Distilling 3D Human Pose from a Single Image

Meet Vora
D-INFK, ETH Zurich
voram@student.ethz.ch

ABSTRACT

In this work, we demonstrate that 3D poses from a single RGB image can be effectively estimated with an end-to-end model that makes use of a two-stage approach. We first estimate the 2D keypoints, following which we use fully-connected layers to estimate our 3D pose. While our work is inspired by some of the prior work, we propose a cyclic loss which improves generalization and gives us an MPJPE score of 89.9. We perform various ablation studies to demonstrate the effectiveness of our model and loss.

1 INTRODUCTION

In recent years, visual understanding, such as object and scene recognition, has witnessed a significant bloom thanks to deep visual representations. Modeling and understanding human behaviour in images, particularly, person detection and pose estimation, has been the focus of a variety of visual tasks due to its importance for numerous practical applications. Specifically, the task of detecting 3D human poses is of great importance in many areas including robotics, human computer interaction and autonomous driving. However, most of our data sources capture human poses in two dimensions. Thus, given the large amounts of annotated data in form of 2D images, estimating 3D poses from a single RGB image is an important challenge to overcome.

Formally, given an RGB image, we aim to produce a 3-dimensional figure that matches the spatial arrangement of a depicted person. In order to estimate the 3D coordinates, a model must learn to be invariant to various factors – illumination, background, occlusions, scale, among others. Previous work tackled these problems by making use of novel CNN architectures which are trained end-to-end. These work hypothesize that various image features are helpful for estimating the correct pose. However, more recent work has been making use of a two-staged approach, where models first predict 2D poses and then estimate 3D poses from the 2D values, without any image features. We follow the same idea as this approach gives us flexibility to explore various state-of-the-art 2D pose estimation approaches, which have been modelled to tackle the invariance challenges.

However, it is important to note the inherent ambiguities in single-view 2D-to-3D mapping. The most common problem is that of depth ambiguity where multiple 3D poses map to the same 2D projections. Many of these configurations may not be anthropometrically reasonable, such as impossible joint angles or lengths.

In this paper, we present a convolutional architecture that is inspired by the current state-of-the-art architecture for 2D pose estimation and extend it to 3D pose estimation using a cyclic loss term that acts as a regularizer and "distills" the 3D pose space to feasible arrangements.

Legi: 18-950-139
Team: MainBhiChowkidar

We train and evaluate our model on images extracted from Human3.6M dataset. The dataset was collected using marker-based motion capture systems and actors dressed with moderately realistic clothing, viewed against indoor backgrounds. It consists of 17 different scenarios depicted in a total of 3.6 million frames, of which we use only 32,000 image samples.

2 PRIOR WORK

2.1 End-to-end monocular 3D pose estimation

The monocular 3D human pose estimation problem is to learn the 3D structure of a human skeleton from a single image or a video, without using depth information or multiple views. As stated previously, it is a severely ill-posed problem. Li and Chan [4] were the first to show that deep neural networks can achieve a reasonable accuracy in 3D human pose estimation from a single image. They used two deep regression networks and body part detection. Tekin et al. [9] show that combining traditional CNNs for supervised learning with auto-encoders for structure learning can yield good results. Contrary to common regression practice, Pavlakos et al. [6] were the first to consider 3D human pose estimation as a 3D keypoint localization problem in a voxel space. Some ideas ([7], [2], [10]) deal with videos and make use of the temporal information.

2.2 2D pose to 3D pose

Recent research works have approached the problem of estimating 3D poses from 2D poses, which are learned apriori from images. These methods employ a state-of-the-art 2D pose detector to estimate 2D poses. Using a 2D pose estimator provides the required invariance to background clutter, clothing variation, among others.

Further, a two-staged approach also makes it possible to infer the accuracy of "lifting" ground truth 2D pose to 3D. The current state-of-the-art, by Martinez et al [5] uses a simple deep feedforward network that takes a 2D human pose as input and estimates 3D pose with very high accuracy. Their results suggest the effectiveness of decoupling the 3D pose estimation problem into two separate problems - namely, 2D pose estimation from an image and 3D pose estimation from a 2D pose. Their 3D pose detector trained on ground-truth 3D poses achieved a remarkable improvement in accuracy (30 %), leading to the implication that the accuracy of 2D pose estimation remains a bottleneck in end-to-end 3D pose estimation.

3 PROPOSED SOLUTION

Our proposed model makes use of two existing works - High Resolution Net, by Sun et al [8] and a simple feedforward network, as proposed in [5].

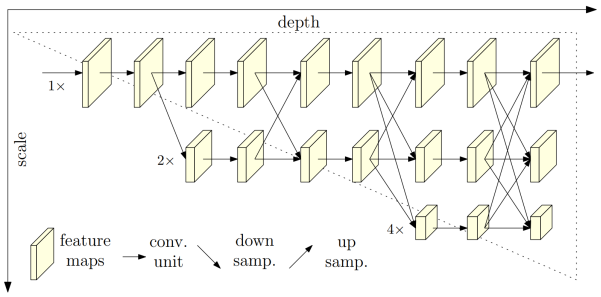

Figure 1: Architecture of High Resolution Net.

3.1 High Resolution Net

In [8], the authors present HighResolution Net (HRNet), a novel architecture that is able to maintain high-resolution representations through the entire process. Starting from a high-resolution subnetwork as the first stage, the model gradually adds various high-to-low resolution subnetworks one-by-one to form more stages, and connect the multi-resolution subnetworks in parallel. This allows the model to perform repeated multi-scale fusion by exchanging information across the parallel multi-resolution subnetworks over-and-over. As the model can retain the original resolution with the predicted final keypoints, one can expect the predicted heatmaps to be spatially more precise.

The HRNet implementation used in our work contains four stages with four parallel subnetworks, whose the resolution is gradually decreased to a half and accordingly the number of channels is increased to the double. The first stage contains 4 residual units where each unit, similar to ResNet-50, is formed by a bottleneck, and is followed by a convolution reducing the number of feature maps to 32. The 2^{nd}, 3^{rd} and 4^{th} stages contain 1, 4, 3 exchange blocks respectively. One exchange block contains 4 residual units. After the final fusion layer, we simply train the network using MSE error over the heatmap predictions. Thus, given J joints, we obtain a volume $(J \times W \times H)$ of heatmaps with activations $\hat{a}^j_{h,w}$, which we try to optimize using the objective:

$$L^{2D}_{jointMSE} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{J}\sum_{j=1}^{J}\sum_{h=1}^{H}\sum_{w=1}^{W}(\hat{a}^j_{h,w} - a^j_{h,w})^2 \tag{1}$$

3.2 Heatmap Regressor

Inspired by [5], our work also makes use of a simple feedforward network to regress 3D coordinates from 2D points. Thus, to facilitate end-to-end learning, we need a differentiable mechanism to extract the coordinates of peak activations from heatmap outputs of HRNet. We make use of a soft-argmax layer to do so. First, we flatten a given heatmap to $h \in R^{W\times H}$ and apply a softmax layer (ϕ) with high temperature (β). Further, we use a vector $\rho \in R^{W\times H}$ which is simply equivalent to `range`$(W \times H)$. A dot product between ρ and ϕ then gives us the most likely coordinate of peak activation in the flattened heatmap h.

$$\phi(a^j_i) = \frac{e^{\beta a^j_i}}{\sum_{k=1}^{W\times H} e^{\beta a^j_k}} = \phi^i_j \tag{2}$$

$$\phi^T_j = \{\phi^0_j, \phi^1_j, \phi^2_j, \ldots, \phi^{W\times H-1}_j\}$$

$$\rho^T = \{0, 1, 2, 3, \ldots, W \times H - 1\} \tag{3}$$

$$\tau_j = \rho^T \phi_j \tag{4}$$

$$(x_j, y_j) = (mod(\tau_j, W), \tau_j/H) \tag{5}$$

Thus, using equations (2 - 5), we can extract the 2D coordinates $\hat{Y}_{2D} \in R^{2\times J}$ from given a volume of heatmaps $(J \times W \times H)$ using a differentiable function. $\hat{Y}_{2D}$ is then fed into the 2D pose autoencoder.

3.3 2D Pose Autoencoder

We make use of an autoencoder architecture that learns the 3D human pose $Y_{3D} \in R^{3\times J}$ (a set of J body joint locations in 3-dimensional space), given the 2D pose $Y_{2D} \in R^{2\times J}$, in a fully-supervised setup. The 2D pose also helps as the decoder translates the 3D values back to 2D space. The proposed network architecture is illustrated in Figure 2.

3.3.1 Encoder. The network consists of a 2D-to-3D encoder for predicting $\hat{Y}_{3D}$ from $\hat{Y}_{2D}$. The encoder follows feedforward architecture proposed by Martinez et al [5]. It consists of two modules, each consisting of two blocks. Within each block is a sequence of linear layer with batch normalization, followed by ReLU activation and dropout. The output of the final block makes use of a skip connection with the input to the module.

$$L_{encoder} = \frac{1}{N}\sum_{i=1}^{N}(\hat{Y}_{3D} - Y_{3D})^2 \tag{6}$$

3.3.2 Decoder. The 3D-to-2D decoder aligns the 2D re-projection $\hat{Y}^{2D}_{proj}$ of the predicted 3D pose $\hat{Y}_{3D}$. The decoder follows the same architecture as the encoder. Lastly, to facilitate training of HRNet, we employ a skip connection from output of heatmap regressor to the output of decoder. Experimentally we see that adding skip connection helps us drastically improve our score.

$$L_{decoder} = \frac{1}{N}\sum_{i=1}^{N}(\hat{Y}^{2D}_{reproj} - \hat{Y}_{2D})^2 \tag{7}$$

3.3.3 Bone Symmetry constraint. To ensure symmetry between contralateral segments of the predicted 3D pose, bone length symmetry loss is used. Various pairs (P) like (knee, foot) are identified for each side of the body and bone lengths (b) are calculated for each such pair. The loss tries to constraint the 3D space such that the bone lengths are same for each side (left: L, right: R) of the predicted 3D pose.

$$L^{3D}_{boneSymm} = \frac{1}{N}\sum_{i=1}^{N}(\frac{1}{|P|}\sum_{k\in P}(b^L_k - b^R_k)^2) \tag{8}$$

Thus, we define our overall loss as

$$\begin{aligned} L &= \alpha L^{2D}_{jointMSE} + \beta L^{3D}_{encoder} + \gamma L^{2D}_{decoder} + \delta L^{3D}_{boneSymm} \\ &= L_{3D} + L_{Reg} \end{aligned} \tag{9}$$

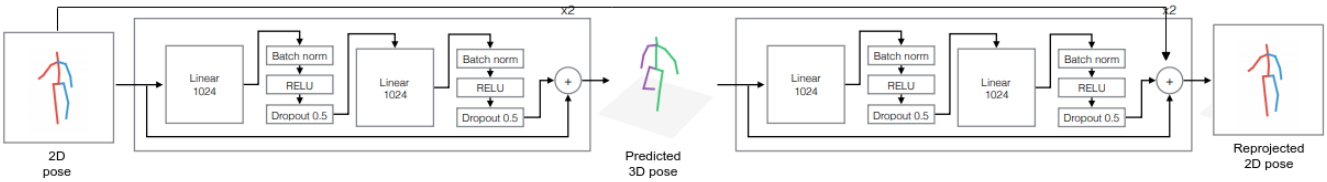

Figure 2: Architecture of 2D Pose Autoencoder.

where $(\alpha, \beta, \gamma, \delta)$ are scalars. We see that $L^{2D}_{jointMSE}$ and $L_{encoder}$ are responsible for estimating 3D poses from a given image (L_{3D}), whereas $L_{decoder}$ and $L^{3D}_{boneSymm}$ together act as a regularizer (L_{Reg}) that "distills" our 3D space.

4 TRAINING

Ground-truth Heatmaps. To train our High Resolution Net, we convert ground-truth 2D values to heatmaps. A given coordinate is set to 1 in a $W \times H$-dimensional zero matrix. This is followed by 2D Gaussian smoothing to reduce penalty for slightly incorrect predictions than completely incorrect ones. We use a kernel of size 3 with standard deviation of 1.

Augmentation. Experimentally we see that training HRNet on our dataset easily leads to overfitting. To overcome this, we use augmentation techniques. Firstly, we randomly jitter the brightness, contrast and saturation of images, followed by horizontal flipping. Thus, we also flip the 2D pose coordinates – thus, 2D coordinates for a left joint j_l is swapped with its right counterpart j_r.

HRNet Initialization. To provide for a good initialization of our model, we first train only the HRNet on $L^{2D}_{jointMSE}$ using our ground-truth heatmaps and augmentation strategies. While image normalization is a standard preprocessing step, we apply horizontal flipping only for HRNet training – thus, we do not need to augment our 3D pose coordinates. Experimentally we see that training HRNet for one epoch using Adam and batch size 32 is good enough for our results.

End-to-end training. We use the weights from HRNet training to initialize our HRNet module and use a normal distribution to sample weights for the 2D pose autoencoder. During end-to-end training, we also make use of color jittering, followed by image normalization. Horizontal flipping is avoided in this setting. We use Adam to optimize our objective L and train our complete model with a batch size of 32 for 8 epochs for the best reported results. We set the learning rate to 0.001 and evaluate the performance of the model on public leaderboard after every epoch.

5 EVALUATION

We conduct ablation studies to study the effectiveness of the proposed model as well as the loss terms. The reported MPJPE scores in table1 are noted from public leaderboard scores as they've been used as the validation set in our setup.

Knowledge Distillation is a concept proposed by Hinton et al [3]. The term distill is used as a wordplay here as [3] proposes temperature-controlled softmax, which is essential for our end-to-end training.

Loss / Model	MPJPE
$L_{encoder}$	170.84
L_{3D}	111.18
$L_{3D} + L_{decoder}$	123.51
$L_{3D} + L^{3D}_{boneSymm}$	99.83
$L_{3D} + L_{Reg}$	98.46
$L_{3D} + L_{Reg}$+ skip-connection	**89.96**

Table 1: Results from ablation study

6 DISCUSSION

Starting from a primitive model that consists of no decoder, we experiment with $L^{2D}_{jointMSE}$ and $L_{encoder}$. We see that simply training the model using $L_{encoder}$ drastically underperforms as learning correct 2D coordinates through soft-argmax without any intermediate supervision is a challenging task for the model. This is circumvented by (i) pre-training HRNet with augmentation (ii) adding $L_{jointMSE}$ as intermediate supervision, which helps us beat the hard baseline. However, we observe that the model has high variance. In order to increase robustness, we employ $L_{decoder}$. Surprisingly, this performs worse than simple L_{3D}. We hypothesize that the autoencoder structure makes it difficult for the HRNet to learn, as direct increase in depth can lead to vanishing / exploding gradients. Thus, we also experiment with a simpler regularizer ($L^{3D}_{boneSymm}$) which leads to better results. On merging all our loss terms, we observe a slight improvement in our score. To address the issue of exploding / vanishing gradients, we add a residual (skip) connection from the output of heatmap regressor to the output of decoder. We see that addition of skip-connection leads to a significant improvement in our results, validating our hypothesis. Further, we also observe that adding skip-connection leads to better resistance from overfitting. We also experiment with decaying α after every epoch, with the goal of improving autoencoder performance. However, we observe negligible difference in performance.

7 CONCLUSION AND FUTURE WORK

We follow a two-stage approach and combine the best performing model for each task using a soft-argmax layer. We further extend these models using a pose autoencoder to improve robustness. In the process, we hypothesize the impact of various factors and propose ideas, which help us outperform a simple merging of the two state-of-the-art models.

We wish to further improve the robustness of our model and evaluate on in-the-wild datasets. Another potential direction is to incorporate richer datasets (MPII [1]) which can improve the performance of HRNet and also attempt techniques like adversarial learning to improve "lifting" performance.

REFERENCES

[1] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2014. 2D Human Pose Estimation: New Benchmark and State of the Art Analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[2] Rishabh Dabral, Anurag Mundhada, Uday Kusupati, Safeer Afaque, Abhishek Sharma, and Arjun Jain. 2018. Learning 3D Human Pose from Structure and Motion. In *European Conference on Computer Vision*. 668 – 683.

[3] Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the Knowledge in a Neural Network. *CoRR* abs/1503.02531 (2015).

[4] Sijin Li and Antoni B. Chan. 2014. 3D Human Pose Estimation from Monocular Images with Deep Convolutional Neural Network. In *Asian Conference on Computer Vision*.

[5] Julieta Martinez, Rayat Hossain, Javier Romero, and James J. Little. 2017. A Simple yet Effective Baseline for 3D Human Pose Estimation. In *International Conference on Computer Vision*.

[6] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis. 2017. Coarse-to-fine Volumetric Prediction for single-image 3D Human Pose. In *IEEE Computer Vision and Pattern Recognition*.

[7] Mir Rayat, Imtiaz Hossain, and James J Little. 2018. Exploiting Temporal Information for 3D Human Pose Estimation. In *European Conference on Computer Vision*. 69 – 86.

[8] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. 2019. Deep High-Resolution Representation Learning for Human Pose Estimation. arXiv:cs.CV/1902.09212

[9] Bugra Tekin, Isinsu Katircioglu, Mathieu Salzmann, Vincent Lepetit, and Pascal Fua. 2016. Structured Prediction of 3D Human Pose with Deep Neural Networks. In *British Machine Vision Conference*.

[10] Xiaowei Zhou, Menglong Zhu, Spyridon Leonardos, Konstantinos G Derpanis, and Kostas Daniilidis. 2016. Sparseness Meets Deepness: 3d human Pose Estimation from Monocular Video. In *IEEE Conference on Computer Vision and Pattern Recognition*. 4966 – 4975.