa. View independent human body pose estimation from a single perspective image. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–II. IEEE, 2004. 2

[48] G. Pavlakos, X. Zhou, K. G. Derpanis, and K. Daniilidis. Coarse-to-fine volumetric prediction for single-image 3d human pose. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 1263–1272. IEEE, 2017. 1, 2, 4, 5, 6

[49] L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. V. Gehler, and B. Schiele. Deepcut: Joint subset partition and labeling for multi person pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4929–4937, 2016. 1

[50] R. Poppe. Condensation-conditional density propagation for visual tracking. *Comput. Vis. Image Underst*, 108:4–18, 2007. 2

[51] V. Ramakrishna, T. Kanade, and Y. Sheikh. Reconstructing 3d human pose from 2d image landmarks. In *European conference on computer vision*, pages 573–586. Springer, 2012. 2, 4

[52] G. Shakhnarovich, P. Viola, and T. Darrell. Fast pose estimation with parameter-sensitive hashing. In *null*, page 750. IEEE, 2003. 1

[53] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 2, 3, 5

[54] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014. 3

[55] C. Stahlschmidt, A. Gavriilidis, J. Velten, and A. Kummert. Applications for a people detection and tracking algorithm using a time-of-flight camera. *Multimedia Tools and Applications*, 75(17):10769–10786, 2016. 2

[56] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *AAAI*, volume 4, page 12, 2017. 4, 7

[57] B. Tekin, I. Katircioglu, M. Salzmann, V. Lepetit, and P. Fua. Structured prediction of 3d human pose with deep neural networks. *arXiv preprint arXiv:1605.05180*, 2016. 1

[58] B. Tekin, A. Rozantsev, V. Lepetit, and P. Fua. Direct prediction of 3d body poses from motion compensated sequences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 991–1000, 2016. 1

[59] A. Toshev and C. Szegedy. Deeppose: Human pose estimation via deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1653–1660, 2014. 1

[60] G. Varol, J. Romero, X. Martin, N. Mahmood, M. J. Black, I. Laptev, and C. Schmid. Learning from synthetic humans. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017)*, pages 4627–4635. IEEE, 2017. 5

[61] C. Wang, Y. Wang, Z. Lin, A. L. Yuille, and W. Gao. Robust estimation of 3d human poses from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2361–2368, 2014. 2

[62] X. Wang. Intelligent multi-camera video surveillance: A review. *Pattern Recognition Letters*, 34(1):3 – 19, 2013. Extracting Semantics from Multi-Spectrum Video. 1

[63] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. Convolutional pose machines. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4724–4732, 2016. 3

[64] L. Xia, C.-C. Chen, and J. K. Aggarwal. Human detection using depth information by kinect. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2011 IEEE Computer Society Conference on*, pages 15–22. IEEE, 2011. 2

[65] W. Yang, W. Ouyang, H. Li, and X. Wang. End-to-end learning of deformable mixture of parts and deep convolutional neural networks for human pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3073–3082, 2016. 1

[66] X. Zabulis, D. Grammenos, T. Sarmis, K. Tzevanidis, P. Padeleris, P. Koutlemanis, and A. A. Argyros. Multicam-

era human detection and tracking supporting natural interaction with large-scale displays. *Machine Vision and Applications*, 24(2):319–336, Feb 2013. 1

[67] H. Zhou and H. Hu. Human motion tracking for rehabilitationa survey. *Biomedical Signal Processing and Control*, 3(1):1–18, 2008. 1

[68] X. Zhou, X. Sun, W. Zhang, S. Liang, and Y. Wei. Deep kinematic pose regression. In *European Conference on Computer Vision*, pages 186–201. Springer, 2016. 2, 4, 5

[69] X. Zhou, M. Zhu, S. Leonardos, and K. Daniilidis. Sparse representation for 3d shape estimation: A convex relaxation approach. *IEEE transactions on pattern analysis and machine intelligence*, 39(8):1648–1661, 2017. 2, 4

[70] X. Zhou, M. Zhu, S. Leonardos, K. G. Derpanis, and K. Daniilidis. Sparseness meets deepness: 3d human pose estimation from monocular video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4966–4975, 2016. 1, 2, 4